Imperial College London
Department of Electrical and Electronic Engineering
Communications and Signal Processing Group

# Building Detection for Monitoring of Urban Changes

Dimitrios Konstantinidis

January 2017

Supervised by Tania Stathaki

Submitted in part fulfilment of the requirements for the degree of Doctor of Philosophy in Electrical and Electronic Engineering of Imperial College London and the Diploma of Imperial College London

## Declaration of Originality

I, Dimitrios Konstantinidis, hereby declare that this thesis is my original work and all other sources used in this work are clearly and appropriately referenced in both text and references.

## Copyright Declaration

The copyright of this thesis rests with the author and is made available under a Creative Commons Attribution Non-Commercial No Derivatives licence. Researchers are free to copy, distribute or transmit the thesis on the condition that they attribute it, that they do not use it for commercial purposes and that they do not alter, transform or build upon it. For any reuse or redistribution, researchers must make clear to others the licence terms of this work.

# Abstract

This thesis addresses the problem of monitoring urban changes by decomposing it to building and change detection. HOG features, in combination with several other discriminative features, such as NDVI, FAST and LBP features, are employed in a search for the development of a robust and accurate building detector. Furthermore, a novel cosine-based distance function is introduced for the computation of distances between the SVM feature vectors in order to suppress the sensitivity of the SVM classifier to the presence of noise and outliers. Moreover, the transformation of SVM scores to probabilities and the definition of a better threshold that differentiates positive from negative feature vectors are proposed. To allow the transition from the object-based building detection to the pixel-based building delineation, a set of novel region refinement processes that includes an unsupervised image segmentation technique and the construction of building candidates by employing the most probable to correspond to buildings image regions based on a novel scoring procedure is proposed.

Taking advantage of the ability of the CNNs to automatically generate discriminative features, another approach to the problem of building detection involves the introduction of a *Modular*-CNN architecture. Two novel layers are proposed and added to the *Modular*-CNN architecture in order to improve its generalisation power and robustness. The change detection task is approached by a top-down approach that employs the computed building masks in order to identify building changes and a bottom-up approach that initially detects changes prior to their modelling using the computed building masks. In the change detection framework, we propose a novel change amplification algorithm that enhances the differences between the compared images in order to be more easily recognised and extracted. Finally, we propose a new robust change detector based on CNNs with the ability to automatically detect building changes, while discarding all other changes.

## Acknowledgements[1]

I would like to thank my principal supervisor, Prof. Tania Stathaki, for the opportunity she gave me to pursue this research, as well as for her guidance and advice in completing the thesis. I would also like to thank my collaborative supervisor at the Information Technologies Institute (ITI), Dr. Nikos Grammalidis for his supervision and support throughout the whole period of my research. Furthermore, I wish to thank Prof. Vasileios Argyriou (Kingston University) for his invaluable assistance in every aspect, with his knowledge and experience.

Finally, I would like to thank my family and my girlfriend Anastasia for their endless love, understanding and moral support and advice during the research and thesis writing periods.

[1]A major part of this thesis was supported by the European Union (European Social Fund-ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF)-Research Funding Program: THALES (UrbanMonitor): Reinforcement of the interdisciplinary and/or inter-institutional research and innovation. The Athens dataset was provided exclusively and solely under THALES (UrbanMonitor) research framework.

# Contents


**1 Introduction** **19**

1.1 Satellite imagery . . . 24
1.2 Contributions . . . 27
1.3 Publications . . . 27
1.4 Thesis organisation . . . 28

**2 Background** **31**

2.1 Building detection . . . 31
2.2 Convolutional neural networks . . . 37
2.3 Change detection . . . 40
2.4 Image pre-processing . . . 44
2.4.1 Multi-spectral band sharpening . . . 44
2.4.2 Image temporal registration . . . 45
2.5 Datasets . . . 46
2.5.1 Athens dataset . . . 46
2.5.2 CIFAR-10 dataset . . . 49
2.5.3 MNIST dataset . . . 50
2.5.4 AICD dataset . . . 51
2.6 Performance evaluation metrics . . . 52
2.7 Critical summary . . . 53
2.7.1 Building detection . . . 53
2.7.2 Change detection . . . 54
2.7.3 Comparison of state-of-the-art building detection algorithms . . . 55

**3 HOG-based Building Detection** **57**

3.1 HOG building detector . . . 59
3.1.1 HOG feature extraction . . . 59
3.1.2 HOG parameter optimisation . . . 63
3.1.3 Evaluation . . . 69
3.2 Improved HOG building detector . . . 72
3.2.1 Platt scaling . . . 73
3.2.2 Computation of NDVI features . . . 74

3.2.3 Computation of FAST features . . . 75
3.2.4 Fusion of HOG, NDVI and FAST features . . . 76
3.2.5 Evaluation . . . 78
3.3 HOG-LBP building detector . . . 84
3.3.1 Local binary patterns . . . 84
3.3.2 HOG-LBP feature extraction . . . 85
3.3.3 Evaluation . . . 88
3.4 Enhanced HOG-LBP building detector . . . 91
3.4.1 Cosine-based distance function . . . 92
3.4.2 Comparison of HOG-LBP building detector variants . . . 95
3.4.3 Comparison with other techniques . . . 99
3.5 Towards a pixel-based building detector . . . 102
3.5.1 Image segmentation . . . 104
3.5.2 Vegetation mask . . . 107
3.5.3 Water mask . . . 108
3.5.4 Building region detector . . . 110
3.5.5 Sensitivity to parameters . . . 113
3.5.6 Evaluation . . . 116
3.6 Comparison of HOG-based building detectors . . . 120
3.7 Summary . . . 120

**4 CNN-based Building Detection** **123**
4.1 *Modular*-CNN architecture . . . 125
4.2 Normalisation layer . . . 128
4.3 Euler transform layer . . . 129
4.4 PlainNet . . . 131
4.5 Evaluation of CNN architectures and novel layers . . . 132
4.5.1 Implementation details . . . 132
4.5.2 *Modular*-CNN optimisation on Athens dataset . . . 135
4.5.3 *Modular*-CNN testing on Athens dataset . . . 137
4.6 Summary . . . 139

**5 Change Detection** **143**
5.1 Top-down approach . . . 145
5.2 Bottom-up approach . . . 146
5.3 CNN-BCD algorithm . . . 149
5.4 Image pre-processing for change detection . . . 151
5.4.1 Illumination normalisation . . . 151
5.4.2 CHAM algorithm . . . 153
5.5 Evaluation of change detection techniques . . . 156
5.5.1 Comparison of illumination normalisation methods . . . 156
5.5.2 CHAM algorithm optimisation . . . 157

5.5.3 Optimisation of NCC and CNN-BCD algorithms . . . . . 160
5.5.4 Effect of CHAM algorithm on change detection performance . . . . 162
5.5.5 Comparison of building change detection approaches . . . . . 167
5.6 Summary . . . . . 168

**6 Conclusions** **171**
6.1 Future prospects . . . . . 173

**A Proof of Euler Transform's Equality to Cosine-based Function** **191**

**B *Modular*-CNN for CIFAR-10 and MNIST classification tasks** **193**
B.1 Experimentation on CIFAR-10 dataset . . . . . 193
B.2 Experimentation on MNIST dataset . . . . . 195
B.3 Conclusions . . . . . 196

# List of Figures


2.1 Multi-spectral and panchromatic image fusion. . . . . 45

2.2 Spectral resolution and band configuration for QuickBird and WorldView-2 satellites. Information taken from [1]. . . . . 47

2.3 Positive and negative training images from the Athens dataset are shown in first and second row respectively. . . . . 48

2.4 (a) QuickBird test image area_1a_2007 and (b) WorldView-2 test image area_2a_2010 of the Athens dataset. . . . . 49

2.5 Images of the CIFAR-10 dataset. . . . . 50

2.6 Images of the MNIST dataset. . . . . 50

2.7 Images of the AICD dataset. . . . . 51

3.1 The proposed HOG building detector. . . . . 60

3.2 The proposed HOG feature extraction procedure. . . . . 61

3.3 The two phases of the proposed rectangle grouping algorithm. . . . . 63

3.4 Illustration of the (a) first, (b) second and (c) third tested block configurations. 66

3.5 Illustration of the employed (a) rectangular and (b) circular kernels. . . . . 66

3.6 The optimisation of each HOG parameter is based on the performance of the proposed HOG building detector on the Athens "sample" validation set. 68

3.7 Detections from the proposed HOG building detector in (a) a QuickBird satellite image and (c) a WorldView-2 satellite image. The detections are shown as green boxes bounding the detected building areas. The binary masks of ground truth building locations (b) and (d) are also included for comparison, corresponding to images (a) and (c) respectively. . . . . 72

3.8 The proposed improved HOG building detector with the addition of NDVI and FAST features. . . . . 73

3.9 Illustration of the way $p_{FAST}$ is computed. The FAST corners are translated towards the centre of the region based on their gradient orientation. . . . . 76

3.10 Detections of the tested building detection methods shown as green bounding boxes. (a), (b) Part of QuickBird and WorldView-2 images respectively. Results from (c), (d) proposed HOG building detector, (e), (f) Ilsever detector [2], (g), (h) Sirmacek fusion [3] and (i), (j) improved HOG building detector. . . . . 83
3.11 The circular order of neighbouring pixels in a $3 \times 3$ LBP operator. . . . . 85
3.12 Block diagram of the proposed HOG-LBP building detector. . . . . 86
3.13 The previously employed cell configuration is shown in (a), while the new cells are shown in (b) and are added to the cell configuration of (a) to form the new cell configuration. . . . . 89
3.14 Comparison of detections from (a), (c) improved HOG building detector and (b), (d) HOG-LBP building detector on two images of the test set. Buildings that are detected or better delineated by the HOG-LBP detector are pointed out by yellow arrows, while false alarms rejected by the HOG-LBP detector are pointed out by red arrows. . . . . 91
3.15 Comparison of the $l_1$-norm, $l_2$-norm and the cosine-based distance function for various values of parameter $\alpha$. . . . . 93
3.16 Computation of distances between first row histogram and second and third row histograms using the $l_1$-norm, $l_2$-norm and the cosine-based distance function for various values of $\alpha$. . . . . 94
3.17 Computed sigmoid functions for transformation of SVM scores to probabilities in the cases of HOG-LBP, Linear HOG-LBP and Enhanced HOG-LBP detectors. The threshold of 0.5 is shown with a black line. . . . . 96
3.18 Performance of the proposed Enhanced HOG-LBP building detector with Platt scaling (M5) on the Athens "image" validation set for different values of parameter $\alpha$. (a) Effect of frequency $\alpha$ on object-based performance. (b) Effect of frequency $\alpha$ on pixel-based performance. . . . . 97
3.19 Presentation of the achieved $F_1$-score for the tested configurations of cascade adaboost classifiers on the Athens "image" validation set. (a) Cascade HAAR classifier. (b) Cascade LBP classifier. DAB: Discrete Adaboost, RAB: Real Adaboost, LB: Logitboost, GAB: Gentle Adaboost . . . . . 100
3.20 Average values of recall, precision and $F_1$-score and their corresponding standard deviations for the tested methodologies (M1-M8) at the object level in the Athens dataset. . . . . 102
3.21 Comparison of the tested methodologies in a satellite image depicting buildings with spectral variations in their rooftops (yellow circles) and buildings covered partially by vegetation (red circles). The initial image is presented in (a), while the results of algorithms M1-M8 are shown in (b)-(i) respectively. 103
3.22 Block diagram of the proposed pixel-based building detector, along with the defined modules and their interactions. . . . . 105
3.23 Block diagram of the proposed unsupervised image segmentation procedure. 106

3.24 Comparison of water detection methods in a problematic image of the Athens dataset. (a) Original image. (b) Ground truth of water locations. (c) Water mask produced by the NDWI index and threshold 0.3. (d) Water mask produced by our proposed water index using Eq. (3.40). . . . . 109

3.25 Block diagram of the building region detector, along with corresponding results of running the algorithm on a HOG-LBP detection. . . . . 110

3.26 Outline of the algorithm that forms the complete building candidates written in pseudo-code. . . . . 114

3.27 Performance of our proposed methodology on the Athens "image" validation set for different parameter settings. (a) Effect of overlap threshold with vegetation and water $T_{over}$, (b) effect of factor of deviation $F_{dev}$, (c) effect of size of smallest accepted region $T_{small}$ and (d) effect of rectangularity threshold $T_{rect}$ on pixel-based performance. . . . . 115

3.28 Comparison of methods M9 and M10 in (a) a QuickBird and (b) a WorldView-2 image of the Athens dataset. (c), (d) Detected buildings from our method M9 for images (a), (b) respectively. (e), (f) Detected buildings from the method M10 for images (a), (b) respectively. True positives are shown in green, false positives are shown in blue and missed pixels are presented in red. . . . . 118

4.1 The proposed basic unit or *module* of the *Modular*-CNN architecture. . . . 126

4.2 Behaviour of convolutional layer of a CNN. . . . . 126

4.3 Plots of (a) hyper-tangent (tanh), (b) sigmoid and (c) ReLU activation functions. . . . . 127

4.4 General architecture of the proposed *Modular*-CNN. Optional *modules* are presented in dashed lines . . . . . 128

4.5 Novel NL layer. The size of the output feature vector is twice the size of the input feature vector $D_n$. . . . . 129

4.6 Novel ETL layer. The size of the output feature vector is twice the size of the input feature vector $D_e$. . . . . 130

4.7 Architecture of PlainNet. In the convolutional layers, the number of output filters is shown last, while "/2" means a stride of 2. The ReLU layers after each convolutional and linear layer are omitted. . . . . 131

4.8 Experimentation on the Athens "sample" validation set with respect to (a) the dimensionality of output feature vectors, (b) batch size and (c) learning rate. . . . . 134

4.9 Evolution of training and validation loss during the *Modular*-CNN training. Over-fitting is prevented by stopping the CNN training in the epoch shown with a red dashed vertical line. . . . . 134

4.10 Optimal *Modular*-CNN architecture for the Athens remote sensing dataset. In the convolutional layers, the number of output filters is shown last, while in the fully connected blocks, the transition from $M$ to $N$ features is depicted. The ReLU units after each convolutional and linear layer are omitted. 136

4.11 Performance of the *Modular*-CNN and PlainNet with and without the novel NL and ETL layers with respect to the average loss on the Athens "sample" validation set. . . . . 136

4.12 Flowchart of the proposed building detection methodology. . . . . 138

4.13 Detections shown as green bounding boxes in two test images of the Athens dataset from (a), (b) the method M5 and (c), (d) the combined approach of our *Modular*-CNN and the method M5. . . . . 139

4.14 Detected building regions from (a), (b) the method M9 and (c), (d) the combined approach of our *Modular*-CNN and the method M9 in two images of the Athens test set. True positives are shown in green, false positives are shown in blue and missed pixels are presented in red. . . . . 140

5.1 The diagram of a top-down approach applied in the building change detection task. . . . . 145

5.2 The diagram of a bottom-up approach applied in the building change detection task. . . . . 146

5.3 Architecture of the proposed CNN-BCD algorithm. The term "p 1" denotes zero padding with 1 pixel. ReLU units that are employed after each convolutional and linear layers are omitted. . . . . 150

5.4 Outline of the proposed CHAM algorithm. The depicted images are presented slightly sharpened and contrast enhanced for better visualisation. . . 154

5.5 Comparison of ideal distributions (red) and distributions of change map values of "change" and "no-change" classes (blue) before (a), (b) and after (c), (d) the standardisation method. The Euclidean distances of the distributions are shown at the top of the figures. . . . . 158

5.6 Performance of the CHAM algorithm on the Athens change validation set for various values of amplification factor $m$ with respect to the average metrics of $F_1$-score and Kappa. . . . . 159

5.7 Effect of the CHAM algorithm on change maps: (a), (b) Compared images, (c) ground truth differences and change maps from (d) CVA method, (e) CVA with CHAM + LPR and (f) CVA with CHAM - LPR. . . . . 160

5.8 Performance of the CNN-BCD algorithm on the validation set for various values of $\alpha$. . . . . 162

5.9 Performance of the tested change detection methods on a pair of images of the Athens dataset. The pair of images is shown in (a), (b), while their ground truth differences are shown in (c). The result of the CHAM pre-processing of images (a) and (b) is shown in (d) and (e) respectively. The change maps of image differencing, regression, median background subtraction, CVA and NCC without the CHAM pre-processing are shown in (f)-(j) respectively, while the change maps of the methods with the CHAM pre-processing are presented in (k)-(o) respectively. . . . . 164
5.10 Performance of the NCC algorithm on the AICD dataset with and without the CHAM algorithm for various viewpoint angles. . . . . 164
5.11 Comparison of CVA and CNN-BCD algorithms with and without the CHAM algorithm. The compared images are shown in (a) and (b), while the CVA results without and with the CHAM algorithm are presented in (c) and (d) respectively. The CNN-BCD results without and with the CHAM algorithm are depicted in (e) and (f) respectively. True, false and missed changes are shown in green, red and blue respectively. . . . . 166
5.12 Performance of change detection approaches on a pair of images of the Athens change test set. The pair of images is depicted in (a), (b), while the building change maps computed from (c) CVA with BMs, (d) CNN-BCD, (e) CNN-BCD with BMs and (f) BM subtraction approaches are shown as well. . . . . 169

B.1 Optimal *Modular*-CNN architecture for the CIFAR-10 dataset. In the convolutional layers, the number of output filters is shown last, while in the fully connected blocks, the transition from $M$ to $N$ features is depicted. The ReLU units after each convolutional and linear layer are omitted. . . . 194
B.2 Performance of the *Modular*-CNN and PlainNet with and without the novel NL and ETL layers with respect to the average loss on the validation set for the CIFAR-10 dataset. . . . . 195
B.3 Optimal *Modular*-CNN architecture for the MNIST dataset. In the convolutional layers, the number of output filters is shown last, while in the fully connected blocks, the transition from $M$ to $N$ features is depicted. The ReLU units after each convolutional and linear layer are omitted. . . . . 196
B.4 Performance of the *Modular*-CNN and PlainNet with and without the novel NL and ETL layers with respect to the average loss on the validation set for the MNIST dataset. . . . . 197

# List of Tables


2.1 Test set of the Athens dataset. QuickBird images are shown as QB, while WorldView-2 images are shown as WV2. The images of the Athens "image" validation set are presented in bold. . . . . 48

2.2 Comparison of state-of-the-art building detection algorithms. OB: Object-based, PB: Pixel-based, MBR: Minimum Bounding Rectangle, MSPA: Morphological Segmentation of binary Patterns, NOAA: National Oceanic and Atmospheric Administration, Prec.: Precision . . . . . 56

3.1 Key characteristics of proposed building detection methodologies. . . . . 59

3.2 Results of the proposed HOG building detector on the Athens test set. . . . 71

3.3 Results, presented in format: mean ± std, of our improved HOG building detector on the Athens dataset for the tested decision rules . . . . . 78

3.4 Performance of the improved HOG building detector on the Athens dataset for all feature combinations. The results are presented in format: mean ± std. . . . . 78

3.5 Comparison of the tested building detection methodologies on the Athens dataset. . . . . 81

3.6 LBP histogram size per cell for different types of LBP features. . . . . 88

3.7 Results, presented in format: mean ± std, of the proposed HOG-LBP building detector on the Athens dataset for combinations of HOG and LBP variants. . . . . 88

3.8 Effect of proposed differentiations on the performance of the HOG-LBP building detector on the Athens dataset. The results are presented in the format: mean ± std. . . . . 90

3.9 Values of Platt parameters for the proposed HOG-LBP building detectors. . 96

3.10 Comparison of the HOG-LBP building detector variants on the Athens dataset. . . . . 98

3.11 Performance of the tested methodologies at object and pixel level in the Athens dataset. The results are presented in the format: mean ± std. Best values are shown in bold. . . . . 102

3.12 Parameter settings for the proposed approach. . . . . 115

3.13 Results of pixel-based building detectors on the Athens dataset. . . . . 117
3.14 Performance of the proposed HOG-based building detectors on the Athens dataset. The results are presented in the format: mean ± std. BD: building detector . . . . . 120

4.1 CNN hyper-parameter value pool. . . . . 132
4.2 Classification performance on the Athens dataset. . . . . 137
4.3 Performance of building detectors on Athens test set. M-CNN: *Modular*-CNN 138

5.1 Evaluation of illumination normalisation methods. . . . . 157
5.2 Performance of CHAM algorithm for various phase-amplified spectral channel combinations. The results are presented in the format: mean ± std. . . 158
5.3 Performance of CVA method with and without the CHAM algorithm. The results are presented in the format: mean ± std. . . . . 160
5.4 Performance of NCC algorithm for various neighbourhood sizes $n$. The results are presented in the format: mean ± std. . . . . 161
5.5 Performance of change detection methods on Athens dataset. The results are presented in the format: mean ± std. . . . . 163
5.6 Performance of change detection methods on AICD dataset. The results are presented in the format: mean ± std. . . . . 163
5.7 Performance of CVA and CNN-BCD algorithms on the Athens change test set. The results are presented in the format: mean ± std. . . . . 165
5.8 Performance of building change detectors on the Athens change test set. The results are presented in the format: mean ± std. . . . . 168

B.1 Classification performance on the CIFAR-10 dataset. . . . . 194
B.2 Classification performance on the MNIST dataset. . . . . 196

# List of Abbreviations

**AICD** Aerial Imagery Change Detection.

**ANN** Artificial Neural Network.

**BM** Building Mask.

**CHAM** Change Amplification.

**CNN** Convolutional Neural Network.

**CNN-BCD** CNN Building Change Detector.

**CPU** Core Processing Unit.

**CS-LBP** Center-Symmetric LBP.

**CVA** Change Vector Analysis.

**EM** Expectation-Maximisation.

**ETL** Euler Transform Layer.

**FAST** Features from Accelerated Test.

**FN** False Negative.

**FP** False Positive.

**GMSR** Gradient-Magnitude-based Support Regions.

**GPS** Global Positioning System.

**GPU** Graphical Processing Unit.

**HOG** Histogram of Oriented Gradients.

**HSV** Hue-Saturation-Value.

**ID-LBP** Improved Directional LBP.

**IGV** Internal Gray Variance.

**ILSVRC** ImageNet Large Scale Visual Recognition Challenge.

**LBP** Local Binary Patterns.

**LPR** Low-pass Residual.

**MLP** Multi-layer Perceptron.

**NCC** Normalised Cross-Correlation.

**NDVI** Normalised Difference Vegetation Index.

**NDWI** Normalised Difference Water Index.

**NIN** Network in Network.

**NIR** Near-infrared.

**NL** Normalisation Layer.

**PCA** Principal Component Analysis.

**PR** Precision-Recall.

**RBF** Radial Basis Function.

**ReLU** Rectified Linear Unit.

**RGB** Red-Green-Blue.

**RR** Region Refinement.

**SIFT** Scale-invariant Feature Transform.

**SVD** Singular Value Decomposition.

**SVM** Support Vector Machine.

**TN** True Negative.

**TP** True Positive.

**VOC** Visual Object Classes.

# CHAPTER 1

## Introduction

Changes in human societies occur with fast pace and they are manifested in various ways. One of the most significant type of changes is urban changes. Since the beginning of humanity, human beings are driven by economic, social, environmental and other factors to change their habitat. Especially in modern ages, there is a boom in urbanisation, meaning that humans move from rural to urban areas because of modernisation and industrialisation. This mass movement of human beings is usually accompanied by large changes in an urban environment and these changes are usually manifested by building construction and destruction. A building is a vital part of the life of a human being as it plays the role of a place of living or work. As a result, the relocation of human beings leads to new buildings being constructed and others getting demolished. This means that buildings or more generally the urban infrastructure reflects and highlights human activities in an area. New buildings are constructed and others are abandoned and get demolished as the population of an urban area fluctuates.

The huge increase of the population of earth leads to a similar escalation of the human activities. Since such activities can be directly linked to construction operations in an area, the changes in an urban environment are dramatically increased. The speed, with which urban changes occur, has led to serious problems in the urban change detection and mapping. This fact leads us to the problem of urban change monitoring that we as society have to face. As urban change monitoring, we refer to the task of identifying and monitoring changes that occur in an urban environment. Changes in road infrastructure, housing and other man-made structures belong to the category of urban changes. However, in the framework of this thesis, we focus our attention on changes in building infrastructure as they are denoted by building construction and destruction.

Building detection from satellite images was almost impossible a few decades ago due to low resolution satellite images that did not allow the identification of individual buildings in an image. Therefore, building detection could only be achieved by employing aerial images or as part of the general problem of land cover classification from satellite images that

studied and evaluated urban growth. However, the last two decades, there were significant advances in the technology of the sensors satellites carry and nowadays the capture of high-resolution multi-spectral satellite images is feasible. These high-resolution images depict the urban environment with great detail, making the detection and classification of individual buildings and other man-made structures from satellite images possible and more accurate than ever before.

Building detection from 2D high-resolution satellite images is a computer vision, photogrammetry and remote sensing task that can be utilised in several applications that require the creation of urban maps or the study of urban changes. A few important application areas, where the development of a system capable of monitoring and modelling urban changes can find usefulness are [4, 5]:

- **Sociology:** The modelling of urban changes and the creation of digital urban maps can greatly assist sociologists towards monitoring the dynamic processes that occur in a complex urban environment. Sociologists can possibly investigate the prosperity or decline of a population as it is depicted in the building expansion or destruction respectively. Afterwards, appropriate assumptions can be made, correlations with social, economic or environmental reasons can be found and certain measures to accelerate the expansion or restrain the decline of an urban area can be taken.

- **Citizen welfare:** The development and constant update of digital urban maps can be extremely helpful for municipalities and city planning services. A city that is constantly expanding can be well-prepared to offer its citizens a comfortable life by organising its services and sources to accommodate the expected population growth. For instance, new schools and hospitals can be built or the electricity and water supply systems can be expanded to fulfil the needs of the spreading population. Moreover, the road infrastructure can be properly and timely developed to provide access to every new building that is constructed in an urban area.

- **City protection:** A significant application of urban change detection is the analysis and assessment of the impact of fire, flood and natural disasters in an urban environment. Police can protect every citizen by knowing how extended the borders of a city are and where people are located. In case of natural disaster or catastrophic accident, city officials, police and fire-fighters can be well-prepared to take necessary measures and precautions in order to help anyone in need, providing they are aware of their whereabouts. In the unfortunate event of an emergent evacuation, such vital information can surely save a lot of lives. Furthermore, the extent of damage inflicted by a catastrophic event can allow municipalities to act appropriately in order to avoid similar disasters in the future. Finally, the abandonment and destruction of buildings in an area can trigger authorities to investigate the possible reasons behind it (i.e. crime rate, pollution, etc.) and come up with certain measures to counteract an arising problem before it further grows and causes severe complications.

- **Illegal construction:** The difficulty in detecting and mapping buildings has contributed towards a growth in illegal building activity. More and more building structures are constructed in forbidden areas, sometimes destroying the natural environment in the process. The creation and constant update of urban maps can not only put a halt to such illicit activities, but also lead to an exemplary punishment of those responsible, contributing as a deterrent for future illegal building construction and preserving the beauty of the natural environment of an area.

- **Navigation:** The accurate modelling of urban changes can assist in the development and constant update of accurate urban maps that can be employed for navigation purposes. This means that companies that specialise in Global Positioning System (GPS) applications in vehicles, mobile phones, etc. can provide to their clients more reliable services, given an access to accurate urban maps.

Although building detection can be achieved manually by human experts, the tediousness of the process, the required time needed to do manual labelling and the speed with which modern cities expand, make the development of automatic, accurate and computationally fast building detection algorithms imperative. Unfortunately, building detection can be a really challenging task. This is because there are many obstacles that need to be overcome to achieve accurate and robust building detection. The challenges and limitations current building detection methodologies have to face are listed below [4, 6]:

1. *Diversity:* The variety and nature of objects appearing in an urban environment is so vast and unpredictable that it is hard to come up with an error-tolerant methodology for building detection and segmentation. In a high-resolution image, there are many objects of various sizes and shapes that are distinguishable and need to be classified, such as roads, cars, swimming pools, crop fields, buildings and vegetation.

2. *Variety of buildings:* Buildings in an urban environment appear in many different sizes and shapes and present a variety of colours and textures, partly due to the different spatial and spectral resolution of the satellite sensors, making it impossible to define a template that characterises most buildings. Moreover, the spectral and spatial resolution differences of satellite images can affect the size and colour of buildings. Finally, human activities and weather conditions (i.e. sun angles, shadows, etc.) can change the size, colour and shape of a building, increasing the difficulty for developing a universal building extraction methodology.

3. *Sensor limitations:* The satellite images that sensors provide, can be severely corrupted with noise. Such noise can lead to great inaccuracies when building detection and segmentation is attempted.

4. *Occlusions:* Buildings can be occluded by other buildings or high vegetation (i.e. trees), making it hard to infer their exact size and shape and proceed to their accurate segmentation. Furthermore, clouds during the acquisition of satellite images can

affect the visibility of buildings and other structures in the captured images.

5. *Shadows:* Shadows are known to complicate several computer vision tasks, such as object detection and image segmentation. Their detection and removal is desirable since shadows have the shape of the objects that cast them and therefore shadows can be mistakenly recognized as object candidates. Furthermore, shadows alter the intensity of objects they are cast on and they can thus affect the accurate segmentation of these objects. However, automatic shadow detection and removal is very challenging since shadows are the outcome of complex interactions between the geometry, illumination, and reflectance present in a scene and there is usually limited information about these properties of a scene.

6. *Availability of data:* Depth information can significantly assist in a building detection task as height can easily differentiate buildings from smaller objects present in an image. However, sensors that provide 3D data are expensive and thus 3D data are usually hard to acquire. Furthermore, different sensors provide different multi-spectral information and thus certain multi-spectral channels may not be available during the processing of a multi-spectral image.

7. *Time efficiency:* On the one hand, high-resolution images enable the development of a more accurate and robust building detection system, but on the other hand, they introduce large amount of data, whose processing can tremendously affect the efficiency of an algorithm. As a result, there is a need for more time-efficient algorithms that can be used in real-time remote sensing applications.

8. *Registration:* In order to compare images and identify differences, images should be properly registered. This means that the images should be aligned so as the same objects appear in the same positions in the compared images. Such a registration is not always an easy task and errors can be introduced that can lead to the degradation of the performance of an urban change detector.

Because of these reasons, accurate extraction of buildings and other man-made structures, along with a robust modelling of urban changes is not an easy problem to solve and is currently accomplished by tedious and error-prone manual or semi-manual techniques. Therefore, there is an urgent need for a system capable of achieving robust performance on the modelling of changes in man-made structures by tackling the problems that were previously mentioned. It is imperative to go beyond the current manual photo-interpretations or semi-automated processing procedures to a paradigm where an automated processing algorithm can handle and register multi-temporal, multi-modal data and can efficiently detect, extract and model urban changes.

In order to successfully identify and model urban changes, this work decomposes the problem to two main tasks that are solved separately; building and change detection. Building detection is concerned with the identification of image regions that describe buildings.

To this end, building detection methodologies are studied and analysed in an attempt to overcome most of the problems and limitations that were previously mentioned and affect state-of-the-art building detection algorithms. The purpose of the building detection task is the development of a robust and accurate building recognition system that operates at the object or pixel level and is capable of identifying and extracting candidate building regions. Two popular and successful object detection approaches are examined and implemented. The first approach is based on Histogram of Oriented Gradients (HOG) [7] features. HOG features were initially introduced for human detection and since then, they were employed in several object detection frameworks, managing to achieve state-of-the-art performances [8, 9, 10]. The discrimination ability of the gradient magnitudes and orientations is very useful for the detection of buildings. Furthermore, HOG features are robust to spectral variations and they can be used to detect a variety of building shapes, given the fact that the training is performed on a representative set of possible shapes. Moreover, an algorithm based on HOG features is able to detect buildings of various sizes as it usually runs on multiple scales and buildings that are partly occluded by other buildings, vegetation, clouds or shadows due to their robustness on slight shape deformations. The second class of building detection methods that is examined is Convolutional Neural Network (CNN). CNNs saw heavy use the last years due to their ability to produce descriptive features on their own, without human intervention. Recent algorithms based on CNNs demonstrate state-of-the-art performances in several object detection tasks [11, 12, 13, 14].

The second task, this work deals with, is change detection. The purpose of this task is the development of a robust and accurate system that detects important changes in an urban environment. Depending on the task that needs to be solved, some urban changes are considered important and others unimportant. In the framework of this thesis, important are the changes in size and shape of the buildings present in an image, while all other changes should be suppressed and discarded. The purpose of the change detection task is the development of a robust change detection system that is capable of comparing two registered images depicting the same urban environment and captured in different time frames, identifying changes and evaluating and modelling these changes in order to preserve and highlight only the changes that are related to building infrastructure. In the case where images are not registered, image registration techniques can be applied to align the compared images so as the same buildings appear in the same image locations.

Change detection can be approached in either a top-down or a bottom-up fashion. In top-down approaches, the objects of interest are initially detected before the changes that are related to the objects of interest are identified and modelled. On the other hand, in bottom-up approaches, all changes are initially identified and then these changes are examined in order to conclude which changes refer to important changes, where the importance of a change is defined according to the task that needs to be solved. Both approaches have their advantages and disadvantages. For instance, bottom-up approaches are implemented in a straightforward manner, changes can be easily located but the modelling of the detected

changes, which is the identification of the type of changes, can be pretty challenging. On the contrary, top-down approaches are hard to implement as the objects of interest should be accurately and robustly detected. However, given the successful and accurate extraction of the objects of interest, changes can then be easily and robustly identified and modelled. In this work, emphasis is mostly put on top-down approaches as the largest part of this thesis is concerned with the accurate and robust building detection. Changes can then be easily identified by comparing the extracted building masks. However, bottom-up approaches are also examined and compared to top-down change detection methods.

The methodologies proposed in the framework of this thesis are designed to achieve the highest possible accuracy in the building and change detection tasks, while less attention is paid on their time efficiency. There is usually a trade-off between classification performance and time efficiency and we believe that for applications that require the creation of accurate urban maps, accuracy should be a priority. Moreover, due to the fact that we process high-resolution satellite images in this thesis, shadows are visible and can affect the detection and segmentation of buildings. However, the distance between the satellite sensors that capture an image and the ground terrain is so large that shadows appear to have small size. As a result, their effect on building detection is minimal and they cannot be easily mistaken as buildings. We decide not to deal directly with the problem of shadow detection and removal. Instead, we rely on HOG features to suppress shadow effects. Moreover, a region scoring algorithm that we propose in this thesis and selects candidate building regions based on several criteria, such as size and local saliency, is responsible for de-valuating shadow regions. As far as change detection is concerned, shadows play a significant role, since they alter the pixel intensities, and thus they are mistakenly recognized as changes. To deal with this problem, we decide to amplify changes on a colour channel, thus increasing the magnitude of changes between regions altered by the appearance or disappearance of objects and regions affected by shadows. This strategy enables the suppression, but not the complete elimination of the effect of shadows that can still affect change detection results.

## 1.1 Satellite imagery

Satellite imagery refers to the task of taking images of earth from sensors placed on artificial satellites that are in orbit around the earth. The satellite images provide important information that can be used in several remote sensing applications, such as meteorology, oceanography, cartography, agriculture observation, urban change detection, etc. According to Campbell in [15], the satellite images can be characterised and categorised by four types of image resolution:

1. *Spatial Resolution:* The spatial resolution of a satellite image refers to the size in $m^2$ of the surface area of the earth, being measured on the ground, that an image pixel

covers. However, the most common metric that is used to describe spatial resolution is $m$, which describes the ground distance each side of a square image pixel depicts. The spatial resolution is affected by the height of the orbit of the satellite around the earth and the type of sensors placed on the satellite. The higher the spatial resolution of an image is, the smaller the surface area a pixel depicts. As a result, a satellite image of high spatial resolution allows for a detailed view of the terrain of the earth, since even small objects are captured and become distinguishable.

2. *Spectral Resolution:* The spectral resolution of a satellite image is defined by the number of spectral bands/channels it contains and the wavelength interval size measured in $nm$ that each band of the satellite image captures. This means that the spectral information extracted from each satellite can vary. Most satellites can provide images with at least 4 channels, where aside of the basic Red-Green-Blue (RGB) channels, a near-infrared channel is also included. For instance, the IKONOS satellite provides 4-channel satellite images, while the LANDSAT-7 satellite captures 7-channel images. Furthermore, even the wavelength interval of the same spectral channel can vary from sensor to sensor and this means that the same spectral channel may contain slightly different information when captured from different sensors. For example, the green channel of the QuickBird satellite lies in the interval $520-600nm$, while the green channel of the WorldView-2 satellite lies in the interval $510-580nm$.

3. *Temporal Resolution:* The temporal resolution of a satellite image refers to the amount of time, usually counted in days, that a satellite needs in order to perform a full circle around the earth and return to the same location as it was before in order to collect images of the same surface area of the earth. This type of resolution mainly depends on the orbit of the satellite around the earth.

4. *Radiometric Resolution:* The radiometric resolution of a satellite image refers to the levels of brightness the sensors on the satellite can capture. The radiometric resolution is described in bits and the most typical values are 8, 11, 12 and 16 bits. The range of values that the pixels of a satellite image can get, depends on the radiometric resolution of the sensor employed to capture the image. More specifically, given that the radiometric resolution of an image is $n$ bits, the intensity of the image pixels are in the range $[0, 2^n - 1]$. Consequently, the pixel values of a 8-bit resolution image belong to the range of $[0 - 255]$, while the pixel values of a 16-bit resolution image belong to the range of $[0 - 65535]$.

The acquisition of satellite imagery initiated as early as 1972 with low-resolution satellite images becoming available and continued with better sensors that enable today the acquisition of high-resolution satellite images. The longest-running and most well-known earth observation satellite system, Landsat, was developed by NASA [16]. In 1972, Landsat 1 was launched allowing the capturing of low-resolution images with a spatial resolution of $60m$. More technologically advanced satellites followed, improving the spatial resolution of

the acquired images. Landsat 8, which was launched in 2013, is currently the last satellite of the Landsat series and allows the capturing of images of panchromatic spatial resolution equal to $15m$ and multi-spectral spatial resolution equal to $30m$ [1].

Another known earth observation satellite program, SPOT, was initially developed by the French Space Agency [17]. The first SPOT satellite was launched in 1986 and provided images with panchromatic spatial resolution equal to $10m$ and multi-spectral spatial resolution equal to $20m$. The next SPOT satellites significantly improved their spatial resolution. More specifically, SPOT 5 was launched in 2002 and provided images with panchromatic spatial resolution equal to $2.5 - 5m$ and multi-spectral spatial resolution equal to $10m$. SPOT 6 and 7 that were launched after 2012 and are still operational provide images with panchromatic spatial resolution equal to $1.5m$ and multi-spectral spatial resolution equal to $6m$ [1].

Currently, one of the most known providers of high-resolution satellite images is DigitalGlobe. The IKONOS satellite that was launched in 2000, is a commercial earth observation satellite and the first to collect publicly available high-resolution images of panchromatic spatial resolution equal to $1m$ and multi-spectral spatial resolution equal to $4m$ [1]. Other satellites followed, such as QuickBird, WorldView-1 and WorldView-2 that improved the spatial resolution of images, reaching a panchromatic spatial resolution of $0.5m$ and a multi-spectral spatial resolution of $2m$. Next-generation satellites, such as the WorldView-3 that was lauched in 2014 by DigitalGlobe, allow the capturing of very high-resolution satellite image with panchromatic spatial resolution equal to $0.31m$ and multi-spectral spatial resolution equal to $1.24m$ [1].

Although high-resolution imagery allows the capturing of great details on the surface area of the earth and thus the detection of small structures, such as buildings, is now possible, high-resolution images introduce significant limitations. To begin with, a high-resolution image consists of a very large number of pixels and the processing of such amount of pixels can be very time-consuming. Moreover, weather conditions and clouds can severely affect the quality of the images obtained from satellites. Finally, a huge disadvantage when dealing with remote sensing applications is the availability of data. Satellite images, especially those captured from high-resolution satellite sensors (IKONOS, QuickBird, WorldView-1,-2,-3, etc.), are products that are sold for a price and these images are usually accompanied with a license that prohibits their owners of rights to copy, distribute or use in applications other than those that are bought for. As a result, techniques that are developed to process satellite images are limited to be tested only on the dataset that is available. This means that it is hard to compare satellite image processing methodologies on different datasets because there are no public datasets with free access and the comparison can only be performed on the available dataset.

## 1.2 Contributions

The original major contributions that have been achieved during this thesis are summarised in this section:

- A novel probabilistic fusion of HOG, Normalised Difference Vegetation Index (NDVI) and Features from Accelerated Test (FAST) features in an attempt to improve the accuracy of an object-based building detector.
- A novel cosine-based distance function that replaces the $l_2$-norm function during the training and testing of a Support Vector Machine (SVM) classifier and improves its robustness in the presence of noise and outliers.
- A proposed scaling of the SVM scores to probabilities and the definition of a new threshold to differentiate between positive and negative feature vectors.
- A set of novel region refinement procedures that enables the transition from an object-based to a pixel-based classification result. These procedures include an unsupervised image segmentation algorithm that automatically determines the optimal number of classes an image can be split, a novel image region scoring procedure for the selection of initial building candidates and a proposed building formation algorithm that merges regions to form complete buildings.
- A Normalisation and an Euler transform layers for the improvement of the accuracy of a CNN-based building detector. The Normalisation layer takes advantage of both raw and normalised features to improve the generalisation ability of the CNN-based building detector, while the Euler transform layer concerns the Euler transformation of the feature vectors as an alternative way of utilising the novel cosine-based distance function for improved classification robustness to noise and outliers.
- A novel method of enhancing changes and improving the performance of change detection algorithms by magnifying the phase differences of the Fourier transforms of image pairs.
- A robust and accurate change detection algorithm based on CNNs with the ability to recognise only changes that correspond to buildings.

## 1.3 Publications

The original contributions in this thesis have led to the following peer-reviewed publications:

**Journals**

- D. Konstantinidis, V. Argyriou, T. Stathaki, and N. Grammalidis, *Phase Amplified Correlation for Improved Sub-pixel Motion Estimation*, IEEE Transaction on Image Processing, 2017 (under preparation).
- D. Konstantinidis, T. Stathaki, V. Argyriou, and N. Grammalidis, *Building detection using enhanced hog-lbp features and region refinement processes*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 10, no. 3, pp. 888-905, 2017.

**Conferences**

- D. Konstantinidis, T. Stathaki, V. Argyriou, and N. Grammalidis, *A probabilistic feature fusion for building detection in satellite images*, in Proceedings of the 10th International Conference on Computer Vision Theory and Applications (VISIGRAPP 2015), pp. 205-212, 2015.
- D. Konstantinidis, T. Stathaki, V. Argyriou, and N. Grammalidis, *A 3D Feature for Building Segmentation based on Shape-from-Shading*, in Proceedings of the 10th International Conference on Computer Vision Theory and Applications - Volume 2: MMS-ER3D, (VISIGRAPP 2015), pp. 595-602, 2015.

## 1.4 Thesis organisation

The structure of the rest of the thesis is as follows:

Chapter 2 serves to review and critically discuss related work of other researchers in the topics of building detection, convolutional neural networks and change detection and to clarify the way their research motivated us and led us to the contributions presented in this thesis. Furthermore, the datasets and the performance evaluation metrics that are employed in this thesis are defined.

Chapter 3 presents the developed building detection methodologies based on HOG features. Starting from optimised to the building detection task HOG descriptors, we investigate other features and techniques to improve the accuracy and robustness of the building detection methods. To this end, we analyse the effect of various discriminatory features (NDVI, FAST and Local Binary Patterns (LBP)) on the performance of a building detector based on HOG features. Furthermore, we propose a novel distance function that replaces the $l_2$-norm in the training and classification of the feature vectors and increases the robustness of the building detector in the presence of noise and outliers. Additionally, we propose the scaling of the SVM scores to probabilities and the definition of a new threshold that decreases the number of false detections of a building detector. Finally, we propose a region refinement procedure that accepts the object-based results of a building detector and allows the accurate delineation of building boundaries, thus improving the pixel-based performance of a building detector. Among the region refinement procedures,

an unsupervised image segmentation algorithm, the identification of initial building candidates based on a novel region scoring procedure and a proposed final building formation algorithm are worth mentioning.

Chapter 4 deals with the problem of building detection using convolutional neural networks. A general shallow CNN architecture is adopted and it is optimised based on the dataset that needs to be classified. Two novel layers are proposed to improve the performance of a CNN classifier on the building detection task. The first layer is based on feature normalisation, while the second is based on the Euler transformation of feature vectors and utilises the novel distance function that was proposed in Chapter 3. The proposed shallow CNN is compared with a deeper CNN on a building and two benchmark object classification datasets and the results are analysed. Finally, the proposed shallow CNN is utilised, in conjunction with the optimal HOG-based building detector, proposed in Chapter 3, and an accurate and robust building detector that takes advantage of both the power of CNN and HOG features is proposed.

Chapter 5 deals with the problem of change detection. Based on a top-down approach, we detect changes by employing the differences of the detected building regions. Afterwards, based on a bottom-up approach, several well-known pixel-based change detection methodologies are examined and their performance on the change detection datasets are evaluated. To improve the performance of the change detection techniques, we propose a novel change amplification algorithm that is based on the transformation of images to the Fourier domain and the magnification of their phase differences. Finally, an accurate building change detection approach based on CNNs is proposed and its performance is compared with the other change detection approaches that were previously mentioned.

Chapter 6 concludes the work of this thesis by summarising the original contributions achieved during this thesis. Furthermore, a discussion is made on how future research on ways to improve the robustness of object detectors and enhance changes can benefit researchers and lead to advances on several image processing application areas, including urban change monitoring.

# CHAPTER 2

# Background

In the previous chapter, we introduced the problem of urban change monitoring, we presented its application areas and challenges and we decomposed the problem to two main tasks; building and change detection that we decided to address separately. In this chapter, we present a literature review of the related research performed in the topics of building detection, image segmentation, convolutional neural networks and image change detection. Furthermore, we make a critical discussion of state-of-the-art building and change detection methodologies and present their advantages and drawbacks. Moreover, we revise two necessary image pre-processing techniques that should be applied prior to the employment of building and change detection algorithms. Finally, we define the datasets and metrics that are employed for the evaluation and comparison of the performance of the proposed building and change detection methodologies.

## 2.1 Building detection

Building detection is considered as the problem of identifying and extracting building regions from images using computer vision and image processing techniques. It is an extremely important, yet challenging task for several remote sensing applications since buildings appear in various sizes and shapes and the appearance of buildings can be affected by weather conditions and human actions. As a consequence, developing a universal methodology that can be applied in a large variety of images can be immensely difficult.

Several methodologies have been proposed to solve the problem of building detection. These methodologies can be classified in two major categories based on the dimensionality and processing method of the available data. The first category consists of algorithms that process 3D data in order to detect buildings. Data that belong to this category are LiDAR point clouds and Digital Surface Models, which acquired from laser/radar sensors, provide information about the height of a terrain. As a result, they can be considered as

coarse 3D representation of the studied urban environment.

In the presence of 3D data, building extraction can be achieved either by a suitable height threshold [18] or by 2D line detection and then mapping to 3D space [19]. In [20] and [21], the planes in the 3D space were identified as a means to describe the rooftops of the buildings. 3D templates were utilised by Karantzalos and Paragios in [22] and Hammoudi and Dornaika in [23], as a way to identify buildings. Since the height of buildings is an important cue towards their differentiation from other objects that appear in an urban environment, methods that rely on 3D data can achieve high accuracy and precision as only few tall trees can be mistakenly recognised as buildings based on the height information. The disadvantage of such methodologies is that the 3D data can be quite inaccurate, introducing significant errors to the building detection task. This is because 3D data depend highly on the resolution of the sensors and the observing procedures used for the data collection. Furthermore, abrupt changes in the height of the terrain can lead to inconsistencies in the height information of neighbouring pixels. Finally, 3D data are usually hard to acquire because of the high cost of acquisition and therefore, most methods cannot take advantage of 3D information.

In the second category of building detection methods, there are algorithms that process 2D data, in the form of multi-spectral images. These algorithms can be classified in two general subcategories based on whether they process images at the object or pixel level. Pixel-based methodologies analyse pixels and group them based on common properties. One of the first attempts to segment an object of interest from an image was based on the notion of "snakes" [24]. A "snake" is an energy minimising, deformable spline influenced by constraint and image forces that pull it towards the boundaries of an object of interest. Based on the research on "snakes", Caselles et al. were the first to introduce geodesic active contours [25] as a means to segment an object of interest from an image based on ideas from Euclidean curve shortening evolution. Such a research has led to energy minimisation and level-set segmentation algorithms that can be used for building extraction. In such a framework, energy functions can be constructed that can characterise the properties of most buildings in an image and by minimising such functions, the required buildings can be identified. An energy function can rely not only on texture and colour information, but also on shape information, making the segmentation process more robust, given that the objects to be extracted have known shape. Such methodologies inspired others to find appropriate initialisation contours by developing a circular cast algorithm [26] and using graph theory [4], or even constructing a suitable energy function for building segmentation [27]. One drawback of such methods is that it is hard to construct a suitable energy function that can characterise every building in an image due to the variety in colour and shape buildings display.

Other pixel-based methodologies rely on image segmentation to detect objects of interest. Image segmentation is a way to divide the pixels of an image into meaningful disjoint sets, where the pixels of such a set share some common characteristics. Such segmentation can

be achieved in various ways. In [28], edge information and principal component analysis were employed to segment a multi-spectral image. In [29], a spatial k-means clustering algorithm was developed for multi-spectral image segmentation. Super-pixels, which are considered as the smallest set of pixels that can be merged in an image, were also used for building detection in [30]. A relatively new research field, known as graph theory, was also utilised for image segmentation and building detection. In graph theory, pixels are considered as nodes of a graph and they are connected with their neighbouring pixels with weights that can be equal to the colour difference between the pairs of pixels. A nice introduction on graph theory and a comparison of a specific type of graph minimisation algorithms, known as max-flow algorithms, was presented in [31]. Sirmacek and Unsalan in [32] proposed a way to employ graph theory for the identification of buildings and urban areas. The works of [33] and [34] on graph-cut energy minimisation in the context of building detection are also worth mentioning. A drawback of image segmentation methods is the meticulous and accurate tuning of the parameters for pixel clustering, which makes it hard to construct a framework that can work in every image. Another drawback is the fact that the resulted clusters cannot easily be associated with buildings, as they can have irregular shapes due to occlusions and illumination variations. Non-optimal parameter values, noise, occlusions and illumination variations can lead to over- and under-segmentation problems.

In an attempt to overcome over- and under-segmentation issues, machine learning is introduced in computer vision applications. Currently, state-of-the-art image segmentation has been achieved by employing graphical models. A very successful type of graphical models is Markov Random Fields (MRFs). Singaraju and Vidal proposed the use of pixel neighbourhood differences as local constraints and the extraction and classification of SIFT features as global constraints for improved performance of MRFs in the task of image segmentation [35]. Zheng and Wang proposed an object-based MRF for semantic segmentation of remote sensing images by employing region size and edge information [36]. Initially, a weighted region adjacency graph for capturing complicated interactions among objects is constructed and then the Maximum A Posteriori (MAP) criterion to iteratively renew the joint probability distribution until the method converges is employed. To overcome the problem of current MRF-based methods regarding the insufficient utilisation of the hierarchical semantic information of remote sensing images, Zheng et al. proposed, apart from the standard label field, the use of two auxiliary label fields to assist in semantic segmentation [37]. The Multilevel Logistic Model (MLL) is then employed to describe the interactions within each label field and a conditional probability distribution is developed to model the interactions between label fields. On the other hand, to address the common problem of over-smoothing of MRF-based methods, Golipour et al. proposed the integration of hierarchical segmentation into the MRF spatial prior by employing statistical region merging segmentation [38]. The authors also proposed the extraction of a multi-level "fuzzy no-border/border" map that weighs and modifies the spatial prior of

MRF-based MLL model.

Conditional Random Fields (CRFs) consist another type of graphical models that are employed for image segmentation. In [39], an image is split in super-pixels, which initially acquire coarse labels through multi-scale boosted classifiers and fine labels by CRF label propagation. Xia et al. in [40] proposed a CRF image segmentation and labelling method, where the unary potentials are computed based on various feature classifiers. On the other hand, Zhou et al. proposed a CRF model that combines colour and geodesic distance information into a graph cut optimisation framework [41]. The authors claim that integrating geodesic distance information can assist in overcoming the bias towards shorter paths that occur frequently with graph cut methods and can lead to inadequate segmentation results. A lot of work has been performed on integrating semantic information in the CRF framework. Zhang and Chen in [42], proposed the capturing of long-range colour contrast to better preserve object boundaries, while Payet and Todorovic in [43] combined a CRF model with a Hough forest that captures intrinsic and contextual properties of objects.

A particular type of MRFs and recurrent neural networks is Boltzmann Machines. Restricted Boltzmann Machines were employed in [44] to model global shapes produced by segmentation models. These global (long-range) interactions between regions are then employed in a CRF framework to achieve state-of-the-art image labelling results. Yang et al. proposed Max-Margin Boltzmann Machines (MMBMs) for object segmentation [45]. In addition to the connections from image to labels, MMBMs also include the connections from the image to hidden variables, which allow direct shape inference from image features (SIFT, colour and contour histograms) using MAP inference. Chen et al. in [46] proposed a two-stage image segmentation approach. During the first stage, a Deep Boltzmann Machine [47] is employed to extract the hierarchical global and local structures of the training, while during the second stage, a shape-driven variational framework is built directly on the space of shape probabilistic representation and it is employed for image segmentation.

Building detection and extraction has also been achieved by fuzzy logic, probabilistic theory, artificial neural networks and support vector machines. Fuzzy logic was employed on the spectral and spatial properties of each pixel in [48] or block of pixels in [49]. MRFs were used as an alternative technique to separate buildings from background in a probabilistic framework by Chai et al. [50]. Furthermore, super-pixels were employed in a MRF framework, whose potentials were computed based on shadow adjacency and rectangularity scores [51]. Similarly, Ngo et al. in [52] proposed a MRF building segmentation methodology based on shadow evidence and super-pixel segmentation. Moreover, morphological operations, such as dilation and erosion, applied on pixel neighbourhoods to detect buildings were successfully employed in literature either in the form of morphological building indices [53, 54] or as a hit-or-miss transform [55, 56].

Several pixel-based techniques take advantage of the multi-modal nature of the images to

separate man-made objects from vegetation and ground [6, 57]. Infrared light is absorbed more from vegetation, rather than reflective surfaces, such as buildings and as a result, the near-infrared channel of a multi-spectral image and the Normalised Difference Vegetation Index (NDVI) can greatly assist in the successful differentiation of buildings from vegetation. Shadow detection has also been incorporated in several methodologies, as a way to denote the existence of nearby tall structures, which can be candidate buildings [6, 58, 59, 60].

The second subcategory of building detection algorithms includes model-based or object-based techniques. Such methods find descriptive features in images that can characterise a building. One of these features can be lines. Buildings tend to have shapes that favour line detection. Lines can either be found by Hough Transform [61] or by detecting edges and forming edge chains [62]. A popular and very successful method for edge detection was developed by Canny [63]. Many authors tried to take advantage of lines to form shapes that describe candidate buildings. Two noticeable contributions were made by utilising graph theory for merging lines [64] and by labelling and grouping of corners [5]. A disadvantage of such methods is the fact that lines cannot always describe the underlying structure of an image. Noise and occlusions tend to disrupt straight lines, posing significant challenges to their extraction.

Other object-based building detection methodologies rely on templates. Templates are parametrised shapes that can describe a candidate building. By adjusting the parameters of a template, the extraction of buildings of the same shape but different size is possible. For instance, a rectangular shape can be characterised by two parameters (i.e. width and height), and it can be matched with any rectangular building in an image. 2D deformable templates and roof topology graphs were used as building detection methodologies in [65] and [66] respectively. Moreover, parametrised rectangles were employed in an energy minimisation framework in [67, 68]. A drawback of such techniques is that it is almost impossible to create a template for every possible building shape that can exist in an urban environment, so one has to make certain assumptions and put limitations on the shape of the searched and extracted buildings.

Several object-based building detection methods rely on other distinctive types of features. Sirmacek et al. in [3] employed corner features (Features from Accelerated Test (FAST) and Harris) and texture features in the form of Gabor filtering and Gradient-Magnitude-based Support Regions (GMSR). The authors perceived these features as observations of building presence and estimated their probability density function using a bivariate kernel function. Afterwards, they employed a weighted decision fusion rule to merge the feature distributions in a single final distribution. The modes (i.e. local maxima) of the final distribution were identified and perceived as centres of the candidate buildings. A disadvantage of this approach is that corner features are strongly affect by noise and low contrast of buildings from background. As a result, the presence of noise can inflict significant degradation in the performance of such algorithms. On the other hand, Ilsever

and Unsalan in [2] employed Histogram of Oriented Gradients (HOG) for the identification of building regions. They also employed a shadow detection technique for the removal of false detections and the improvement of the performance of their building detector.

Finally, there are methods that employ discriminatory features for the problem of object detection. Such methodologies can also find application in the problem of building detection as well. Rectangular Haar-like features [69] compute sum of pixel intensities in a parts of a rectangular region. Local Binary Patterns (LBP) [70, 71] features explore the relationship between a central pixel and its neighbours. Eigenfaces [72] and Fisherfaces [73] are techniques that rely on principal component analysis and Fisher's discriminant analysis respectively and they were successfully employed for the problem of human face recognition.

One of the most popular object detection methods is the Deformable Part Model (DPM) [74]. The DPM consists of a root filter (i.e. detection window) that contains coarse information of the object of interest. Then, part filters are employed and placed at a higher resolution to capture finer resolution features that are localised to greater accuracy when compared to the features captured by the root filter, along with their position with respect to the root filter. The score of a hypothesis is given by the scores of each filter at their respective locations (i.e. the data term) minus a deformation cost that depends on the relative position of each part with respect to the root (i.e. the spatial prior). Instead of DPMs, Endres et al. proposed the training of part detectors that can be applied individually to categorise objects and avoid structured models such as the DPM that require joint inference [75]. The part models can either describe global or local information of the objects and once the part detectors are learned, boosting is used to learn a classifier over their collective responses for each region.

On the other hand, Wang et al. proposed regionlets, a radically different way of modelling object deformation compared to existing DPM approaches [76]. A regionlet is a base feature extraction region defined proportionally to a detection window at an arbitrary resolution (i.e. size and aspect ratio). These regionlets are organised in small groups with stable relative positions in order to describe fine-grained spatial layouts inside objects. Their features are aggregated into a one-dimensional feature within one group so as to be flexible to tolerate deformations. The most discriminative regionlets for each object class are selected through a boosting learning procedure. The regionlet model can adapt itself to detect rigid objects and objects with small local or large long-range deformations. Other successful general object detection approaches include energy minimisation on object proposals using the point process density function [77], a sequential model that balances the trade-off between exploration (i.e. improving accuracy) and exploitation (i.e. efficiently terminating search when sufficient evidence about target location has been gathered) using reinforcement learning [78] and a constrained conditional model that formulates a tree structure of the hierarchy of super-pixel merging and associates region merging with likelihoods predicted from an ensemble boundary classifier [79]. Ohn-Bar and Trivedi studied

the modelling limitations of boosted decision trees on the problem of object detection [80]. They found out that although data augmentation improves the accuracy of boosted trees, the trees have limited representation ability and increasing their complexity leads to early saturation problems.

## 2.2 Convolutional neural networks

Convolutional Neural Networks (CNNs) were heavily employed in the 1990s [81, 82] but they were later abandoned, especially for computer vision tasks, when the Support Vector Machine (SVM) classifier was introduced [83]. The interest in CNNs was rekindled in 2012, when Krizhevsky et al. in [11] showed the superior performance of CNNs on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [84]. The new success of the CNNs is based on the large publicly available image datasets, consisting of thousands or millions labelled images and the availability of improved graphical processing units that enable fast training and testing of CNNs.

Since the development of the first CNNs in the 1990s, several modifications have been proposed to increase the classification accuracy and generalisation power of CNNs. A detailed overview of several recent improvements to deep neural networks can be found in [85]. The most important improvements and those that are relevant to the work performed during this thesis are presented in this section. Nair and Hinton in 2010 introduced the Rectified Linear Unit (ReLU) as an alternative to the sigmoid and hyperbolic tangent activation functions [86]. ReLU is a linear non-saturated activation function that prunes the negative part of the input to zero and retains only the positive part. It has been shown that ReLU outperforms other activation functions in deep neural networks and allows a CNN to be trained faster and obtain easier sparse representations [11, 87].

Dropout is a regularisation technique proposed by Hinton et al. in order to prevent overfitting during the training of deep neural networks [88]. Dropout omits the activation of random hidden units with a certain probability, thus forcing them to be independent of other units. Following the success of the Dropout technique, several Dropout variants have been proposed. A fast Dropout training that samples from a Gaussian distribution was introduced in [89], while an adaptive Dropout method, where the probability for each hidden variable is computed by employing a binary belief network that shares parameters with the deep neural network, was proposed by Ba and Frey [90]. Furthermore, Tompson et al. proposed SpatialDropout [91], which extends the Dropout value across the entire feature map. An alternative to Dropout, called maxout, was proposed by Goodfellow et al. in 2013 [12]. The maxout unit computes the maximum across the feature maps, thus avoiding dropping any input.

Batch Normalisation was proposed by Ioffe and Szegedy in [92] as a means to achieve faster and more accurate training in deep neural networks. Batch Normalisation fixes the mean

and variances of layer inputs after each mini-batch is introduced in the network. It has been shown that Batch Normalisation reduces internal covariant shift and the dependence of the gradients on the scale of the parameters or of their initial values, thus providing a beneficial effect on the gradient flow through the network [92].

Other attempts to improve the performance of CNNs led to the use of smaller receptive window size and smaller stride in the first convolutional layer [93, 94]. Such approaches achieved best performances on the ILSVRC 2013. In [95], Howard investigated data augmentation and prediction transformation techniques to improve the performance and generalisation ability of deep neural networks. Simonyan and Zisserman in [96] dealt with another aspect of the CNN architectures, namely the depth. They increased the depth of the network by adding convolutional layers with very small $(3 \times 3)$ convolution filters, while they kept the other parameters of the network fixed. They demonstrated that an increase in depth can lead to an increase in the performance of a deep neural network.

To enhance model discriminability and avoid over-fitting, Lin et al. proposed the Network in Network (NIN), which concerns the use of Multi-Layer Perceptrons (MLPs) inside the deep neural network [13]. A MLP can be considered as a general non-linear function that maps the input to the output feature vector. Their work led Szegedy et al. to propose the Inception module [97], which uses variable filter sizes to capture patterns of different size. This allows for an increase in depth and width of a CNN without increasing the computational complexity. The Inception module was employed in GoogleNet, a 22-layer deep neural network that achieved superior performance on the ILSVRC 2014.

Many researchers combined deep neural networks with other classification algorithms. Girschik et al. in [98] combined region proposals with features extracted by a deep neural network. The features were then classified using a linear SVM classifier. With this approach, they achieved excellent performance on Visual Object Classes (VOC)$2010 - 2012$ dataset [99]. Zhining and Yunming proposed the optimisation of CNN parameters using a genetic algorithm [100]. They considered a CNN as an individual, whose hyper-parameters (i.e. filter sizes, number of feature maps, etc.) can be optimised using a genetic algorithm. CNNs were also employed as general patch-based feature extractors for image recognition and classification [101, 102].

Finally, He et al. proposed residual learning to address the problem of degradation in deep neural networks [14]. Residual learning concerns the use of short-cut connections that pass the input to deeper layers unchanged, thus bypassing a few layers of the deep neural network. They claim that the residual mapping can be better optimised than the original mapping and therefore, the problem of the accuracy saturation of a deep neural network can be tackled.

The powerful modelling abilities of CNNs have led to their employment for the problem of general objection detection and image segmentation. More specifically, high-capacity CNNs were employed in [103] for feature extraction from region proposals. Class-specific

linear SVMs were then trained on the extracted features to localise and segment objects. GoogleNet [97], pre-trained on ImageNet and fine-tuned on aerial images, was employed for image classification and object detection in aerial images in [104]. A deep hierarchical network, named HyperNet, was proposed in [105] for jointly handling region proposal generation and object detection. HyperNet is based on an elaborately designed Hyper Feature, which firstly aggregates hierarchical feature maps and then compresses them into a uniform space. A region proposal generation network is then constructed to produce region proposals with scores and bounding boxes. These proposals are finally classified and adjusted based on the detection module.

On the other hand, Li et al. in [106] proposed the Attention to Context CNN (AC-CNN) that effectively incorporates global and local contextual information into the region-based R-CNN detection framework. The AC-CNN consists of one Attention-based Global Contextualised (AGC) sub-network and one Multi-scale Local Contextualised (MLC) sub-network. To capture global context, the AGC sub-network recurrently generates an attention map for an input image to highlight useful global contextual locations through multiple stacked long short-term memory layers. To capture surrounding local context, the MLC sub-network exploits both the inside and outside contextual information of each specific proposal at multiple scales. The global and local contexts are then fused together to make the final decision for detection. In [107], an object segmentation algorithm that integrates both global shape and local edge information in a deep learning framework is proposed. The convolutional layers extract image features, while a fully connected section represents shapes specific to a given object class. Finally, Ren et al. introduced the Region Proposal Network (RPN) that enables nearly cost-free region proposals [108]. A RPN is a fully convolutional network that simultaneously predicts object bounds and objectness scores at each position. The RPN is trained end-to-end to generate high-quality region proposals, which are used by the Fast R-CNN detector [109] for object detection. The authors also merged RPN and Fast R-CNN into a single network by sharing their convolutional features and proposed Faster R-CNN, where the RPN component tells the unified network where to look.

As far as image segmentation is concerned, Zhou et al. proposed the use of a CNN to automatically learn features and create local predictions and global structure consistency by combining fine and coarse layers [110]. Afterwards, a CRF model utilises the extracted semantic information for image segmentation. In [111], the concept of Deep Randomly Connected Conditional Random Fields (DRCRF) was introduced. The DRCRF fuses fully connected random fields and deep random fields together to obtain benefits from long-range interactions among image objects and attempts to address the problem of smoothed segmentation boundaries, from which most MRF-based methods suffer. Lin et al. in [112] proposed the capturing of semantic correlations between neighbouring image patches using CNN-based pairwise potential functions and the modelling of the patch-background context by employing a multi-scale CNN and sliding pyramid pooling. Finally, Cao et

al. proposed the augmentation of RGB monocular images with depth information for accurate object detection and semantic segmentation [113]. Depth estimation is achieved from a deep convolutional neural fields model that jointly learns the unary and pairwise potentials of CRF in a single CNN framework [114]. Since an RGB image has two labels after depth estimation: one semantic label and one depth label, a CNN with a multi-task loss (i.e. semantic label prediction and depth value regression) is proposed.

## 2.3 Change detection

Change detection in computer vision can generally be defined as the problem of identifying the type and magnitude of changes between two compared images that depict the same scene and have been acquired at different temporal periods. According to the goals of each task, some changes are important and they need to be further analysed and modelled and others are unimportant or irrelevant and they should be discarded. In the framework of this thesis, the main interest lies in the building change detection. As a result, changes that refer to differences in size and shape of buildings are considered significant, while changes in other urban objects, such as roads, ground and vegetation, are considered insignificant.

Change detection can be a challenging task as it is affected by various elements, such as spatial, spectral and temporal constraints, radiometric resolution and atmospheric conditions [115]. Several change detection techniques have been developed in an attempt to overcome the difficulties and provide accurate change detection results. A comprehensive review of change detection methods is made in [116], where the authors grouped the change detection techniques in 7 classes with respect to the application area. Other authors grouped change detection techniques based on different viewpoints and categorised them in pre-classification or post-classification methods [117] and pixel-based or object-based methods [118].

Pixel-based change detection methods consider the image pixel as the basic unit of image analysis and change. These techniques, which can also be considered as pre-classification techniques due to the fact that they are applied on raw data, are employed most often in the literature of change detection methods. Several pixel-based change detection methods have been developed and their advantages and disadvantages have been analysed in an attempt to improve the change detection accuracy [119, 120, 121]. Known pixel-based change detection techniques are: image differencing, image ratioing, image regression, background subtraction, Principal Component Analysis (PCA) and Change Vector Analysis (CVA).

Image differencing computes the absolute differences of the grayscale pixel values between the compared images and it is the easiest to implement but no information on the type of change is produced. An example of employing image differencing for land cover change detection was presented in [122]. Image ratioing computes the ratio of the grayscale pixel values and it has the advantage that it can better handle calibration errors [123] but no

complete matrices of change information is provided. Image regression attempts to fit the pixel values of the target image to those of the reference image using regression analysis. It accounts for differences in the mean and variance between pixel values for different dates and thus it reduces the adverse effects caused by atmospheric conditions and sun angles [119]. Unfortunately, it is difficult to define accurate regression functions due to local illumination variations present in images.

Background subtraction [121] attempts to retrieve the background information from the compared images and then subtracts it from the images in order to acquire change information. Although easy to implement, background subtraction suffers from low accuracy [116]. PCA employs dimensionality reduction to remove data redundancy between multispectral bands and emphasise different information. PCA was employed in [124] as a distance metric to identify the magnitude of change in the value of a pixel and then Bayes rule was used to separate pixels in two classes, "change" and "no-change". The disadvantages of PCA change detection methods are the difficulty in the interpretation and labelling of the PCA results and the lack of extracted information about the type of changes [116].

Contrary to the previously presented pixel-based change detection techniques, CVA can provide information about the type of changes. CVA employs all the available multispectral information of an image, considers each pixel as a vector and differencing is performed on vectors. As a result, the Euclidean distance of the vectors describes the magnitude of changes, while the angle of the vectors provides information on the type of changes. A disadvantage of the CVA change detection method is the fact that it requires an accurate radiometric correction in order for atmospheric effects to be alleviated and such an accurate radiometric correction is not feasible [125].

All pixel-based change detection methods suffer from two significant disadvantages or limitations that led research towards object-based change detection techniques. The first disadvantage is that a threshold is required to differentiate between pixels that belong to the "change" class from those that belong to the "no-change" class. The selection of an optimal threshold is a difficult procedure and usually pixel-based techniques suffer from mis-detections or over-detections [115, 116]. This is because too low a threshold includes too many areas of change, while too high a threshold excludes areas of change. The second limitation of pixel-based techniques is the fact that they do not take into account the spatial relations between neighbouring pixels [126]. This exclusion of spatial context leads to the removal of vital information that would greatly assist towards accurate change detection results.

A special subcategory of pixel-based change detection techniques are the texture analysis based methods. These methods operate at the window/block level, meaning that statistical features are computed over overlapping windows that move all over the compared images. As a result, these techniques take into consideration the spatial context of pixels and they

can be more robust to noise than other pixel-based techniques. Change detection results in these algorithms are based on the evaluation of similarity metrics between the compared windows. A common similarity metric that is employed in several texture analysis based change detection methodologies is correlation that can be employed either in the time domain in the form of Normalised Cross-Correlation (NCC) [127] or in the frequency domain by transforming images using Fourier and then analysing the power spectrum (Phase Correlation (PC)) [128]. Furthermore, the Bhattacharyya distance is employed for the comparison of the pixels inside the compared overlapping windows [120, 129]. Texture analysis based methods suffer from the same limitation as other pixel-based techniques, which is the need for an optimal threshold that differentiates the pixels that correspond to true changes from all other pixels.

In an attempt to overcome the limitations of pixel-based change detection techniques, object-based methods were developed. Object-based methods can usually be described as post-classification methods due to the fact that the images are usually segmented to regions or classes before any change detection technique is applied. The unit for analysis in object-based change detection techniques is an image object with rich information, such as texture, shape and spatial relationships with neighbouring objects, allowing the exploitation of the spatial context [130, 131]. Image segmentation is at the core of object-based change detection methods and various supervised or unsupervised segmentation techniques are employed with varying results. Most object-based change detection techniques can be grouped into either boundary or edge-based (discontinuity of pixels) or area-based (similarity of pixels) methods [132].

Image segmentation algorithms that are usually employed in object-based change detection methodologies are the Expectation-Maximisation (EM) algorithm [133], Artificial Neural Networks (ANNs) [134] and SVM classifiers [135]. Change detection methods based on the EM algorithm achieve higher accuracy than other methods but they require a priory knowledge of the joint class probability [116]. The ANN approach can provide good change detection results, especially when classes are not normally distributed, but it needs a lot of training time and it is sensitive to the amount of training data [116]. Finally, the SVM classifiers have the advantage of making no assumptions on the underlying data distribution and treat the change detection task as a binary classification problem. The SVM classifier learns from the training data and an optimal threshold is automatically defined [136] from the spectral features in order to classify pixels to the "change" and "no-change" classes. Other machine learning algorithms that are employed for classification and change detection monitoring include genetic programming [137], random forest [138, 139] and fuzzy logic [140].

To address the need for optimal parametrisation in image segmentation and improve change detection results, graphical models were proposed. Graphical models, such as MRF and CRF, employ low-level features and neighbourhood potentials to segment images and identify changes. Zhong et al. in [141] employed colour and corner information,

while Hoberg et al. in [142] employed structure features (HOG), texture features (correlation, contrast, entropy, etc.) and colour information in a change detection framework based on MRFs. Sziranyi and Shadaydeh in [143] proposed the use of colour and texture information from several temporal images in an unsupervised or partly supervised multi-layer MRF framework. The MRF segmentation was performed on fused layer data, resulting in multi-layer labelling. The multi-layer labelling was then used as a preliminary training map for each image layer.

On the other hand, Wei et al. in [144] recognised the problem of defining an optimal threshold for the differentiation between the "change" and "no-change" classes and they proposed the use of a MRF model to fuse the information from several different change detection algorithms and accurately identify true changes. Yousif and Ban in [145] proposed a change detection algorithm based on a MRF model capable of dealing with the effect of noise in the data by employing non-local context information. More specifically, the authors assumed that pixels with more similar local structures have a larger probability to belong to the same class. They then modelled this probability into the MRF framework in order to preserve spatial details under the effect of noise. Similarly, non-local information was employed in [146], where a non-local means filter that captures the redundancy property of the image texture was employed to improve change detection results.

Finally, Zhou et al. in [147] proposed a CRF-based change detection approach, where fuzzy c-means clustering was employed in order to compute the CRF unary potentials. The authors also incorporated super-pixel segmentation information in the CRF framework in order to preserve edge boundaries and address the common problem of over-smoothing that methodologies based on CRFs face.

Object-based techniques that rely on image segmentation suffer from under- and over-segmentation errors. Such errors create objects that do not accurately represent real-world objects [148]. Under-segmentation creates bigger objects that cover more than one real-world objects and they may also contain mixed classes of objects, whereas over-segmentation creates smaller parts of real-world objects that need to be merged to create a more realistic representation of actual objects [149]. Unfortunately, the errors from the employed image classification procedure are propagated in the final change map, reducing the accuracy of the final change detection results [150]. In order to improve change detection results, the classification of individual images has to be as accurate as possible. An attempt to avoid under- and over-segmentation issues was performed by Doxani et al. in [151], where scale-space filtering was employed to accurately detect objects in multiple scales. Afterwards, the multivariate alteration detection algorithm was applied to identify possible changes, which were further analysed using a knowledge-based classification scheme.

In [116], Lu et al. identified several other important categories of change detection techniques. There are techniques that rely on advanced spectral mixture and reflectance

models and there are also methodologies that take advantage of Geospatial information to derive knowledge about changes in an urban environment. However, such techniques are beyond the scope of this thesis and they are not analysed. The interested reader can find a comprehensive review of such methodologies in [116].

## 2.4 Image pre-processing

Two important pre-processing steps that are usually employed when dealing with the tasks of object and change detection in multi-spectral images are discussed in this section. The first pre-processing step is called multi-spectral band sharpening and it is concerned with the enhancement of the details present in multi-spectral images. The second pre-processing step is called image temporal registration and is concerned with the accurate alignment of images depicting the same scenery but taken in different time frames.

### 2.4.1 Multi-spectral band sharpening

An important pre-processing step that can be applied to multi-spectral images in order to improve the accuracy of a building detector is multi-spectral band sharpening. This is because a satellite can provide panchromatic images of much higher spatial resolution than the resolution of multi-spectral images. For instance, QuickBird and WorldView-2 satellites provide panchromatic images of spatial resolution $4\times$ larger than the resolution of the multi-spectral images. To take advantage of the higher spatial resolution of the panchromatic images, a procedure, known as multi-spectral band sharpening is usually employed. The goal is to fuse the multi-spectral and the panchromatic images in order to create a multi-spectral image with the same spatial resolution as the panchromatic image. Such a sharpened high-resolution multi-spectral image can significantly enhance the accuracy and robustness of a building detection algorithm. Vrabel tested various sharpening algorithms to find out that the CN algorithm performs the best [152]. According to the CN algorithm [153], if $MS_i$ is the $i^{th}$ low-resolution multi-spectral band and $PAN$ is the panchromatic band, then the following equation produces the $i^{th}$ high-resolution colour normalised multi-spectral band $CN_i$

$$CN_i = \frac{(MS_i + 1.0) * (PAN + 1.0) * 3.0}{\sum_i MS_i + 3.0} - 1.0 \tag{2.1}$$

The values of $MS_i$ and $PAN$ are considered to be in the range $[0, 2^{n-1}]$, where $n$ is the radiometric resolution of the satellite image, measured in bits. The small additive constants in Eq. (2.1) are included to avoid division by zero, while the last negative term $(-1.0)$ is employed so that a black pixel of the initial image corresponds to exactly the same black pixel of the sharpened image. The result of applying the CN algorithm to a

multi-spectral image is shown in Fig. 2.1. The visual superiority of the sharpened multi-spectral image, with respect to the multi-spectral image before the sharpening procedure can be easily observed. The resolution of the sharpened multi-spectral image is enhanced and the details are more visually distinguishable. Any image processing technique can then be applied to the sharpened high-resolution multi-spectral image with improved results.

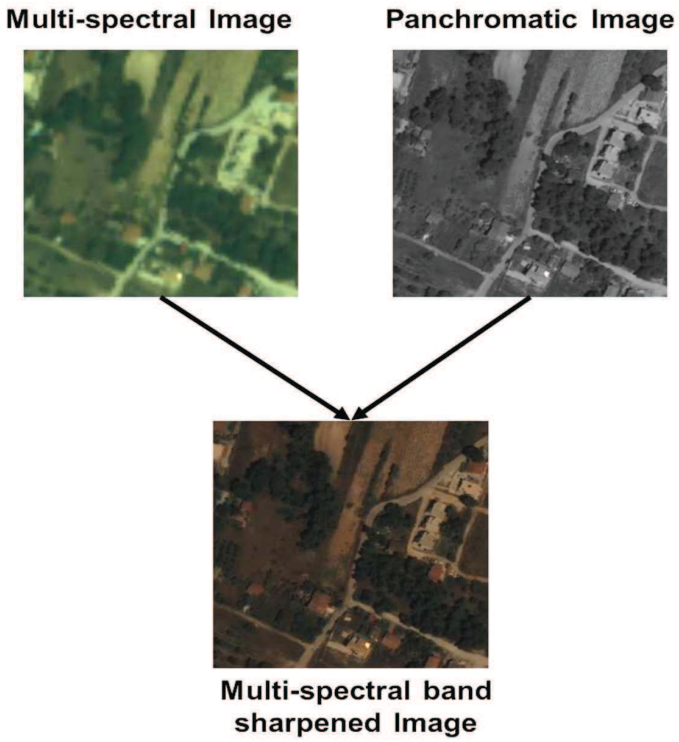



Figure 2.1: Multi-spectral and panchromatic image fusion.

### 2.4.2 Image temporal registration

Image temporal registration is concerned with the task of aligning two images depicting the same urban environment and captured at different time frames in a way that allows the accurate matching of the objects present in the images. This means that the registered images should have the same objects appear in the same image locations. Image temporal registration is differentiated from other image registration methods that perform template, viewpoint or multi-modal registration as the other techniques are not concerned with the factor of time. Image registration can be a really challenging task as the compared images may present strong illumination variations, noise, geometric distortions and spectral and spatial resolution differences. Moreover, in multi-spectral satellite images, atmospheric conditions and cloud interactions can cause significant problems in the image registration procedure.

According to [154], there are several classes of image registration techniques, but the most important of which are the area-based and the feature-based methods. The area-based or

similarity-based methods compare image regions by employing a sliding window over the images and identifying how similar the corresponding areas between the pair of images are. The comparison is made based on the values of the corresponding pixels inside the sliding window. Known area-based image registration techniques rely on mutual information [155] and correlation either in the time domain (NCC) [156] or in the frequency domain (PC) [157].

Feature-based image registration methods are considered illumination invariant as they extract features in the form of corners, edges and lines and image registration is performed based on this information. A few representative samples of the work done on feature-based image registration methods are the employment of the Radon transform to detect and register straight lines [158], the extraction of corners and their matching using a rotation invariant block matching method [159] and the extraction and matching of Scale-invariant Feature Transform (SIFT) [160] features across the compared images [161].

In the framework of this thesis, no image registration techniques are required as the images of the datasets that are employed for the comparison of the tested change detection methodologies are already registered. However, in the case of unregistered images, the use of an image registration technique is necessary before the employment of any change detection method in order to avoid the detection of false changed areas that are caused by image displacement.

## 2.5 Datasets

In this section, we present the datasets that are employed during the work of this thesis in order to evaluate the performance of the developed algorithms and compare them with other state-of-the-art methodologies.

### 2.5.1 Athens dataset

The Athens dataset consists of a large QuickBird satellite image with a size of $6201 \times 4596$ pixels and 29 smaller images of an average size of $1000 \times 1000$ pixels. All images depict areas of the suburban part of Athens, Greece. The 29 small images depict 6 different areas, each acquired on 5 different temporal periods and more specifically in years 2006, 2007, 2009, 2010 and 2011. The images of the temporal period $2006 - 2009$ are captured by the QuickBird satellite, while the images of the temporal period $2010 - 2011$ are obtained from the WorldView-2 satellite. These 29 images are orthorectified, meaning that distortions caused from the sensor and the earth's terrain have been geometrically removed. As a result, the orthorectified images allow for a more direct and accurate measurement of angles and distances. Furthermore, the images that depict the same urban area in different timestamps are registered, so that buildings appearing in one image lie in exactly the same

position as in the other images. Finally, the 29 multi-spectral images are sharpened in order to have the highest possible spatial resolution that allows a more detailed view of the depicted urban scenes. The 29 satellite images are also accompanied with the ground truth data of the buildings present on them. As a result, these images can be used both for the task of building detection and segmentation and for the task of change detection by simply subtracting the ground truth building masks in order to acquire the ground truth change detection masks.

The QuickBird satellite, launched by DigitalGlobe at 2001, provides multi-spectral images with spatial resolution of $2.4m$ and panchromatic images with spatial resolution of $0.6m$. Furthermore, a QuickBird multi-spectral image consists of 4 channels, namely, red, green, blue and Near-infrared (NIR). The radiometric resolution of the QuickBird satellite is 11 bits, meaning that the intensity values for the pixels of a QuickBird satellite image are in the range $[0, 2^{11} - 1]$ or $[0, 2047]$.

The WorldView-2 satellite was launched by DigitalGlobe at 2009 and it provides multi-spectral images with spatial resolution of $2m$ and panchromatic images with spatial resolution of $0.5m$. Furthermore, a WorldView-2 multi-spectral image consists of 8 channels; the 4 standard channels, namely, red, green, blue and near-infrared (NIR1) and 4 new channels, namely red-edge, coastal, yellow and near-infrared 2 (NIR2). The radiometric resolution of the WorldView-2 satellite is 11 bits, meaning that the intensity values for the pixels of a WorldView-2 satellite image lie in the range $[0, 2047]$.

Fig. 2.2 presents the spectral resolution and the band configuration of the QuickBird and WorldView-2 satellites, whose images are provided as part of the Athens dataset. The channels in the satellite images are ordered in the way they are shown in Fig. 2.2, meaning that for WorldView-2 images, channel #1 corresponds to the coastal band, channel #2 corresponds to the blue band, etc. From Fig. 2.2, it can be observed that even the same bands for the QuickBird and WorldView-2 images vary in the wavelength interval they represent, meaning that the same surface area of the earth may be captured slightly different when observed from these two satellite sensors.

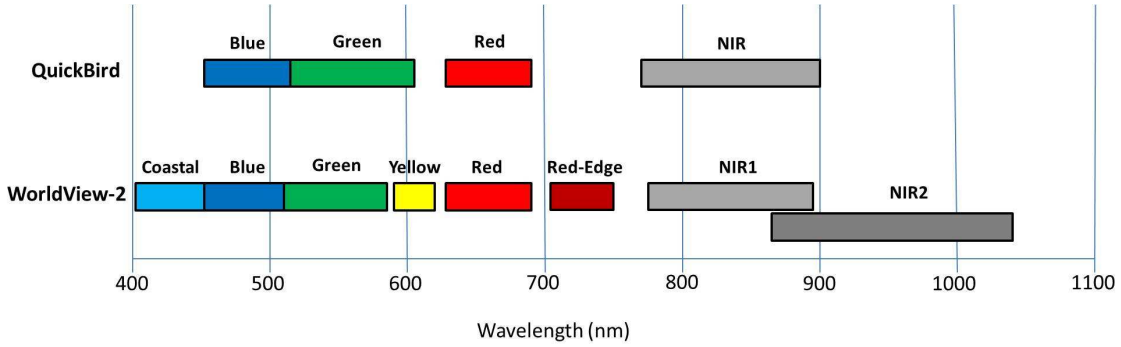



Figure 2.2: Spectral resolution and band configuration for QuickBird and WorldView-2 satellites. Information taken from [1].

The training set of the Athens dataset consists of 700 positive and 1000 negative samples that are cropped from the large QuickBird satellite image and are manually annotated. There are two validation sets defined in the Athens dataset. The first validation set that is denoted as "sample" validation set consists of 3000 positive and 6000 negative image patches that are extracted arbitrarily from the 29 small satellite images of the Athens dataset based on the overlap of the image patches with the ground truth mask. The second validation set is referred as "image" validation set and consists of 5 small satellite images (3 QuickBird and 2 WorldView-2 images) out of the 29 satellite images. These images are referred as area_2b_2007, area_2c_2007, area_1a_2009, area_1a_2010 and area_1b_2011 in Table 2.1. The test set of the Athens dataset consists of the 29 small satellite images, where 5999 buildings are depicted. In the case where the "image" validation set is employed, the Athens test set is restricted to the 24 remaining small satellite images. A few positive and negative training samples from the Athens dataset are presented in Fig. 2.3. The Athens test set, with the QuickBird and WorldView-2 images clarified, is presented in Fig. 2.1.



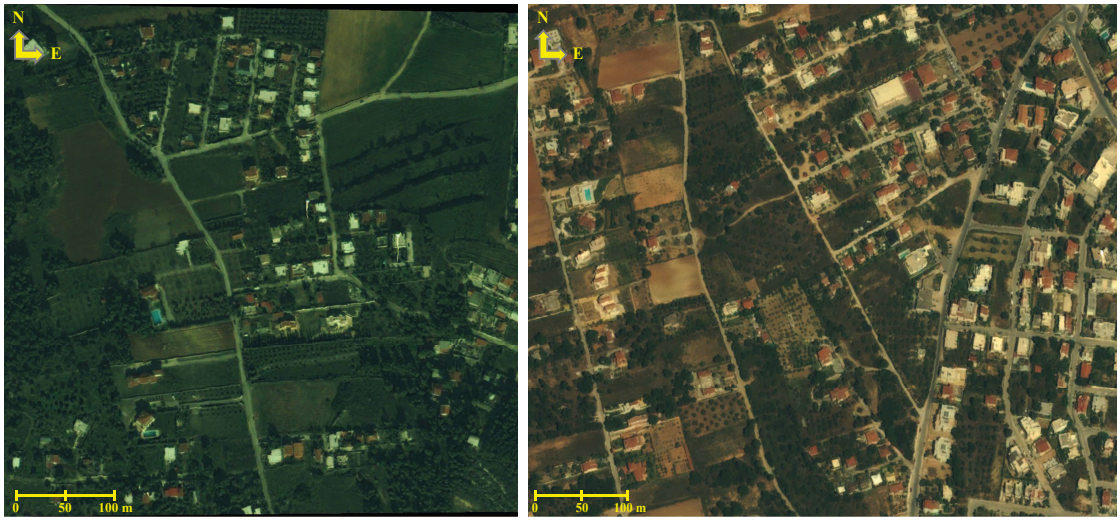
Figure 2.3: Positive and negative training images from the Athens dataset are shown in first and second row respectively.

Table 2.1: Test set of the Athens dataset. QuickBird images are shown as QB, while WorldView-2 images are shown as WV2. The images of the Athens "image" validation set are presented in bold.

| Area \ Year | 2006 | 2007 | 2009 | 2010 | 2011 |
|---|---|---|---|---|---|
| 1a | area_1a_2006 (QB) | area_1a_2007 (QB) | **area_1a_2009 (QB)** | **area_1a_2010 (WV2)** | area_1a_2011 (WV2) |
| 1b | area_1b_2006 (QB) | area_1b_2007 (QB) | - | **area_1b_2010 (WV2)** | area_1b_2011 (WV2) |
| 1c | area_1c_2006 (QB) | area_1c_2007 (QB) | area_1c_2009 (QB) | area_1c_2010 (WV2) | area_1c_2011 (WV2) |
| 2a | area_2a_2006 (QB) | area_2a_2007 (QB) | area_2a_2009 (QB) | area_2a_2010 (WV2) | area_2a_2011 (WV2) |
| 2b | area_2b_2006 (QB) | **area_2b_2007 (QB)** | area_2b_2009 (QB) | area_2b_2010 (WV2) | area_2b_2011 (WV2) |
| 2c | area_2c_2006 (QB) | **area_2c_2007 (QB)** | area_2c_2009 (QB) | area_2c_2010 (WV2) | area_2c_2011 (WV2) |

Finally, two images of the Athens test set are presented in Fig. 2.4. The training and test sets of the Athens dataset reveal the difficulties in detecting buildings in an urban environment. Buildings appear in various sizes and shapes and present colour and texture variations. Furthermore, illumination variations, spectral differences between the QuickBird and WorldView-2 satellite images and low contrast of buildings from their surroundings pose significant problems in the development of a general building detection methodology

that can accurately and robustly detect buildings in every image of the Athens test set.




(a) (b)

Figure 2.4: (a) QuickBird test image area_1a_2007 and (b) WorldView-2 test image area_2a_2010 of the Athens dataset.

As it has already been pointed out in the introduction of this thesis, the Athens dataset consists of high-resolution satellite images and thus shadows are visible. However, the resolution of the images and the sun angles the time these images were captured make shadows to appear small in size comparable to the buildings we want to detect as one can deduce from Fig. 2.4. Since shadows cannot be easily mistaken with buildings and due to the fact that the scene properties of an image are usually not known for successful shadow extraction, we decide not to employ any shadow removal algorithm. However, we rely on HOG features and a proposed building region scoring algorithm to suppress possible shadow regions that can be classified as buildings. This shadow suppression effect is achieved by employing the gradient information and the region size and local saliency.

As far as the change detection task is concerned, the images of areas 1b, 2a and 2c are used for the validation of the various parameters that the tested change detection algorithms depend on and they consist the Athens change validation set. The performance of the change detection methods is evaluated in the images of areas 1a, 1c and 2b, which consist the Athens change test set.

### 2.5.2 CIFAR-10 dataset

The CIFAR-10 dataset [162] is a well-known benchmark dataset that is used for multi-class image classification tasks. The dataset consists of 60000 colour images of small size (i.e. $32 \times 32$ pixels) that are equally distributed between 10 classes. These classes are "Airplane", "Automobile", "Bird", "Cat", "Deer", "Dog", "Frog", "Horse", "Ship" and "Truck". The CIFAR-10 dataset is split in a training set of 50000 images and a test set

of 10000 images. The images of both the training and the test set are equally distributed between the 10 classes. The validation set is formed by randomly selecting 10000 images out of the CIFAR-10 training set. A few images of the CIFAR-10 dataset are presented in Fig. 2.5.

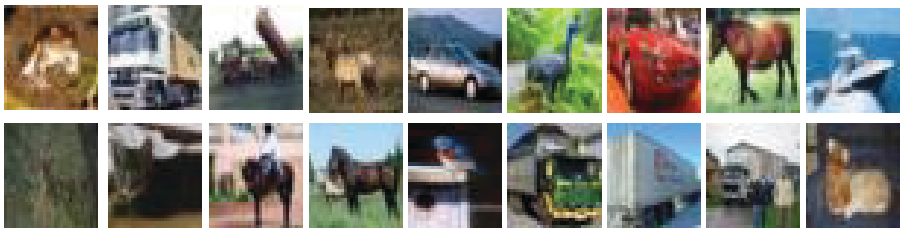


Figure 2.5: Images of the CIFAR-10 dataset.

The CIFAR-10 dataset is employed as a general object classification dataset to further support the novel contributions of the developed CNN-based building detection methodology in a different object classification task.

### 2.5.3 MNIST dataset

The MNIST dataset [163] is a well-adopted by the computer vision community and publicly available benchmark dataset that is employed for the classification of handwritten digits. The MNIST dataset consists of 70000 small-sized (i.e. $28 \times 28$ pixels) images, split in a training set of 60000 images and a test set of 10000 images. The images of the dataset are grayscale, size-normalised, centred and depict digits in the range $0 - 9$. The images of the dataset are unequally distributed between the 10 classes. The validation set is formed by randomly selecting 10000 images out of the MNIST training set. A few images of the MNIST dataset are depicted in Fig. 2.6.

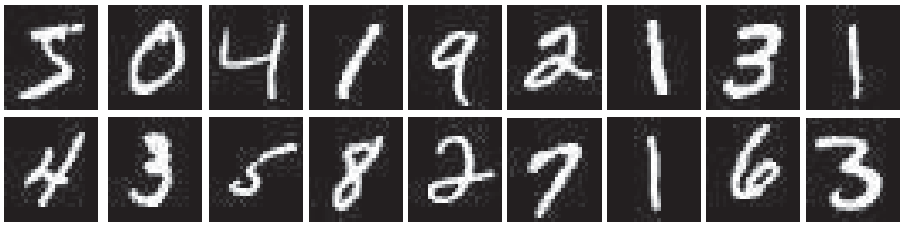


Figure 2.6: Images of the MNIST dataset.

The MNIST dataset is employed, similar to the CIFAR-10 dataset, as another general object classification dataset in order to demonstrate the advantages of employing the novel contributions of the developed CNN-based building detection methodology in a CNN-based algorithm that is concerned with the classification of handwritten digits.

### 2.5.4 AICD dataset

The Aerial Imagery Change Detection (AICD) dataset was created and presented by Bourdis et al. in [164]. The AICD dataset is hosted in the site Computer Vision Online and it was created using the realistic rendering engine of the serious game Virtual Battle Station 2, developed by Bohemia Interactive Simulations. The AICD dataset consists of 100 different scenes of size $800 \times 600$ pixels containing several objects (trees, buildings, etc) and moderate ground relief. Each scene is rendered under 5 different viewpoint angles with a difference of $10^{\circ}$ in order to present the scenes under different illumination variations and with or without shadows. As a result, the total number of images of the dataset is 1000. Each scene is also accompanied with a ground truth mask, depicting the differences between a reference and a test image. There is only a single important change between the reference and the test images and this change is not restricted only to buildings, while shadows do not count as changes.

In this thesis, we utilise the 100 different scenes without shadows and for all viewpoint angles in order to test the proposed change detection methods under different illumination variations. The images that contain shadows are not considered as we do not employ any shadow detection and removal technique to suppress the effect of shadows in the change detection results. As a result, the total number of tested images is equal to 500. Two images of the AICD dataset are depicted in Fig. 2.7.



(a) (b)

Figure 2.7: Images of the AICD dataset.

The AICD dataset is employed as another change detection dataset in order to further demonstrate the effectiveness of the proposed novel contributions of this thesis in the change detection task.

## 2.6 Performance evaluation metrics

To evaluate the performance of the proposed building and change detection algorithms and to compare them with other methodologies, a set of metrics is employed. Before a thorough analysis of the metrics is performed, some basic terminology should be defined. In a building detection framework and at the object level, true positive (TP) refers to the number of buildings that are correctly identified, false positive (FP) refers to the number of false detections and false negative (FN) refers to the number of buildings that are not detected. A building is considered detected in our object-based building detection methodologies if at least 50% of the building's area is included in a detected rectangular area that a building detection methodology outputs. At the pixel level, TP refers to the number of pixels that are correctly identified to belong to the "building" class and FP refers to the pixels that are wrongly classified as belonging to the "building" class. Furthermore, true negative (TN) refers to the number of pixels that are correctly classified as belonging to the "non-building" class and FN refers to the number of "building" pixels that are not detected.

In the change detection framework and at the pixel level, TP refers to the number of pixels that are correctly identified as changes and FP refers to the number of pixels that are wrongly classified as changes. Finally, TN refers to the number of pixels that are correctly classified as presenting no changes and FN refers to the number of pixels that are wrongly recognised as presenting no changes.

Given the previous definitions, the employed metrics for the building and change detection algorithms are:

- **Recall:** Recall or sensitivity or true positive rate refers to the number of true positives divided by the number of positives and it quantifies the avoiding of the false negatives. Recall takes values in the range $[0, 1]$ and higher values correspond to a better classifier. Recall is defined as follows

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2.2}$$

- **Precision:** Precision or positive predictive value refers to the number of true positives divided by the number of detections and it quantifies the avoiding of false detections. Precision takes values in the range $[0, 1]$ and higher values correspond to a better classifier. Precision is defined as follows

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2.3}$$

- **$F_1$-score:** Recall and precision present opposing behaviours, meaning that usually when recall increases, precision drops and vice versa. $F_1$-score is defined as the har-

monic mean of recall and precision and it can be considered to be a more descriptive metric of the performance of an algorithm due to the fact that it takes into consideration both recall and precision. $F_1$-score takes values in the range $[0, 1]$ and higher values correspond to a better classifier.

$$F_1\text{-score} = \frac{2 \cdot \text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}} \tag{2.4}$$

- **Kappa index:** Kappa index or Cohen's kappa coefficient [165] is a statistic, which measures inter-rater agreement for categorical results. It is generally thought to be a more robust measure than simple percent agreement calculation, since Kappa index takes into account the possibility of the agreement occurring by chance. The kappa index takes values in the range $[0, 1]$ and higher values correspond to a better classifier. Given the observed relative agreement between a classifier and the ground truth $p_o$ and the hypothetical probability of chance agreement $p_e$ between a classifier and the ground truth, the Kappa index is defined as

$$\text{Kappa} = \frac{p_o - p_e}{1 - p_e} \tag{2.5}$$

where the relative agreement $p_o$ and the hypothetical probability of chance agreement $p_e$ between a classifier and the ground truth are given by

$$p_o = \frac{TP + TN}{TP + FP + TN + FN} \tag{2.6}$$

$$p_e = \frac{(TP + FP)(TP + FN) + (TN + FP)(TN + FN)}{(TP + FP + TN + FN)^2} \tag{2.7}$$

## 2.7 Critical summary

This section summarises the key findings of the literature review that was presented in this chapter and establishes links between these findings and the contributions of this thesis.

### 2.7.1 Building detection

The problem of building detection and object detection in general is commonly addressed by methodologies that utilise discriminative hand-crafted features that can differentiate between the objects of interest and the background. In the building detection task, the use of vegetation indices, shadow and height information, edges, lines, corners and texture descriptors is very significant and it has been shown to improve the accuracy of building detection methodologies. However, noise, occlusions and low contrast of buildings with the

background can severely affect the extraction of such features, leading building detection algorithms to produce inconsistent results across different datasets. Moreover, shadow-based building detection methodologies suffer from cases of undetected buildings due to missed or no visible shadows and cases of false detections due to non-building structures casting shadows.

More sophisticated and robust solutions were also presented, such as 2D deformable templates, roof topology graphs and level-set algorithms, but such features had trouble describing the huge variety of building shapes, sizes, texture variations and colours present in an image. The need for addressing the problem of noise and occlusions and proposing more powerful object detectors has led to significant advances in the field of machine learning and a huge shift of the interest of researchers towards algorithms that can build probabilistic models (i.e. graph-cuts, MRFs, CRFs, etc.), which link features with the presence or absence of the objects of interest and deep learning methodologies that automatically construct descriptive features for the problem at hand (i.e. CNNs and their variants).

Although CNNs have led to state-of-the-art performances on the majority of object detection problems, there is still a lot of research on methods that utilise machine learning techniques on hand-crafted features. This is mainly due to the fact that hand-crafted features have a logical interpretation (e.g. shadows are known to exist close to tall structures) and the object detection results can give useful feedback on what measures can be taken for performance improvement. On the other hand, the automatic feature extraction ability of CNNs makes trial-and-error the only means to identify whether and why a CNN may perform poorly.

The building detection method that we propose in this thesis can be considered as a combination of hand-crafted and automatically generated features. We propose the use of HOG and LBP descriptors that are considered very discriminative gradient and texture features respectively and are quite robust to illumination variations, noise and occlusions. Moreover, we refine the building detection results by employing CNNs. In this way, we attempt to improve the performance of hand-crafted features without losing the intuition behind the selection of these features for the building detection task.

### 2.7.2 Change detection

As far as change detection is concerned, the previous literature work is concentrated on three main change detection approaches: pixel-based, region-based and object-based. Pixel-based methods examine the magnitude of change per pixel between two compared images without taking into consideration the pixel neighbourhoods. Pixel-based algorithms are fast and easy to implement but suffer from the effect of noise, shadows and illumination variations. On the other hand, region-based methods, such as NCC and PC,

attempt to address the problem of noise by taking into consideration pixel neighbourhoods. These algorithms are currently considered as one of the most accurate change detection techniques in the literature as they can suppress the effect of noise. Finally, object-based change detection methodologies rely on image segmentation and the detection and extraction of the objects of interest. The performance of such algorithms largely depends on the accuracy of the object segmentation approaches that are employed and therefore, object-based methods have the potential to be very accurate and robust to noise. However, such algorithms usually suffer from low contrast of objects with background, sub-optimal segmentation parameters, ill-defined number of classes and inaccuracies in object extraction.

In this thesis, we investigate pixel-based and region-based approaches and we propose the combination of such methods with object-based approaches to achieve accurate and robust change detection results. Furthermore, we propose a change amplification algorithm, capable of magnifying the true differences between images, thus making them more easily separable from noise and illumination variations. Finally, we develop a CNN-based change detector that is able to both achieve region-based accurate change detection results and identify only changes that are related to buildings. In this way, we make an initial attempt to introduce deep learning techniques in the change detection framework and more advanced and sophisticated CNNs can be proposed and developed in order to achieve state-of-the-art performance in the change detection task.

### 2.7.3 Comparison of state-of-the-art building detection algorithms

In this section, we present and compare the object-based and pixel-based performances of state-of-the-art building detection methodologies on their tested datasets, whenever this information is available, in order for the reader to get a sense of the accuracy the building detection algorithms proposed in the framework of this thesis are expected to have. It is obvious however that these methods cannot be directly compared with each other as they are tested on different datasets. Furthermore, the limitations of the methods in the building detection task are reported. The methods are presented in a chronological order, starting from the oldest one and moving towards more recent methodologies.

Table 2.2: Comparison of state-of-the-art building detection algorithms. OB: Object-based, PB: Pixel-based, MBR: Minimum Bounding Rectangle, MSPA: Morphological Segmentation of binary Patterns, NOAA: National Oceanic and Atmospheric Administration, Prec.: Precision

| Algorithm | Dataset description | Limitations | OB results | PB results |
|---|---|---|---|---|
| Point process of rectangles [67] | Aerial images | Depends on low-level features that may be absent or not correctly extracted. | Recall: 0.91<br>Prec.: 0.961<br>$F_1$-score: 0.934 | Recall: -<br>Prec.: -<br>$F_1$-score: - |
| Corner and texture feature fusion [3] | Ikonos + Aerial images | Noise affects the extraction of corners. | Recall: 0.887<br>Prec.: 0.767<br>$F_1$-score: 0.823 | Recall: -<br>Prec.: -<br>$F_1$-score: - |
| Birth-death dynamics [68] | Aerial images | Depends on low-level features that may be absent or not correctly extracted. | Recall: 0.914<br>Prec.: 0.965<br>$F_1$-score: 0.944 | Recall: 0.837<br>Prec.: 0.668<br>$F_1$-score: 0.743 |
| HOG building detector [2] | Ikonos images | Cannot detect buildings whose shadow is missing. | Recall: 0.658<br>Prec.: 0.66<br>$F_1$-score: 0.944 | Recall: -<br>Prec.: -<br>$F_1$-score: - |
| Hit-or-Miss transform [55] | Ikonos + Geoeye images | Discards multi-spectral information, user-defined classes | Recall: 0.914<br>Prec.: 0.841<br>$F_1$-score: 0.829 | Recall: -<br>Prec.: -<br>$F_1$-score: - |
| Grabcut partitioning of fuzzy landscapes [34] | QuickBird + Geoeye images | Cannot detect buildings whose shadow is missing, tall structures may be mistaken for buildings. | Recall: 0.914<br>Prec.: 0.965<br>$F_1$-score: 0.944 | Recall: -<br>Prec.: -<br>$F_1$-score: - |
| Percentage Occupancy Hit-or-Miss transform [56] | Ikonos + Geoeye + WorldView-2 images | Cannot detect buildings with complex shapes or roofs, may confuse shadowed areas or objects casting shadows. | Recall: 0.918<br>Prec.: 0.822<br>$F_1$-score: 0.867 | Recall: -<br>Prec.: -<br>$F_1$-score: - |
| CNN with MRF refinement [166] | QuickBird + WorldView-2 images | Relative small training set for CNNs, CNN not fine-tuned for building detection. | Recall: 0.794<br>Prec.: 0.917<br>$F_1$-score: 0.851 | Recall: -<br>Prec.: -<br>$F_1$-score: - |
| MRF labeling of super-pixels [51] | NOAA aerial images | Cannot detect buildings whose shadow is missing. | Recall: 0.847<br>Prec.: 0.882<br>$F_1$-score: 0.863 | Recall: 0.9<br>Prec.: 0.877<br>$F_1$-score: 0.887 |
| Flexible multi-label graph partitioning [33] | Ikonos-2 + QuickBird images | Cannot detect buildings with weak colour evidence or complex shapes. | Recall: -<br>Prec.: -<br>$F_1$-score: - | Recall: 0.835<br>Prec.: 0.844<br>$F_1$-score: 0.839 |
| Self-supervised decision fusion (SSDF) [60] | QuickBird + Geoeye + WorldView-2 images | Cannot detect buildings substantially different from the rest, roads can be mistaken for buildings. | Recall: 0.859<br>Prec.: 0.704<br>$F_1$-score: 0.774 | Recall: 0.792<br>Prec.: 0.868<br>$F_1$-score: 0.819 |
| Morphological building index + MSPA [54] | Geoeye + WorldView-2 images | The hypothesis that buildings are brighter than their surroundings can lead to false alarms and missed buildings. | Recall: 0.906<br>Prec.: 0.947<br>$F_1$-score: 0.926 | Recall: -<br>Prec.: -<br>$F_1$-score: - |
| Internal gray variance [58] | Ikonos + WorldView-2 images | Cannot detect buildings whose shadow is missing, tall structures may be mistaken for buildings. | Recall: 0.88<br>Prec.: 0.946<br>$F_1$-score: 0.912 | Recall: -<br>Prec.: -<br>$F_1$-score: - |
| MRF labeling of super-pixels + recursive MBR [52] | NOAA aerial images | Over-detection of building boundaries when low contrast with background, missed buildings when occluded by vegetation. | Recall: 0.833<br>Prec.: 0.854<br>$F_1$-score: 0.843 | Recall: 0.744<br>Prec.: 0.838<br>$F_1$-score: 0.788 |

# CHAPTER 3

## HOG-based Building Detection

In the previous chapters, we decided to decompose the problem of urban change monitoring in two main tasks; building and change detection. Furthermore, we performed a critical review of relevant building and change detection methods and we pointed out their advantages and drawbacks. In this chapter, we address the first task of urban change monitoring, namely the building detection task and we attempt to overcome the challenges other building detection methodologies face.

The possession of 2D high-resolution multi-spectral images, along with the desire to avoid some of the inherent disadvantages of pixel-based building detection approaches, such as the need of meticulous parameter tuning for pixel clustering and the fact that the clusters of pixels cannot easily be associated with buildings, motivated us to turn our attention to object-based building detection methodologies. In this framework, a building is perceived as an entity or a group of pixels with certain properties and a search for discriminatory features that can describe the existence of buildings in an image is performed. As a result, two classes are considered for the building detection problem, the "building" class that describes the existence of buildings and the "non-building" class that characterises any other type of objects that may appear in an urban environment, such as roads, vegetation, ground, water, etc.

To solve the building detection task, we perform a thorough examination and analysis of various features that can be extracted from an image and used for the discrimination of buildings from other objects in an urban environment. Our analysis begins from the heavily and successfully employed for various computer vision tasks, Histogram of Oriented Gradients (HOG), proposed initially by Dalal and Triggs in [7]. The HOG features form the basis, over which other features are searched and combined with, in order to solve the building detection task. The Normalised Difference Vegetation Index (NDVI) [167], the Features from Accelerated Test (FAST) [168] and the Local Binary Patterns (LBP) [70, 71] are considered and their effect on the development of an accurate and robust building detection methodology, when used in conjunction with the HOG features, is

analysed.

Furthermore, a novel distance function that replaces the traditional $l_2$-norm for the computation of distances between feature vectors is proposed. The introduction of the new distance function to a Support Vector Machine (SVM) classifier leads to an enhanced version of our proposed building detector that demonstrates improved robustness to the effect of noise and outliers. Finally, a pixel-based building detection approach that refines the result of our proposed object-based building detection methodology in order to achieve an accurate and robust delineation of building boundaries is introduced. All the algorithms and building detection methodologies that are presented in this chapter are implemented using the C++ programming language, unless otherwise stated. Furthermore, several built-in functions from the OpenCV image processing library [169] are employed.

The main contributions of this chapter are pointed out below:

- A novel probabilistic fusion of HOG, NDVI and FAST features for an improved building detection methodology.
- A complementary use of both HOG and LBP features for an accurate building detector.
- The use of a novel distance function that suppresses the effect of noise and outliers present in datasets.
- The classification of feature vectors using a probabilistic threshold based on the transformation of SVM scores to probabilities using Platt scaling.
- A set of novel region refinement processes for the transition from object-based to pixel-based building detection. Among these processes, the introduction of an unsupervised image segmentation technique, the definition of a novel region scoring procedure that selects initial building candidates and a proposed region merging algorithm for the formation of building hypothesis, are worth mentioning.

The structure of the chapter is as follows. Section 3.1 presents and analyses the HOG feature extraction methodology for the detection of buildings in multi-spectral images. Section 3.2 introduces the NDVI and FAST features that are probabilistically fused with the HOG features for improved building detection performance. Section 3.3 presents the advantages of the supplementary use of both HOG and LBP features, while Section 3.4 extends the HOG-LBP features by proposing a novel and robust distance function for their classification. Section 3.5 introduces a set of region refinement processes that makes the transition from object-based to pixel-based building detection possible. Finally, Section 3.6 compares the performances of the proposed building detection methodologies, while Section 3.7 concludes the work done in this chapter by discussing the contributions to the building detection task.

The structure of the chapter is in a form, in which the proposed HOG-based building

detectors are presented both in chronological order and in ascending (i.e. based on their performance on the building detection task) order. This means that the worst performance is achieved by the HOG building detector that was initially developed, while chronologically subsequent attempts to improve the performance of a building detector led to the best object-based performance achieved by the enhanced HOG-LBP building detector. Table 3.1 summarises the key characteristics of the building detection methodologies that are analysed and presented in the next sections.

Table 3.1: Key characteristics of proposed building detection methodologies.

| Methodology | Key characteristics |
|---|---|
| HOG building detector | • Optimised for the building detection task HOG descriptors |
| Improved HOG building detector | • HOG building detector<br>• Extraction of NDVI and FAST features<br>• Novel probabilistic fusion of HOG, NDVI and FAST features |
| HOG-LBP building detector | • HOG building detector<br>• Extraction of LBP features<br>• HOG feature extraction from additional colour channel<br>• New block configuration for HOG and LBP feature computation |
| Enhanced HOG-LBP building detector | • HOG-LBP building detector<br>• Novel distance function for SVM training and testing<br>• Platt scaling and new threshold for classification of feature vectors |
| Pixel-based building detector | • Enhanced HOG-LBP building detector<br>• Novel region refinement processes |

## 3.1 HOG building detector

The HOG descriptor was initially introduced in 2005 as a robust feature that detects pedestrians in images [7]. Since then, the HOG features were adopted by many state-of-the-art object detection methodologies that were successfully employed in several computer vision applications. There are many reasons why an object detection methodology based on HOG features is suitable for building detection. First of all, the aggregation of local gradients in histogram bins enables the HOG features to be invariant to geometric transformations and occlusions, which is important in urban environments, where buildings can be occluded by vegetation or other buildings. Secondly, the HOG features are illumination invariant due to the fact that their computation is based on gradient magnitudes and orientations. This is especially important in remote sensing images, where the position of the sun, the time images are captured, can cause severe illumination variations in the images. Finally, the HOG features are fast to compute, and therefore a classification method employing HOG features can run in real-time.

### 3.1.1 HOG feature extraction

The proposed HOG building detector exploits the multi-spectral information of the images. This means that the implemented algorithm takes full advantage of the provided data, in

the form of the available multi-spectral channels of the images, and more specifically the red, green, blue and near-infrared channels. The use of all available channels allows a better discrimination ability and classification accuracy for the proposed building detection algorithm.

A HOG descriptor is computed in an image region that is further divided into subregions, which are called cells. In each cell, a 1D histogram of the orientations of the gradients of the pixels present inside the cell is computed. By tuning the parameters that affect the creation of the histograms, a feature can be developed that can differentiate image regions that contain buildings from those that do not. The proposed HOG building detector, illustrated in Fig. 3.1, is an extension of the standard approach for creating and employing HOG features, suggested in [170] and it consists of two phases, i.e. a training and a testing phase.

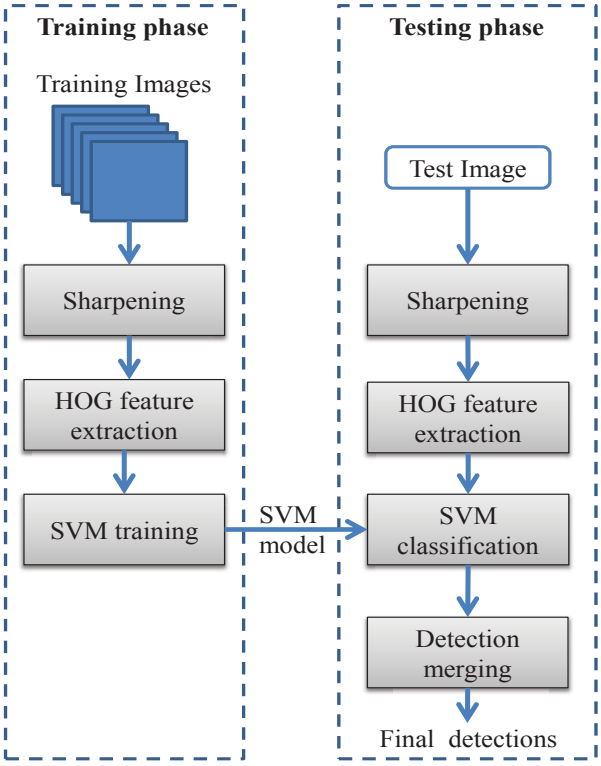



Figure 3.1: The proposed HOG building detector.

In the training phase, manually labelled images are employed and HOG descriptors for the "building" and "non-building" classes are extracted. Every image is pre-processed before the computation of the HOG descriptors. All pixels are initially divided with the maximum discretisation value, so that they are in the range $[0, 1]$ and then a sharpening filter is applied. A sharpened image, denoted as $sharp(I)$, is the product of the subtraction of the initial image $I$ with the same image convolved with a Gaussian filter $G$

$$sharp(I) = (1 + \zeta)I - \zeta I * G \tag{3.1}$$

The parameter $\zeta$ takes values in the range $[0, 1]$ and affects the amount of sharpening that is applied to an image. The purpose of the sharpening procedure is the enhancement of the edges of an image so that buildings can become more distinguishable in an urban environment.

After pre-processing, signed gradients are computed by employing the Scharr operator separately to each multi-spectral image channel. In each cell, the computed gradients are used to cast votes into histogram bins, weighted by their magnitudes. Moreover, the gradient magnitudes are trilinearly interpolated in the neighbouring cells and bins to increase the robustness of the HOG detector in slight rotations or translations of the object of interest [170]. A single block with a rectangular kernel is employed for the HOG descriptor extraction in an image region. The computed histograms, one for each channel of a multi-spectral image, are concatenated into a single histogram/descriptor. The HOG descriptors remain unnormalised, since such a strategy produces better results than any normalisation scheme (see Section 3.1.2). The entire HOG feature extraction procedure is illustrated in Fig. 3.2.

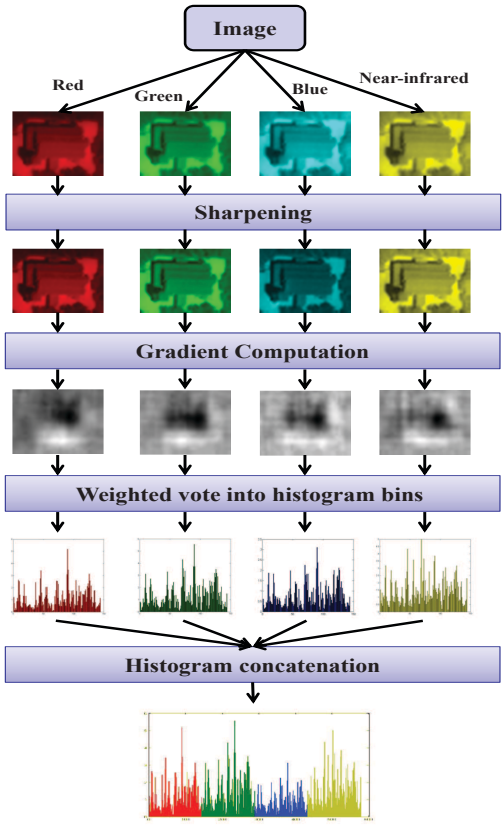



Figure 3.2: The proposed HOG feature extraction procedure.

The extracted HOG descriptors are then introduced to a SVM classifier for training. A SVM classifier is a supervised machine learning algorithm that was initially introduced by Vapnik [171]. Given a set of training examples $\mathbf{x_i}$, $i = 1, ..., N$, along with their labels $y_i \in \{-1, 1\}$, the SVM classifier finds a hyperplane described by equation $\mathbf{w} \cdot \mathbf{x} + b = 0$

to separate the training samples to two classes. In most cases, the training samples cannot be linearly separated and therefore non-negative values $\xi_i$, which are equal to $max(0, 1 - y_i(\mathbf{w}x_i + b))$, are introduced and allow samples to be classified in the opposite class with a penalty. The soft-margin SVM classifier solves the classification problem by minimising

$$\arg\min_{\mathbf{w},b} \|\mathbf{w}\|^2 + C\sum_{i=1}^{N} \xi_i, \tag{3.2}$$

$$\text{subject to } y_i(\mathbf{w} \cdot \mathbf{x_i} + b) \geq 1 - \xi_i \tag{3.3}$$

The solution to this minimisation problem can be expressed as a linear combination of only a few of the training samples that are called support vectors and for which, it holds that $y_i(\mathbf{w} \cdot \mathbf{x_i} + b) = 1$. The cost of outliers $C$ controls the tolerance of the classifier to outliers. This means that large outlier costs will make the classifier choose a smaller-margin hyperplane to avoid the misclassification of training samples. On the other hand, small outlier costs will make the classifier choose a wider-margin hyperplane at the expense of classifying several training samples incorrectly. Kernels can be employed to transform the training samples $\mathbf{x_i}$ to a higher-dimensional space, where the training data can be more easily separated. One of the most common kernels for a higher-dimensional mapping and the one that is also used in the proposed methodology is the Radial Basis Function (RBF), which can be expressed with the general form

$$RBF = e^{-\gamma\|\mathbf{x_i}-\mathbf{x_j}\|^2} \text{ for } \gamma \geq 0 \text{ and } i, j = 1, ..., N \tag{3.4}$$

A SVM model is trained and all the parameters that affect the performance of the SVM classifier (i.e. cost of outliers $C$ and kernel parameter $\gamma$) are optimally chosen so that the cross-validation estimate of the test error is minimal. The SVM model is afterwards used for classification in the testing phase.

In the testing phase, each test image is split in overlapping regions of multiple sizes (scales) and a HOG descriptor is extracted for each image region. Afterwards, the HOG descriptors are classified to the "building" or "non-building" class using the SVM model that was previously trained. The detections with positive scores are accepted as candidates for the presence of buildings in an image.

The HOG building detector produces overlapping building candidate regions; hence a detection merging procedure is required. For this reason, a mean-shift algorithm is employed to reduce the number of detected regions [170]. More specifically, the regions can be considered as points $(x, y, z)$ in the 3D space weighted by a score, which in this case is the output of the SVM classifier. The variables $x$, $y$ and $z$ represent the horizontal and vertical coordinates of the centre of a region and the logarithm of the scale where the region was

detected respectively [170]. The use of logarithm for the scale parameter ensures detection homogeneity in the 3D space as the scale parameter increases geometrically in a multi-scale framework. A mean-shift algorithm is a mode seeking algorithm that evaluates the distribution of points in the 3D space and identifies the local maxima of the distribution. This allows detections that are close to each other in the 3D space to merge into a single detection. A set of smoothing parameters that affects the Gaussian smoothing kernel used for the kernel density estimate of the mean-shift algorithm should also be defined. These parameters describe how far in the 3D space points can be in order to be merged.

However, there may still be overlapping detections that cannot be merged because the selected smoothing parameters may not sufficiently describe the distribution of detections in a tested image. To cope with this problem, a rectangle grouping algorithm that identifies and merges overlapping detections is proposed. Two detection windows are considered overlapping, if at least half the area of one detection window is enclosed within the other detection window. In the first phase, the algorithm discards large detection windows that overlap with two or more smaller detection windows that do not overlap with each other. Such large detection windows cannot be considered as feasible building regions because they usually contain two or more buildings that are described by the smaller detection windows. In the second phase, the remaining pairs of overlapping detection windows are compared and the detection with the highest score is preserved, while the other is discarded. An example of the two phases of the developed region merging algorithm is presented in Fig. 3.3, where the detections that are removed by the algorithm are shown in red colour. The remaining detections define the final output of the proposed object-based building detectors.

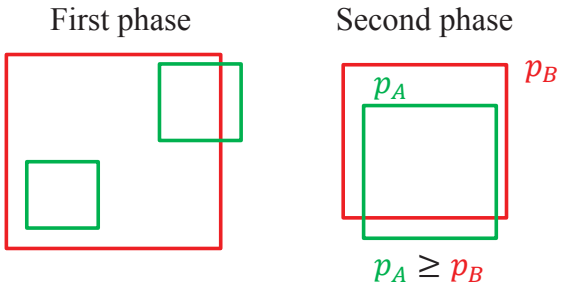



Figure 3.3: The two phases of the proposed rectangle grouping algorithm.

### 3.1.2 HOG parameter optimisation

This section presents the experiments performed for the optimisation of the parameters that affect not only the values of the computed gradient magnitudes and orientations, but also the size of the HOG descriptors. It is shown that the optimisation of such parameters is crucial in order for the HOG descriptors to be as discriminatory as possible and the performance of a HOG-based building detector to substantially improve. The

optimisation of the HOG parameters is presented in Fig. 3.6 and is thoroughly explained below. The optimal parameters are selected based on the Precision-Recall (PR) curves that the proposed HOG building detector of Section 3.1 achieves on the Athens "sample" validation set, defined in Section 2.5. It should be mentioned that the most appropriate way to optimise the values of the HOG parameters is by performing grid search. However, in order to avoid the computational overhead, we decide to perform an initial search based on our knowledge of the problem and acquire a good set of HOG parameter values and then optimise each HOG parameter irrespectively of the values of the other HOG parameters. The optimal HOG parameter values are kept fixed for the other proposed HOG-based building detectors, unless otherwise stated.

More specifically, experiments are performed to define how the performance of the HOG building detector is affected by the type of pre-processing or filtering that can be applied to the training set prior to the computation of the HOG descriptors. Various filtering techniques, such as Gaussian filtering, bilateral filtering, median filtering and sharpening, as it is defined in Eq. (3.1) are attempted. The size of the filters for every pre-processing technique is equal to $5 \times 5$ pixels, while the parameter $\zeta$ for the sharpening technique is chosen equal to 0.5. The results, presented as PR curves in Figure 3.6(a), show that the sharpening technique performs slightly better than the use of no pre-processing and presents a sufficient advantage over other pre-processing methods. This means that enhancing details improves significantly the performance of the HOG building detector and, therefore, techniques that blur the image, such as median and Gaussian filtering, degrade the classification accuracy of the algorithm.

Furthermore, experiments are conducted in order to define optimal gradient magnitudes and orientations. Since there are $N$ number of channels in a multi-spectral image, there are $N$ different gradient magnitudes and orientations that can be computed for each image pixel. On this basis, we test three different approaches. The first approach computes a different histogram for each spectral channel and then the histograms are concatenated in a single large histogram, while the other two approaches achieve a type of gradient merging in order to compute only a single histogram for each multi-spectral image and thus reducing the size of the computed HOG descriptor.

Given that the terms $C_{pX}$ and $C_{pY}$ represent the gradients of a pixel $p$ of a channel $C$ of a multi-spectral image along the horizontal and vertical directions respectively, the first gradient merging method identifies for each pixel the channel associated with the largest gradient magnitude among the channels of a multi-spectral image

$$Cmax_p = \arg\max_C \sqrt{C_{pX}^2 + C_{pY}^2} \tag{3.5}$$

Based on this method, the gradient magnitude $F_p$ and orientation $\theta_p$ for each pixel $p$ are computed in a straightforward manner

$$F_p = \sqrt{Cmax_{pX}^2 + Cmax_{pY}^2} \tag{3.6}$$

$$\theta_p = \arctan\left(\frac{Cmax_{pY}}{Cmax_{pX}}\right) \tag{3.7}$$

The second tested gradient merging method was initially proposed in [172] and it is based on the notion of structure tensors. In this case, the gradient tensor $G_p$, the gradient magnitude $F_p$ and the orientation $\theta_p$ of a pixel $p$ are computed as follows

$$G_{pxy} = \sum_C C_{px} C_{py}, \qquad x, y \in \{X, Y\} \tag{3.8}$$

$$\theta_p = \frac{1}{2}\arctan\left(\frac{2G_{pXY}}{G_{pXX} - G_{pYY}}\right) \tag{3.9}$$

$$F_p = G_{pXX}\cos^2(\theta_p) + 2G_{pXY}\cos(\theta_p)\sin(\theta_p) + G_{pYY}\sin^2(\theta_p) \tag{3.10}$$

If $\theta_p$ is a solution, so is $\theta_p \pm \frac{\pi}{2}$. In such cases, the orientation associated with the largest gradient magnitude $F_p$ is used for the histogram computation. In the special case, where $G_{pXX} = G_{pYY}$ and $G_{pXY} = 0$, $\theta_p$ cannot be computed from Eq. (3.9) and the corresponding pixel is not taken into account for the histogram computation.

The performance of the proposed HOG building detector for the tested gradient merging methods on the Athens "sample" validation set is presented in Fig. 3.6(b) and shows that both gradient merging methods are inferior with respect to the approach of computing individual HOG features for each multi-spectral channel of an image and then concatenating them into a single histogram.

Another experiment concerns the selection of an optimal mask for the gradient computation. Three masks are considered and tested. The first mask is a simple centred mask, which can be expressed as a $[-1\ 0\ 1]$ mask. The other masks are the Sobel and the Scharr masks with sizes $3 \times 3$ pixels. The effect of the different masks on the performance of the HOG building detector on the Athens "sample" validation set are demonstrated in Fig. 3.6(c). According to the experiments, the Scharr mask performs slightly better than the Sobel mask and much better than the centred mask. As a consequence, the Scharr mask is chosen as the optimal gradient computation mask for the proposed HOG building detector.

When a HOG descriptor is extracted from an image region, the HOG algorithm computes a set of histograms of gradients for each block of the image region. A block is defined as an area that can cover the whole image region or a part of it. As a result, many block configurations can be applied that select certain parts of an image region and such configurations can severely affect the performance of a HOG algorithm. In the framework of this thesis, three block configurations are examined in order to demonstrate their effect

on the performance of the HOG building detector. The first configuration is a simple block that covers the entire image region. The second configuration consists of four non-overlapping blocks that each covers a quarter of the area of an image region, while the third configuration consists of five blocks, each covering a quarter of the area of an image region. Four blocks are placed as in the second configuration, while the fifth lies in the middle of the image region, partially overlapping with all other blocks. Fig. 3.4 clarifies the three block configurations that are used to evaluate the classification accuracy of the HOG building detector.

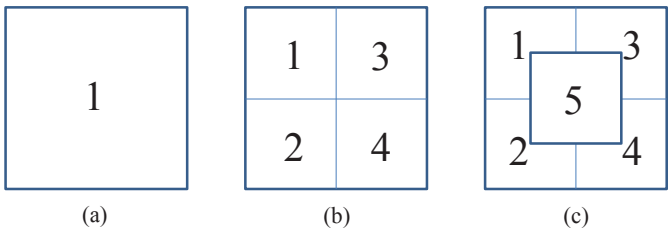



Figure 3.4: Illustration of the (a) first, (b) second and (c) third tested block configurations.

The number of histograms that are computed for each block depends on the number of cells a block consists of. For this purpose, both rectangular and circular HOG kernels are tested for the three block configurations that were presented above. A rectangular kernel is divided in four smaller rectangular cells, while a circular kernel consists of two radial cells with the outer cell split in four angular cells. As a result, a circular HOG kernel is composed of five cells. Fig. 3.5 clarifies the configuration of cells in the employed rectangular and circular kernels.

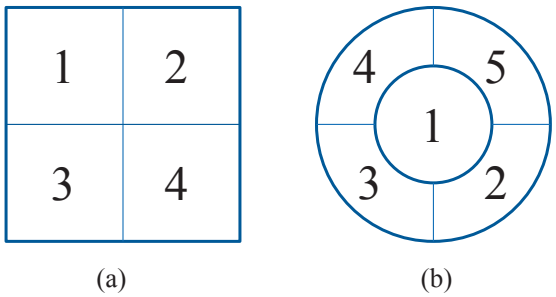



Figure 3.5: Illustration of the employed (a) rectangular and (b) circular kernels.

The results of the experimentation on the kernel type and block configuration on the Athens "sample" validation set are presented in the PR curves of Fig. 3.6(d). It can be concluded that the configuration with a single block and a rectangular kernel of four cells outperforms all the other kernel and block configurations. The second best configuration is a single block with a circular kernel. As a result, a single rectangular kernel is chosen as the optimal kernel for the extraction of HOG descriptors.

Experiments are also conducted in order to determine how the signed (0°-360°) and unsigned (0°-180°) gradients and the size of the histogram bins affect the performance of

the HOG building detector. These parameters directly affect the size of the computed HOG descriptors. Two sizes for the histogram bins are tested, namely 10° and 20°. As a result, the histogram bins vary in number from 9, in the case when unsigned gradients with 20°-sized bins are used, to 36 when signed gradients with 10°-sized bins are employed.

The results of the experimentation with the number of histogram bins are summarised in Fig. 3.6(e). It can be concluded that the HOG building detector with signed gradients and histogram bins of size 10° performs slightly better than the HOG building detector with signed gradients and histogram bins of size 20° and much better than the HOG detectors that employ unsigned gradients. As a consequence, the signed gradients with 10°-sized histogram bins are chosen for the proposed HOG building detector. This choice results in the extraction of a 36-bin HOG feature vector for each multi-spectral channel in a cell of an image block.

An important factor that affects the performance of an algorithm that is based on HOG features is the normalisation of the HOG descriptors. Feature normalisation is usually performed in order to make a classifier illumination invariant. We experiment on various normalisation schemes and evaluate the performance of the HOG building detector. More specifically, we attempt block normalisation using either the $l_1$-norm or the $l_2$-norm, no normalisation and whole feature normalisation after block normalisation using the $l_1$-norm and $l_2$-norm. The $l_1$-norm and $l_2$-norm of a descriptor $\mathbf{A}$ are defined in Eq. (3.11) and (3.12) respectively.

$$\|\mathbf{A}\|_{l_1} = \sum_i A_i \tag{3.11}$$

$$\|\mathbf{A}\|_{l_2} = \sqrt{\sum_i A_i^2} \tag{3.12}$$

The results from running the HOG building detector on the Athens "sample" validation set and employing the various normalisation schemes are shown as PR curves in Fig. 3.6(f). The conclusion that can be drawn from these PR curves is that the performance of the HOG building detector, when the HOG descriptors are raw (i.e. no normalisation) is optimal with respect to the other tested normalisation techniques that employ either the $l_1$-norm or the $l_2$-norm. This can be attributed to the fact that the accumulative raw gradient magnitudes in the HOG descriptors allow for an accurate discrimination between the "building" and the "non-building" classes and this effect subsides, when any kind of normalisation is performed. As a result, the HOG descriptors remain unnormalised in the implementation of the proposed HOG building detector.

Another experiment is performed so as to determine the effect of the gamma correction on the performance of the HOG building detector. Gamma correction takes the square root of the value of each pixel as the pixel's representative value in order to compensate for

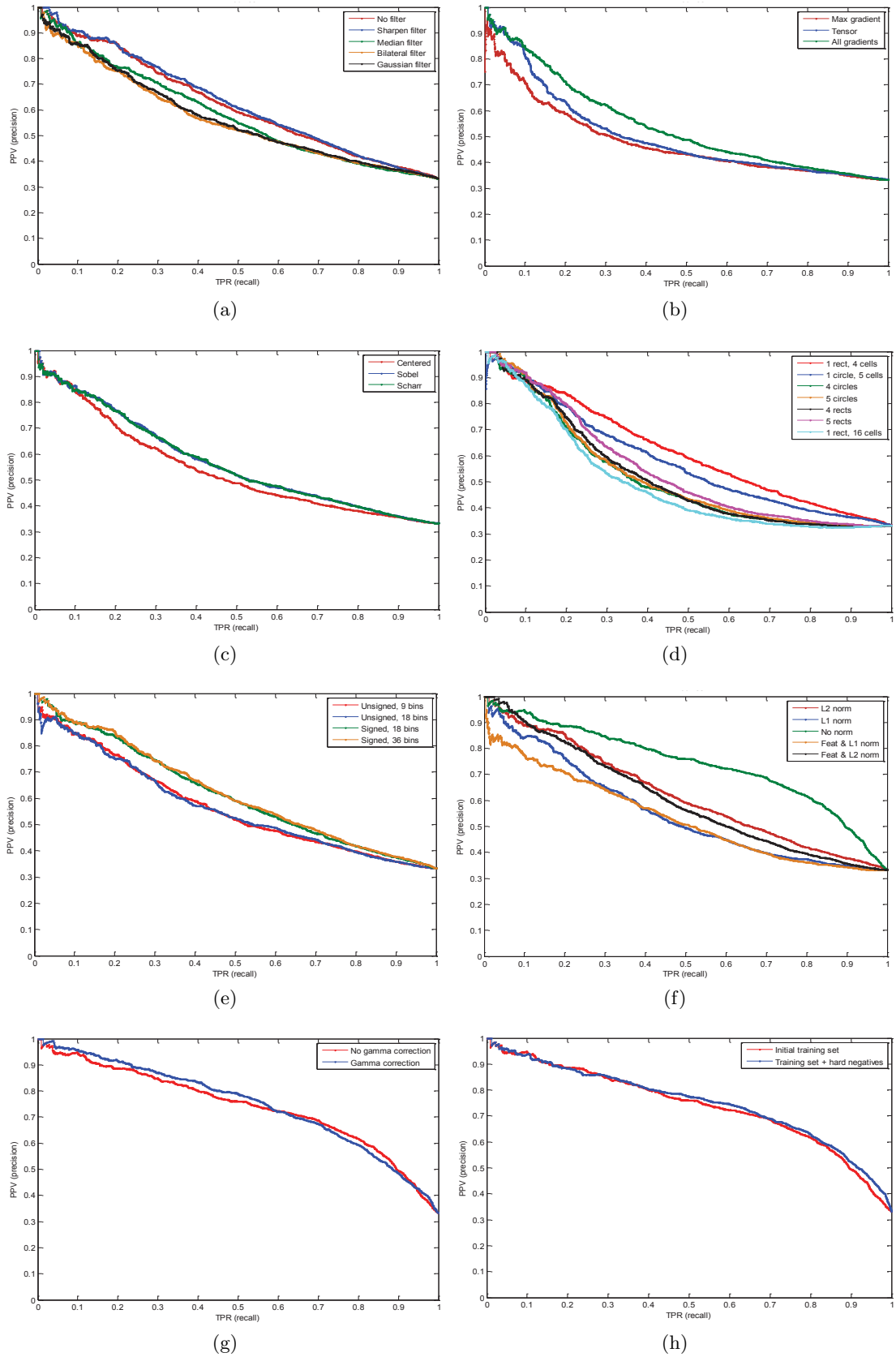



Figure 3.6: The optimisation of each HOG parameter is based on the performance of the proposed HOG building detector on the Athens "sample" validation set.

distortions in the viewing process. Gamma correction was suggested by Dalal and Triggs as it improves the performance of a human detector [7]. The PR curves in the case, where no gamma correction is used and in the case, where gamma correction is applied are presented in Fig. 3.6(g). A comparison of these PR curves cannot easily clarify whether gamma correction improves the performance of the proposed HOG building detector. However, a comparison of the maximum $F_1$-score achieved on the Athens "sample" validation set by the HOG building detector with and without gamma correction reveals that the maximum $F_1$-score drops by about 1.4% when gamma correction is employed. Consequently, applying gamma correction in the proposed HOG building detector is not an optimal choice.

Finally, a useful technique to improve the classification performance of an algorithm is the introduction of hard negatives to the training procedure. Hard negatives are negative examples that are incorrectly classified by the tested classifier. These examples are re-introduced to the classifier for further training in order for the new algorithm to learn from its own mistakes. Such a technique has been successfully employed in the literature [173, 174]. To improve the classification performance of our proposed HOG detector in the building detection task, images that contained no buildings (i.e. negative examples) are segmented from the large QuickBird image of the Athens dataset and they are used for testing. The examples that are incorrectly classified as buildings (i.e. false positives), are re-introduced to the algorithm and a new training phase initiates. The number of the hard negatives is equal to 400. Afterwards, the algorithm is tested on the Athens "sample" validation set and the PR curves are plotted in Fig. 3.6(h) in order to analyse the influence of the introduction of the hard negatives on the classification performance of the proposed HOG building detector.

The experimentation shows that the new training set (i.e with the introduction of the hard negatives) improves slightly the classification performance of the HOG building detector on the Athens "sample" validation set, as it was expected. More specifically, the maximum $F_1$-score increases by 1.3%, when the hard negatives are introduced. The introduction of hard negatives in the training set concludes the series of experiments that are performed in order to select the optimal parameters that affect the extraction of the HOG descriptors. The optimal parameter configuration leads to four histograms, one for each channel of the multi-spectral image (i.e. red, green, blue and near-infrared) and each of these histograms has 144 features (i.e. 4 cells $\times$ 36 bins per cell). The total length of the HOG descriptors is therefore 576.

### 3.1.3 Evaluation

In order to detect buildings of various sizes in an image, an algorithm based on HOG features should run in multiple scales (i.e. sizes of image regions), covering a range between a minimum and a maximum scale. The initial size in the case of a QuickBird image is $20 \times 20$ pixels and in the case of a WorldView-2 image is $22 \times 22$ pixels. Given that the

spatial resolutions of the two satellites are $0.6m$ per pixel for the QuickBird and $0.5m$ per pixel for the WorldView-2 satellite, the chosen initial searching window sizes are found to be adequate to detect buildings with areas as small as $50m^2$. The ratio between two consecutive scales is selected to be equal to 1.1. This means that the size of the next searching window is 1.1 times bigger than the size of the previous searching window.

Moreover, in order to keep the pixels inside the searching window constant no matter how large the searching window is, instead of increasing the size of the searching window, the size of the tested image is scaled down and the size of the searching window remains as initially defined. The maximum scale that should be searched depends on the largest building that the algorithm should detect. In the framework of this thesis, the image is scaled down to such an extent that in the last scale the searching window is equal to $110\times110$ and $130\times130$ pixels for the Quickbird and WorldView-2 images respectively. Such sizes can make a HOG building detector capable of identifying large industrial buildings of sizes approximately equal to $3000m^2$.

An algorithm based on HOG features should not only search in multiple scales, but also in overlapping windows of the same scale in order to increase the chance of successfully extracting the objects of interest. For this purpose, the displacement between two consecutive extracted image regions is selected to be equal to 5 pixels in the horizontal or the vertical direction regardless the type of multi-spectral image tested. Such a displacement could perhaps be considered small, given the sizes of the searching windows, leading to 75% and 77.2% overlap between two consecutive searching windows, in the case when a QuickBird and a WorldView-2 satellite image is tested respectively. However, such a displacement ensures that most of the buildings will be correctly identified and extracted.

As already mentioned in Section 3.1.1, the mean-shift algorithm, which is responsible for the merging of detections by locating the local maxima of their distribution requires a set of smoothing parameters that defines how close the detections should be in order to be merged. After experimentation, these smoothing parameters are chosen to be equal to 3 pixels for both horizontal and vertical direction and $\log(1.3)$ for the scale. Such values are found to be both adequate for merging detections that have a significant overlap and sufficiently small so as to distinguish buildings that are quite close to each other.

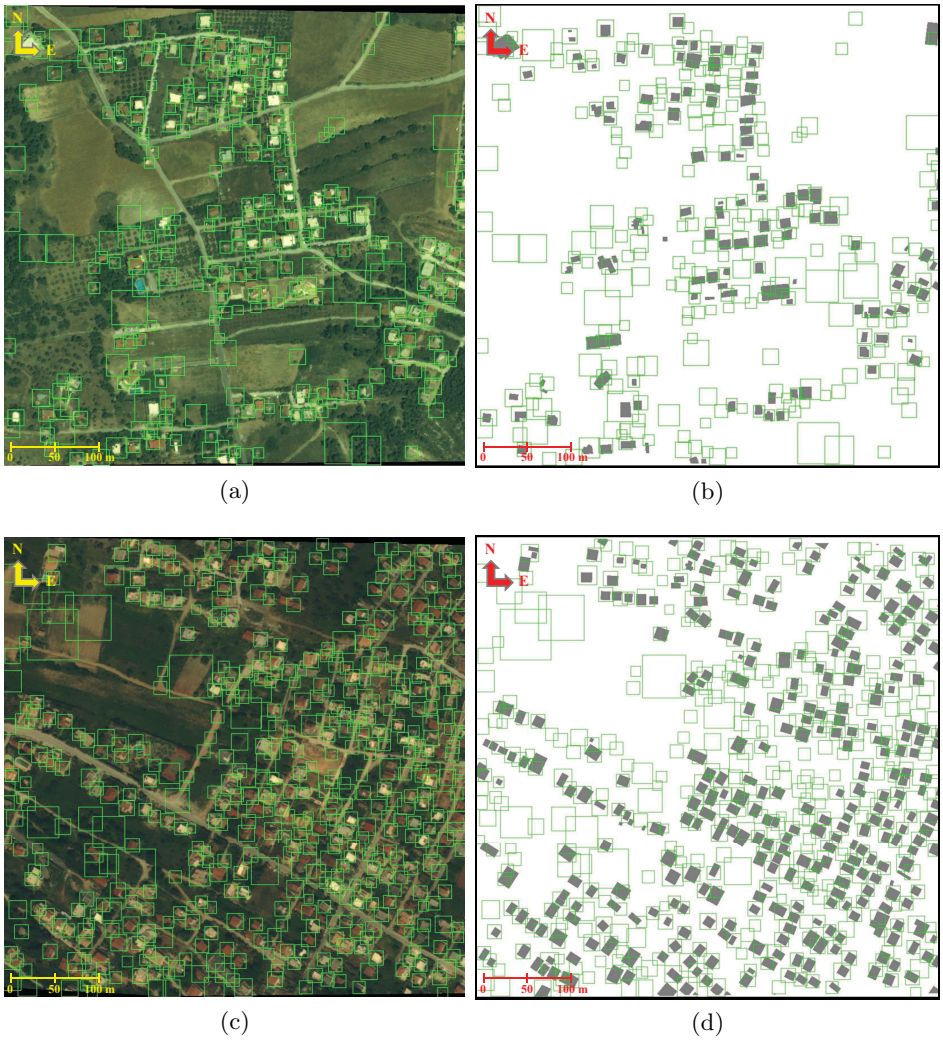
The output of the proposed methodology is the final extracted image regions after the mean-shift algorithm and our rectangle grouping procedure. These candidate building regions are then checked whether they detect a building or not, and the true positives, false alarms and missed buildings are computed based on the ground truth data that are available. The performance of the proposed HOG building detector is evaluated on the entire Athens test set and the metrics of recall, precision and $F_1$-score, defined in Section 2.6, are presented in Table 3.2. Furthermore, an illustration of the building detections of the HOG building detector in two images of the Athens test set is presented in Fig. 3.7. The HOG detections are shown as green bounding boxes overlaid on the test images.

Table 3.2: Results of the proposed HOG building detector on the Athens test set.

| Image | Recall | Precision | $F_1$-score |
|---|---|---|---|
| area_1a_2006 | 0.914 | 0.397 | 0.554 |
| area_1b_2006 | 0.934 | 0.469 | 0.624 |
| area_1c_2006 | 0.916 | 0.59 | 0.718 |
| area_2a_2006 | 0.898 | 0.414 | 0.566 |
| area_2b_2006 | 0.89 | 0.368 | 0.521 |
| area_2c_2006 | 0.906 | 0.481 | 0.629 |
| area_1a_2007 | 0.752 | 0.629 | 0.685 |
| area_1b_2007 | 0.677 | 0.866 | 0.76 |
| area_1c_2007 | 0.687 | 0.904 | 0.781 |
| area_2a_2007 | 0.729 | 0.714 | 0.721 |
| area_2b_2007 | 0.653 | 0.667 | 0.66 |
| area_2c_2007 | 0.714 | 0.77 | 0.741 |
| area_1a_2009 | 0.895 | 0.467 | 0.613 |
| area_1c_2009 | 0.906 | 0.652 | 0.758 |
| area_2a_2009 | 0.904 | 0.514 | 0.656 |
| area_2b_2009 | 0.879 | 0.47 | 0.612 |
| area_2c_2009 | 0.925 | 0.559 | 0.697 |
| area_1a_2010 | 0.932 | 0.506 | 0.656 |
| area_1b_2010 | 0.917 | 0.584 | 0.713 |
| area_1c_2010 | 0.908 | 0.712 | 0.798 |
| area_2a_2010 | 0.917 | 0.538 | 0.678 |
| area_2b_2010 | 0.935 | 0.448 | 0.606 |
| area_2c_2010 | 0.936 | 0.522 | 0.67 |
| area_1a_2011 | 0.851 | 0.502 | 0.632 |
| area_1b_2011 | 0.833 | 0.634 | 0.72 |
| area_1c_2011 | 0.871 | 0.735 | 0.797 |
| area_2a_2011 | 0.888 | 0.554 | 0.682 |
| area_2b_2011 | 0.866 | 0.505 | 0.638 |
| area_2c_2011 | 0.842 | 0.57 | 0.68 |
| Mean $\pm$ std | 0.858 $\pm$ 0.085 | 0.577 $\pm$ 0.131 | 0.675 $\pm$ 0.069 |

From the results of the proposed HOG building detector in the Athens test set, a few conclusions can be drawn. The performance of the HOG building detector is not severely degraded, despite the fact that the images of the test set exhibit strong illumination variations and spectral and spatial resolution differences. The robustness of an algorithm based on HOG features to such variations and resolution differences is the reason it was initially chosen for the detection of buildings in images. Our HOG building detector manages to detect most of the buildings present in the test images, achieving a high recall, which plays the most significant role in a building detection framework. On the contrary, the precision of the proposed HOG building detector is mediocre. A first comparison with the performances of state-of-the-art methods, presented in the literature (Section 2.7.3) shows that the proposed building detector cannot currently compete with them as far as precision is concerned. However, the overall performance of the proposed HOG building detector is decent given the fact that the algorithm employs only HOG features and the

Athens dataset is quite challenging, with images presenting noise, significant illumination variations and depicting buildings with complex shapes and occluded by vegetation. In the following sections of this chapter, we study ways to improve the precision and generally, the performance of a building detector by introducing additional features and ways to increase the robustness of a classifier to the effect of noise.




(a) (b)

(c) (d)

Figure 3.7: Detections from the proposed HOG building detector in (a) a QuickBird satellite image and (c) a WorldView-2 satellite image. The detections are shown as green boxes bounding the detected building areas. The binary masks of ground truth building locations (b) and (d) are also included for comparison, corresponding to images (a) and (c) respectively.

## 3.2 Improved HOG building detector

To improve the performance of the proposed HOG building detector, presented in Section 3.1, the extraction of additional features is considered in this section. More specifically, the NDVI and FAST features are studied and their effect on the performance of the HOG building detector on the Athens dataset is evaluated. The new information is fused with

the information derived from the HOG features in order to form a new methodology, called improved HOG building detector that outperforms the HOG building detector, proposed in the previous section. An outline of the improved HOG building detector is presented in Fig. 3.8. The part of the improved HOG building detector that computes the HOG descriptors is the same as the one presented in Section 3.1.1. The differences between the two proposed methodologies lie in the computation of the NDVI and FAST features, the transformation of the HOG scores to probabilities and the probabilistic fusion of the HOG, NDVI and FAST probabilities.

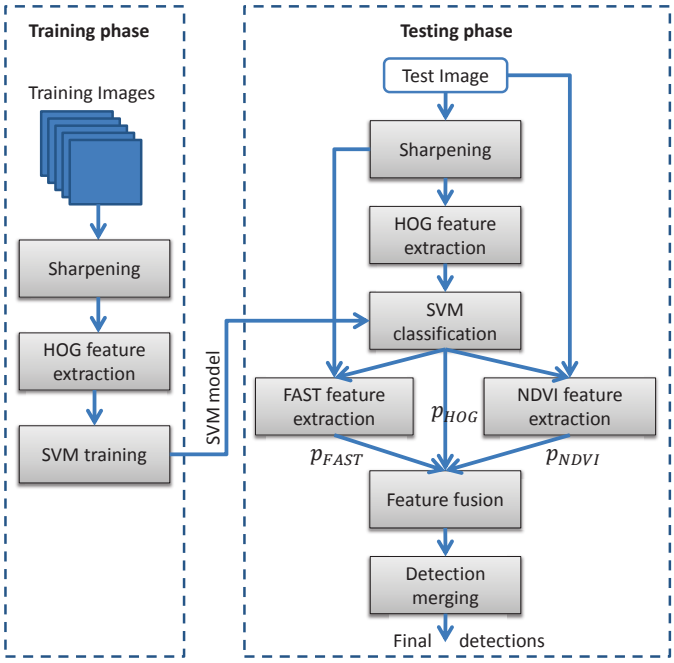



Figure 3.8: The proposed improved HOG building detector with the addition of NDVI and FAST features.

### 3.2.1 Platt scaling

To probabilistically fuse the information derived from the HOG, NDVI and FAST features, it is imperative that the SVM scores that describe the distances of the HOG descriptors from the SVM separating hyperplane are expressed as probabilities. This can be achieved by Platt scaling [175]. The Platt scaling concerns the transformation of scores $x$ of a SVM classifier to probabilities $p_{HOG}$ using the sigmoid function

$$p_{HOG} = \frac{1}{1 + e^{Ax+B}} \tag{3.13}$$

The constant terms $A$ and $B$ are determined iteratively by minimising the negative log likelihood of the training data

$$min(-\sum_{i}(t_i * log(p_{HOG_i}) + (1 - t_i) * log(1 - p_{HOG_i}))) \tag{3.14}$$

where $t_i$ is equal to 0 for negative samples and 1 for positive samples and $p_{HOG_i}$ is the probabilistic result of Eq. (3.13) for a given sample [175]. The purpose of the minimisation is to make the sigmoid function better fit the training data. In order to produce the optimal estimates of the Platt parameters $A$ and $B$, it is crucial that the minimisation is performed on a validation set that is different from the training set. To achieve this, we split the training set in non-overlapping subsets and the HOG algorithm is trained each time on all but one subset and the Platt parameters are evaluated using the remaining subset (i.e. cross-validation).

An initial set of candidate building regions is formed by keeping only the image regions that are classified in the "building" class by the SVM model (i.e. SVM score $x$ higher than 0), since experimentation shows that the separating hyperplane gives the best discriminative power to the developed algorithm. Afterwards, the NDVI and FAST features are computed only for the initial set of candidate image regions. The goal of this approach is to compute the additional NDVI and FAST features for the fewest possible candidate building regions, thus decreasing the additional computational overhead of the proposed improved HOG building detector and reducing the possibility of the algorithm producing additional false alarms. A disadvantage of this approach is that building areas that are discarded from the classification procedure due to negative SVM scores have no chance of being recovered. However, the HOG building detector, proposed in the previous section, is already very accurate in detecting buildings and we do not expect any significant loss on its building detection capabilities.

### 3.2.2 Computation of NDVI features

The NDVI [167] is a well-known index that is used in several remote sensing applications to distinguish vegetated from non-vegetated areas. This index takes advantage of the near-infrared band that most satellites provide to identify vegetation. The use of NDVI is based on the fact that light is absorbed more from vegetation than from buildings and roads and consequently, this index produces higher values for vegetated, rather than non-vegetated areas. NDVI is computed using the near-infrared $\rho_{NIR}$ and red $\rho_R$ channels as shown below

$$NDVI = \frac{\rho_{NIR} - \rho_R}{\rho_{NIR} + \rho_R} \tag{3.15}$$

After the computation of the NDVI index for each positive HOG detection, a binary NDVI mask is formed that identifies which pixels correspond to vegetation. To achieve this, a threshold is required and pixels with NDVI values that surpass the threshold are

considered as vegetation. Instead of manually computing such a threshold, we employ the Otsu's method to automatically define an optimal threshold. The Otsu's method selects the optimal threshold that separates two classes based on the minimisation of the intra-class variance, which is the same as the maximisation of the inter-class variance. A probability $p_{NDVI}$, defined in Eq. (3.16), is then computed for each region $R$ and is equal to the number of pixels in the region that do not belong to the "vegetation" class $V$, divided by the total number of pixels in the region. The higher the probability $p_{NDVI}$ is, the fewer pixels belong to vegetation and the higher the chance that the candidate region describes a man-made urban area that may correspond to a building.

$$p_{NDVI} = \frac{\#[p \notin V]}{\#p}, p \in R \tag{3.16}$$

### 3.2.3 Computation of FAST features

Buildings, due to their rectangular shape, usually have strong corners, which is an important cue for building segmentation. The FAST algorithm is a robust corner detector, which provides a set of corner features (pixels) along with a corresponding score (i.e. intensity difference to their neighbouring pixels) [168]. In the proposed improved HOG building detector, FAST features are computed for each sharpened multi-spectral channel of an image. If two or more FAST features are identified in the same position (pixel) among the different channels, the one with the largest score is preserved.

For each candidate building region, a probability $p_{FAST}$ is computed based on the FAST features by employing the following strategy. This strategy is based on the notion that building corners typically point towards the centre of the building, as opposed to irrelevant corners. For every FAST feature $F_i$, a line $L_i$ is defined, passing through the feature and having the orientation of the image gradient at $F_i$. Then, a new feature $F_i'$ is computed as the point in $L_i$ closest to the centre of the candidate building region. For every feature $F_i'$, a 2D Gaussian distribution having as peak the score of $F_i$ is defined. A sum of these distributions is computed for every pixel in a candidate building region $R$ and the maximum value $V_R$ is identified, which indicates the strength of the corners in the region. The value $V_R$ is larger for several identified FAST corners pointing towards the centre of a candidate building region. A large number of corners with an orientation towards the centre of an image region indicates the existence of a building in the region. The probability $p_{FAST}$ for each region $R$ is equal to the value $V_R$ divided by the maximum $V_R$ value among all the candidate regions as shown in Eq. (3.17), assuming that the region with the highest value represents a true building. A clarification of how the probability $p_{FAST}$ is computed, is presented in Fig. 3.9. In this figure, $F_0$ and $F_1$ are two FAST features that correspond to building corners, while $F_2$ is an irrelevant FAST feature. Fig. 3.9 demonstrates how the original red locations of the FAST features are translated to the final green locations that are the closest possible points of the lines $L_i$ to the central

yellow point $C$.

$$p_{FAST} = \frac{V_R}{max_R(V_R)} \tag{3.17}$$

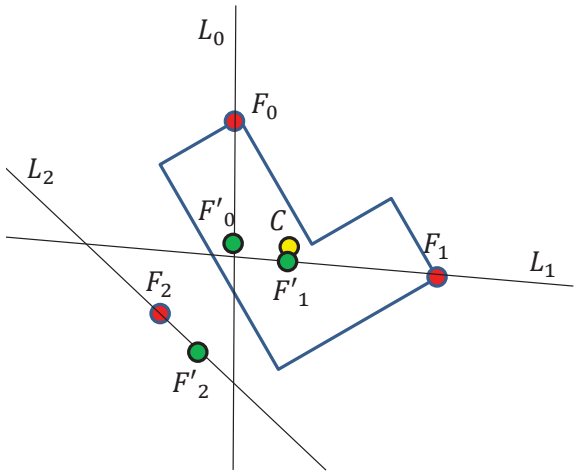



Figure 3.9: Illustration of the way $p_{FAST}$ is computed. The FAST corners are translated towards the centre of the region based on their gradient orientation.

### 3.2.4 Fusion of HOG, NDVI and FAST features

A way to take advantage of the information provided by each feature (HOG, NDVI and FAST) is to apply a fusion scheme that derives a combined knowledge for the existence of buildings. This combined knowledge can be expressed with the posterior probability that a region describes a building and is the result of the merging or fusion of the three probabilities $p_{HOG}$, $p_{NDVI}$ and $p_{FAST}$, which for shorter notation are referred from now on as $p_H$, $p_N$ and $p_F$ respectively. Several fusion schemes are examined based on the work in [176] and their performance on the building detection task is evaluated. These fusion schemes have the form of decision rules that define how the corresponding probabilities are combined in order to produce the posterior probability $p(B)$ of an image region to belong to the "building" class. Special care should be taken so as a normalisation factor is applied to ensure that the posterior probabilities of an image region to belong to the "building" and "non-building" classes add up to unity. This is achieved by the denominator terms in each corresponding decision rule. The tested decision rules are:

- *Min rule*: According to the min rule, the posterior probability of a class is equal to the minimum of the class probabilities computed by the different features. As a result, the posterior probability of an image region to describe a building is given by

$$p(B) = \frac{\min(p_H, p_N, p_F)}{\min(p_H, p_N, p_F) + \min(1-p_H, 1-p_N, 1-p_F)} \tag{3.18}$$

- *Max rule*: The max rule defines the posterior probability of a class to be equal to the maximum of the class probabilities computed by the different features. As a consequence, the posterior probability of an image region to belong to the "building" class is given by

$$p(B) = \frac{\max(p_H, p_N, p_F)}{\max(p_H, p_N, p_F) + \max(1 - p_H, 1 - p_N, 1 - p_F)} \tag{3.19}$$

- *Median rule*: The median rule states that the posterior probability of a class is equal to the median of the class probabilities computed from the different features. The median of a set of values is the middle value, when the values are placed in a sorted order. In this case, the median of the observed probabilities is equal to the probability that is not the minimum or the maximum of the three observed probabilities. Eq. (3.20) clarifies how the posterior probability in the case of the median rule is computed, given that the operation "*med*" refers to the computation of the median value.

$$p(B) = \frac{med(p_H, p_N, p_F)}{med(p_H, p_N, p_F) + med(1 - p_H, 1 - p_N, 1 - p_F)} \tag{3.20}$$

- *Product rule*: According to the product rule, the posterior probability of a class is equal to the product of the class probabilities computed from the different features. This means that the posterior probability of an image region to belong to the "building" class is given by

$$p(B) = \frac{p_H p_N p_F}{p_H p_N p_F + (1 - p_H)(1 - p_N)(1 - p_F)} \tag{3.21}$$

- *Sum rule*: The sum rule states that the probability of a class is equal to the sum of the class probabilities computed from the different features. In the building detection framework, the posterior probability of an image region to describe a building is given by

$$p(B) = \frac{p_H + p_N + p_F}{p_H + p_N + p_F + (1 - p_H) + (1 - p_N) + (1 - p_F)} = \frac{p_H + p_N + p_F}{3} \tag{3.22}$$

The probabilities $p_H$, $p_N$ and $p_F$ of all the initial candidate building regions are fused with a selected decision rule and posterior probabilities for each candidate building region are computed. Afterwards, thresholding is performed, where candidate building regions with a posterior probability equal or higher than 0.5 are accepted, while the others are discarded.

### 3.2.5 Evaluation

In this section, the decision rules that affect the probabilistic fusion of the HOG, NDVI and FAST features are evaluated. Furthermore, the effect of the NDVI and FAST features on the performance of the proposed improved HOG building detector on the Athens dataset is evaluated. Finally, a comparison between the improved HOG building detector and the HOG building detector, proposed in Section 3.1, is made.

The performance of the proposed improved HOG building detector, when the decision rules are applied, is presented in Table 3.3. The best result for each metric is shown in bold. The conclusion that can be drawn is that the sum rule achieves the highest performance on the building detection task. This is consistent with the conclusions in [176] that state that the sum rule is more robust to estimation errors than the other decision rules and therefore, it leads to more reliable decisions.

Table 3.3: Results, presented in the format: mean ± std, of our improved HOG building detector on the Athens dataset for the tested decision rules

| Decision rules | Recall | Precision | $F_1$-score |
|---|---|---|---|
| Min rule | 0.843 ± 0.061 | 0.594 ± 0.126 | 0.697 ± 0.073 |
| Max rule | 0.844 ± 0.061 | 0.598 ± 0.125 | 0.7 ± 0.073 |
| Median rule | 0.842 ± 0.06 | 0.597 ± 0.128 | 0.699 ± 0.075 |
| Product rule | 0.85 ± 0.059 | 0.602 ± 0.125 | 0.705 ± 0.074 |
| Sum rule | **0.853 ± 0.058** | **0.606 ± 0.12** | **0.709 ± 0.067** |

The purpose of the next set of experiments is to evaluate the effect of the NDVI mask and the FAST features on the performance of the improved HOG building detector. The proposed building detection algorithm is tested on the Athens test set and the metrics of total recall, precision and $F_1$-score are computed and presented in Table 3.4 for all combinations of the employed features. The optimal result for each metric is shown in bold.

Table 3.4: Performance of the improved HOG building detector on the Athens dataset for all feature combinations. The results are presented in the format: mean ± std.

| Features | Recall | Precision | $F_1$-score |
|---|---|---|---|
| HOG | 0.858 ± 0.085 | 0.577 ± 0.131 | 0.675 ± 0.069 |
| HOG+NDVI | **0.86 ± 0.058** | 0.545 ± 0.124 | 0.667 ± 0.071 |
| HOG+FAST | 0.829 ± 0.06 | **0.616 ± 0.117** | 0.707 ± 0.068 |
| HOG+NDVI+FAST | 0.853 ± 0.058 | 0.606 ± 0.12 | **0.709 ± 0.067** |

An analysis of Table 3.4 shows that the introduction of only the NDVI mask leads to a slight drop in the performance of the improved HOG building detector since the $F_1$-score decreases by almost 1.2%. The marginal increase in the metric of recall is attributed to the detection merging procedures that are employed after the computation of the initial detections. A decrease in the number of initial detections leads to changes in the distribution of the detections, which can lead to changes in the number and position of the

final detections (i.e. local maxima of distribution) after the employment of the mean-shift algorithm. The use of FAST and HOG features leads to a drop in the number of detected buildings, but a significant drop in the number of false detections that the improved HOG building detector produces is noticed as well. A comparison of the computed $F_1$-scores shows an increase by 4.7%, when the FAST features are introduced.

Finally, the employment of HOG, NDVI and FAST features increases the classification accuracy of the proposed algorithm by 5% with respect to using only the HOG features. The number of detected buildings is slightly reduced, but the number of false detections decreases significantly. By examining the results of Table 3.4, it can be assumed that the NDVI feature is not a robust building detection indicator, since the NDVI feature leads to a decrease in the precision of the proposed algorithm. On the one hand, this is true since the NDVI index is designed to detect vegetation and it cannot distinguish roads or ground from buildings. However, a more thorough analysis shows that the use of the NDVI feature retains the high number of detected buildings and prevents buildings with few FAST corners and low contrast with the background to be rejected by the proposed building detection methodology.

To further demonstrate the effectiveness of the proposed improved HOG building detector on the Athens test set, a comparison with the HOG building detector, defined in Section 3.1 and two other state-of-the-art building detection methodologies is performed. There are several reasons behind the selection of the building detection algorithms of Ilsever and Unsalan in [2] and Sirmacek and Unsalan in [3] to form a baseline for the performance assessment of our proposed improved HOG building detector. On the one hand, the algorithm of Ilsever and Unsalan follows a similar strategy with our proposed method, extracting the same type of features (i.e. HOG descriptors) but differing in the way these features are computed. By comparing our method with the algorithm in [2], we want to both point out the importance of selecting appropriate HOG parameter values (i.e. number of bins, kernel size, etc.) for the computation of the HOG descriptors and justify the optimisation procedure that we followed to come up with the proposed optimal HOG parameter values. It can be shown that sub-optimal HOG parameter values can significantly affect the performance of an object detector that is based on the HOG features.

On the other hand, the building detection method of Sirmacek and Unsalan in [3] proposes the probabilistic fusion of several features (i.e. corners and texture descriptors) to form a decision about the final location of buildings. In this way, our proposed method can be considered similar to theirs as it proposes the probabilistic fusion of HOG, NDVI and FAST features and thus the building detection method of Sirmacek and Unsalan consists an appropriate candidate for comparison with our method. Furthermore, we want to demonstrate the increased robustness of the HOG features, as opposed to corner features, to the effect of noise and strong illumination variations in a challenging dataset, such as the Athens satellite dataset. Noise can significantly affect the extraction of corner features

and pose difficulties in object detection methods that rely on low-level features. The HOG features, however, are invariant to illumination variations and they may comprise a better choice than corner features for accurate and robust object detection methodologies.

Before the comparison of our building detector with the other building detection methodologies, a brief description of the other methods follows. The building detector of Ilsever and Unsalan in [2], which for short notation is called Ilsever detector from now on, computes HOG features based on the panchromatic image only, discarding the multi-spectral information that was also provided. Furthermore, the centred mask was employed for the gradient computation and unsigned gradients (0°- 180°) were extracted. The histogram bin size was equal to 20. Gamma correction was applied and the HOG descriptor was normalised using the $l_1$-norm normalisation method. The authors demonstrated the effectiveness of their method, but they noticed the large number of false alarms that their method produces. To this end, they employed a shadow detection procedure that increased the precision of their algorithm at the cost of restricting their method to detect only tall buildings capable of casting detectable shadows. The same method was implemented in this thesis, but without employing the proposed shadow detection technique, in order to perform a fair comparison of the algorithms, avoiding any restrictions on the height of detected buildings.

Sirmacek and Unsalan in [3] proposed the fusion of Harris, FAST, Gradient-Magnitude-based Support Regions (GMSR) and Gabor filtering local features. Their method is shortly called Sirmacek fusion from now on. Harris and FAST are two well-known algorithms that extract corners from an image. The GMSR local features are based on the detection of regions, where the gradients are smooth and the Gabor features are capable of identifying textures. The authors perceived these features as observations of building presence and estimated their probability density function using a bivariate kernel function. Afterwards, they employed a weighted decision fusion rule to merge the feature distributions in a single final distribution. The modes (i.e. local maxima) of the final distribution were identified and perceived as centres of the candidate buildings. The code for this building detection method was provided by the authors and it is written in MATLAB [177].

To achieve a fair comparison among all the different building detection methodologies and obtain valid conclusions, care should be taken so that the performance metrics of all tested algorithms are computed in a similar way. This is needed because the output of Sirmacek fusion in the form of building centres is different from the building regions that the HOG-based methodologies output. As a result, in the case of Sirmacek fusion, rectangles of size $11 \times 11$ pixels span from the computed candidate building centres and a building is considered detected if it overlaps with any of these rectangles. This is the same as the HOG-based building detectors for the minimum searching windows. The performance of the tested building detection methodologies on the Athens dataset is summarised in Table 3.5. All values refer to the metric of $F_1$-score, unless otherwise stated. The optimal result for each metric is shown in bold.

Table 3.5: Comparison of the tested building detection methodologies on the Athens dataset.

| Image \ Method | Our HOG building detector | Ilsever detector [2] | Sirmacek fusion [3] | Our improved HOG building detector |
|---|---|---|---|---|
| area_1a_2006 | 0.554 | 0.216 | 0.533 | **0.625** |
| area_1b_2006 | 0.624 | 0.389 | 0.537 | **0.657** |
| area_1c_2006 | 0.718 | 0.522 | 0.6 | **0.768** |
| area_2a_2006 | 0.566 | 0.277 | 0.482 | **0.587** |
| area_2b_2006 | 0.521 | 0.187 | 0.388 | **0.552** |
| area_2c_2006 | 0.629 | 0.249 | 0.513 | **0.649** |
| area_1a_2007 | 0.685 | 0.218 | 0.586 | **0.747** |
| area_1b_2007 | 0.76 | 0.416 | 0.677 | **0.822** |
| area_1c_2007 | 0.781 | 0.547 | 0.634 | **0.791** |
| area_2a_2007 | 0.721 | 0.322 | 0.617 | **0.777** |
| area_2b_2007 | 0.66 | 0.223 | 0.416 | **0.667** |
| area_2c_2007 | 0.741 | 0.29 | 0.483 | **0.783** |
| area_1a_2009 | 0.613 | 0.228 | 0.385 | **0.658** |
| area_1c_2009 | 0.758 | 0.54 | 0.595 | **0.803** |
| area_2a_2009 | 0.656 | 0.309 | 0.525 | **0.692** |
| area_2b_2009 | 0.612 | 0.203 | 0.434 | **0.669** |
| area_2c_2009 | 0.697 | 0.274 | 0.534 | **0.719** |
| area_1a_2010 | 0.656 | 0.2 | 0.497 | **0.71** |
| area_1b_2010 | **0.713** | 0.392 | 0.469 | 0.701 |
| area_1c_2010 | 0.798 | 0.498 | 0.549 | **0.804** |
| area_2a_2010 | 0.678 | 0.282 | 0.432 | **0.722** |
| area_2b_2010 | 0.606 | 0.188 | 0.349 | **0.621** |
| area_2c_2010 | 0.67 | 0.247 | 0.505 | **0.704** |
| area_1a_2011 | 0.632 | 0.22 | 0.49 | **0.678** |
| area_1b_2011 | **0.72** | 0.399 | 0.498 | 0.695 |
| area_1c_2011 | 0.797 | 0.545 | 0.524 | **0.82** |
| area_2a_2011 | 0.682 | 0.276 | 0.386 | **0.709** |
| area_2b_2011 | 0.638 | 0.229 | 0.395 | **0.659** |
| area_2c_2011 | 0.68 | 0.57 | 0.435 | **0.7** |
| Mean recall ± std | 0.858 ± 0.085 | **0.954 ± 0.051** | 0.505 ± 0.107 | 0.853 ± 0.058 |
| Mean precision ± std | 0.577 ± 0.131 | 0.18 ± 0.153 | 0.48 ± 0.139 | **0.606 ± 0.12** |
| Mean $F_1$-score ± std | 0.675 ± 0.069 | 0.326 ± 0.124 | 0.499 ± 0.081 | **0.709 ± 0.067** |

An analysis of the performance of the building detection methodologies on the Athens dataset reveals that the proposed improved HOG building detector outperforms the other methodologies in almost every image of the dataset, with respect to the metric of $F_1$-score. Furthermore, the improved HOG building detector achieves the best total precision, while the proposed HOG building detector comes next. The Ilsever detector manages to recognise the largest number of buildings in the multi-spectral test images, however the inferior precision of the algorithm leads to suboptimal overall performance on the Athens dataset.

A comparison between our improved HOG building detector and the Ilsever detector shows the significance of an optimal tuning of the HOG parameters. Although both algorithms employ the same type of features to detect buildings, they follow a completely different parameter selection strategy and the results show a significant difference in the performance of the two algorithms on the test set.

Some conclusions can also be drawn with respect to the performance of the Sirmacek fusion in the Athens dataset. Although the employed corner and texture features are

quite discriminatory and they can significantly assist in the detection of buildings, they are severely affected by noise and low contrast. Several images of the Athens dataset exhibit cases of buildings that cannot be easily distinguished from the background or buildings that demonstrate strong texture variations in their rooftops. Such cases make the extraction of corners and texture features problematic and they can play a significant role in the severe performance degradation of the feature fusion algorithm.

A study of the standard deviation of the $F_1$-score in the Athens dataset shows that the improved HOG building detector achieves the best performance with a standard deviation of 0.067. The second best performance is achieved by the proposed HOG building detector with a standard deviation of 0.069, while the worst performance is achieved by the Ilsever detector with a standard deviation of 0.124. The standard deviation of a metric reflects the robustness of an algorithm to the conditions present in the tested satellite images with respect to the specific metric and therefore its evaluation is considered important. The conditions that are present in the Athens dataset and can significantly affect the results are the illumination variations, the sun location the time the image was captured, the spatial resolution and the spectral differences between the two types of multi-spectral images. The experiments show that the improved HOG building detector is the most robust of the tested building detection methodologies. The ability of a method to work with images captured by different satellites without suffering substantial performance degradation should be a vital attribute for a system that performs accurate and robust building detection and segmentation.

Finally, an illustration of the tested building detection methods on a part of a QuickBird and a WorldView-2 images of the test set is presented in Fig. 3.10. The purpose of the visual comparison is to clarify the differences between the building detection approaches and verify what was previously reported. The Ilsever detector identifies too many candidate building regions, covering most of the image areas, thus producing a large number of false alarms along with detecting most of the buildings. On the other hand, the Sirmacek fusion manages to identify only a limited number of buildings due to the noise present in the satellite images. This can be in a large degree noticed in the WorldView-2 image, where the low contrast of the buildings from their surroundings prevents the identification of correct corner and texture features that can lead to the detection of these buildings.

Both our proposed HOG building detectors allow for an excellent building detection rate, while they simultaneously keep the number of false alarms as small as possible. The visual comparison of the results from the proposed HOG building detectors shows that the improved HOG building detector reduces the number of false alarms of the HOG building detector by successfully employing the NDVI and FAST features.

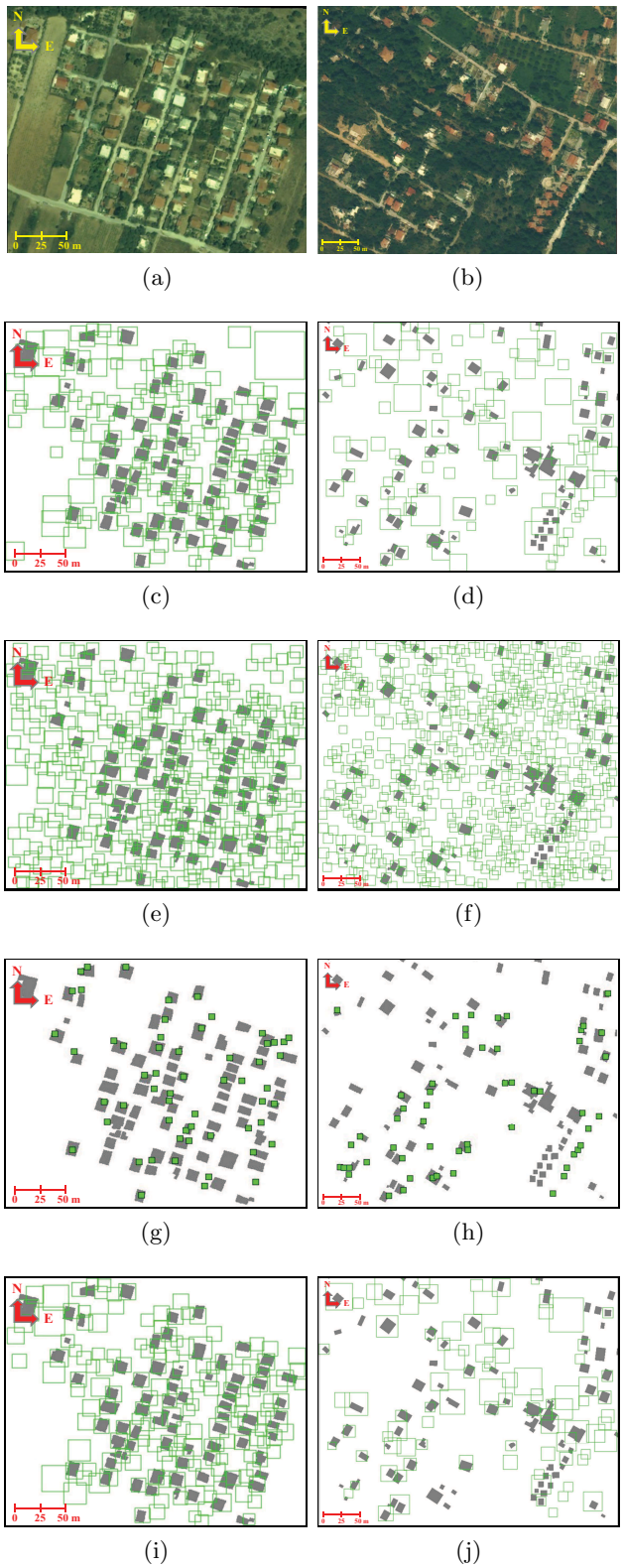


Figure 3.10: Detections of the tested building detection methods shown as green bounding boxes. (a), (b) Part of QuickBird and WorldView-2 images respectively. Results from (c), (d) proposed HOG building detector, (e), (f) Ilsever detector [2], (g), (h) Sirmacek fusion [3] and (i), (j) improved HOG building detector.

## 3.3 HOG-LBP building detector

To further improve the classification performance of a HOG-based building detector, a study of the effect of the LBP features on the building detection accuracy, when combined with the HOG descriptors, is performed. LBP is a successful texture descriptor that is used in several computer vision applications to solve the task of object detection. It has been shown that when LBP features are used in conjunction with HOG features, their effect is complementary, thus allowing for the development of a robust and accurate object recognition algorithm [178, 179].

A brief overview of the LBP features is given next, along with an analysis of how the HOG-LBP descriptors are formed and what type of LBP features are examined. Experiments are presented that justify the selection of the optimal type of LBP features and the performance of the HOG-LBP building detector on the Athens test set is analysed and compared with the performance of the improved HOG building detector.

### 3.3.1 Local binary patterns

Before the employment of the LBP features for the building detection task, a brief introduction of the LBP features and the way they are computed is necessary. The basic LBP feature was introduced by Ojala in [70] under the assumption that a texture has locally two complementary aspects, a pattern and its strength. The basic LBP was later extended by Ojala in [71] and the generic LBP feature that extends the neighbourhood of the LBP operation to sizes larger than $3 \times 3$ pixels was proposed.

The LBP feature is computed in a small (usually $3 \times 3$ pixels) neighbourhood and it describes the relative intensity difference between the central pixel and its neighbouring pixels. The comparison of the pixels is thresholded, meaning that if the central pixel $p_C$ has a larger intensity than a tested neighbouring pixel $p_N$, the result is 0, otherwise 1 as it is shown below:

$$d(p_N - p_C) = \begin{cases} 1, & \text{if } p_N \geq p_C \\ 0, & \text{if } p_N < p_C \end{cases} \tag{3.23}$$

The main difference between the basic and the generic LBP feature, except the size of the neighbourhood, lies in the fact that the neighbouring pixels that are chosen for comparison with the central pixel lie in a circle and are taken in circular order. Each central pixel $p_C$ is compared with all its neighbours lying in the radius of the chosen neighbourhood and their thresholded differences, computed by Eq. (3.23), are weighted by powers of two and then summed up. Eq. (3.24) clarifies how the final LBP value for a tested pixel $p_C$, given its neighbours $p_i$ for $i = 0...P-1$ is computed. Furthermore, an illustration of a

$3 \times 3$ pixel neighbourhood and the way the order of neighbours is considered for the LBP feature computation is presented in Fig. 3.11.

$$LBP(p_C) = \sum_{i=0}^{P-1} d(p_i - p_C)2^i \tag{3.24}$$

The circular order of the selection of a pixel's neighbours for comparison is the reason for the rotation invariance that the LBP features exhibit. The result of the LBP operator can also be viewed as a $P$-bit code, where the $i$-th bit is the thresholded result of the comparison between the central pixel and its $i$-th neighbour, when the neighbours are taken in a circular order. The LBP value of a pixel is then added to a histogram of $2^P$ bins. This histogram is the final LBP descriptor that is used for classification. Several LBP variants were introduced in the next years in an attempt to reduce the large size of the generic LBP descriptor and improve its classification abilities [180, 181, 182, 183, 184]. Some of these LBP variants are analysed in the next section, as their effect on the performance of the proposed HOG-LBP building detector is evaluated.

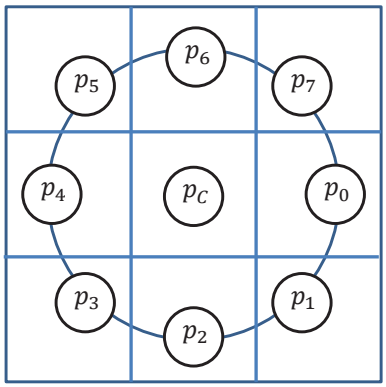



Figure 3.11: The circular order of neighbouring pixels in a $3 \times 3$ LBP operator.

### 3.3.2 HOG-LBP feature extraction

In the proposed HOG-LBP building detector, the extraction of the LBP descriptors follows a similar strategy with the extraction of the HOG descriptors. This means that LBP features are extracted for each multi-spectral channel separately and then concatenated into a large LBP histogram. Furthermore, the LBP features remain unnormalised, as is the case with the HOG features. The multi-spectral channels are not pre-processed with any filter, in contract with the computation of the HOG descriptors, where a sharpening filter is applied as a pre-processing step.

However, two significant differentiations are introduced to the proposed HOG-LBP detector that were not present in the previous HOG-based detectors. These differentiations concern the way the HOG and LBP features are computed and, as we demonstrate, im-

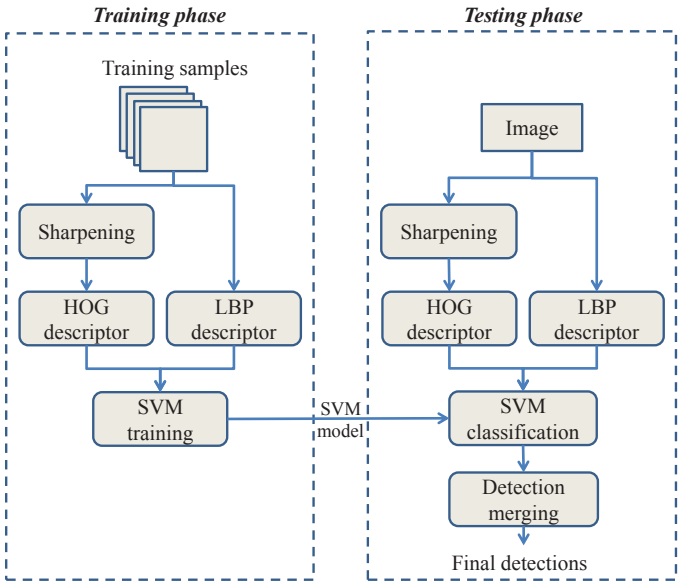



Figure 3.12: Block diagram of the proposed HOG-LBP building detector.

prove the classification performance of the proposed HOG-LBP detector in the building detection task. The first differentiation concerns the computation of the HOG descriptors in an additional colour channel, namely the red-difference chroma component (Cr) of the YCbCr colour space, while the second differentiation concerns the employment of a new rectangular block with two more overlapping cells for the computation of the HOG and LBP descriptors.

The reason behind the addition of the Cr channel lies in the fact that the Cr channel holds information about the presence and strength of the red colour in an image. Since the red colour usually appears in building rooftops, the Cr colour channel holds valuable information for the identification of buildings. Moreover, although overlapping cells add the same information to the descriptors more than once, they improve the robustness of a classifier since the same information counts differently, when processed on the different cells. The proposed HOG-LBP building detector is presented in Fig. 3.12 and it exhibits the same philosophy with the HOG building detector presented in Section 3.1.1.

Several LBP variants are implemented and tested in order to identify the variant that exhibits the best discrimination ability for the building detection task, when combined with the HOG features:

- *Classical LBP:* This is the basic LBP feature developed in [70, 71] and computed in a $3 \times 3$ neighbourhood of pixels. The pixels in the neighbourhood are thresholded by the value of the central pixel, multiplied by powers of two and then summed to form the label of the central pixel. Afterwards, the pixel labels are employed in the computation of the classical LBP histograms. Eq. (3.24) defines the way the classical LBP features are computed.

- *Uniform LBP:* This LBP variant was formed based on the work in [180]. In their study, the probabilities of pattern occurrences (i.e. 8-bit LBP values) were examined and it was deduced that the patterns with maximum 2 bit transitions (i.e. $0-1$ or $1-0$ transitions) appear more often than others in natural images [181]. As a result, they suggested that only the patterns that appear more often need to occupy separate bins in an LBP histogram, while all the other patterns are placed in the same bin. The uniform LBP is thus computed exactly like the classical LBP with the difference being in the limited number of histogram bins.

- *Center-Symmetric LBP (CS-LBP):* The CS-LBP feature was developed in an attempt to reduce the length of the classical LBP descriptor. The CS-LBP descriptor takes into account the differences between pairs of pixels opposed symmetrically with respect to the central pixel. As a result, the CS-LBP descriptor is closely related to the gradient operator. In this way, the CS-LBP descriptor takes advantage of both the LBP descriptor and a gradient based feature [182]. The CS-LBP feature for a central pixel $p_C$, given its $P$ neighbours can be computed using the following equation:

$$\text{CS-LBP}(p_C) = \sum_{i=0}^{(P/2)-1} d(p_i - p_{i+(P/2)})2^i, \text{ where } d(x) = \begin{cases} 1, x > T \\ 0, \text{otherwise} \end{cases} \tag{3.25}$$

  The threshold $T$ is employed to improve the robustness of the CS-LBP descriptor on flat regions, where the intensity difference between neighbouring pixels is really small. A small value is chosen for the threshold $T$, which is usually set equal to 1% of the maximum intensity value a pixel can get.

- *Improved Directional LBP (ID-LBP):* The ID-LBP was proposed in [183] as an improvement of the Directional LBP feature [184]. The ID-LBP descriptor is computed in a $3 \times 3$ pixel neighbourhood and is based on the differences between the average value $m$ of the pixels in the neighbourhood and the values of the pairs of the pixels opposed symmetrically with respect to the central pixel.

$$\begin{aligned} \text{ID-LBP}(p_C) = \sum_{i=0}^{(P/2)-1} d_1(p_i, m)d_2(p_{i+4}, m)2^i, \text{ where} \\ d_1(a,b) = \begin{cases} 1, a \geq b \\ 0, \text{otherwise} \end{cases} \quad \text{and } d_2(a,b) = \begin{cases} 1, a < b \\ 0, \text{otherwise} \end{cases} \end{aligned} \tag{3.26}$$

The computed HOG and LBP descriptors are concatenated into a single HOG-LBP descriptor that is fed to a SVM classifier for training or classification. The output of the SVM classifier is a set of detections (i.e. candidate building regions), along with scores that

describe the confidence of the classifier in its predictions. These detections are overlapping and therefore a mean-shift algorithm and our proposed rectangle merging procedure, as defined in Section 3.1.1, are employed to merge overlapping detections. The output of these merging procedures constitutes the final output of the HOG-LBP building detector.

### 3.3.3 Evaluation

Experiments are initially conducted in order to determine which LBP variant achieves the greatest improvement in the performance of the proposed HOG-LBP building detector, when the HOG and LBP descriptors are concatenated. Each LBP variant leads to the computation of LBP descriptors of different size. Table 3.6 illustrates the number of histogram bins that a LBP descriptor computed in a cell of an image block and for a single multi-spectral channel consists of. The final size of the LBP descriptor is therefore computed based on not only the histogram size but also the number of multi-spectral channels, for which LBP features are computed and the number of cells an image block consists of. The results of the experimentation with the LBP variants are presented in Table 3.7, where the best results are shown in bold. The experimentation reveals that the best performance of the proposed HOG-LBP building detector on the test set is achieved with the classical LBP descriptors, followed by the slightly worse performance of the uniform LBP descriptors.

Table 3.6: LBP histogram size per cell for different types of LBP features.

| LBP type | Number of features |
|---|---|
| Classical LBP | 256 |
| Uniform LBP | 59 |
| CS-LBP | 16 |
| ID-LBP | 16 |

Table 3.7: Results, presented in the format: mean ± std, of the proposed HOG-LBP building detector on the Athens dataset for combinations of HOG and LBP variants.

| Feature combination | Recall | Precision | $F_1$-score |
|---|---|---|---|
| HOG + Classical LBP | 0.86 ± 0.057 | **0.61 ± 0.115** | **0.714 ± 0.077** |
| HOG + Uniform LBP | **0.866 ± 0.057** | 0.603 ± 0.12 | 0.711 ± 0.081 |
| HOG + CS-LBP | 0.864 ± 0.058 | 0.495 ± 0.129 | 0.629 ± 0.089 |
| HOG + ID-LBP | 0.848 ± 0.059 | 0.596 ± 0.12 | 0.7 ± 0.084 |

The next experiments concern the two differentiations of the HOG-LBP detector from the proposed HOG building detectors that were presented in the previous sections. As it has already been noted, the first differentiation has the form of the introduction of the Cr channel of the YCrCb colour space as an additional colour channel for the computation of the HOG descriptors. The Cr channel holds information about the presence and strength of the red colour in an image and thus it can be considered important for the discrimination of buildings. A similar, but slightly inferior performance is achieved, when the u channel

of the Luv colour space is employed. The u channel holds similar information with the Cr channel and therefore, it can be considered as an alternative to the Cr channel for the identification of buildings. A futile attempt to employ either the Cr or the u channel for the computation of the LBP descriptors was performed. In this case, a drop in the performance of the HOG-LBP building detector is observed and it can be attributed to the smooth and texture-less appearance of building rooftops. Therefore, the u and Cr colour channels cannot contribute to the differentiation of buildings from other objects in an urban environment, when employed for the computation of the LBP descriptors.

The second important differentiation comes in the form of the cell configuration in the image block, where the HOG and LBP descriptors are computed. As we have already noted, the optimal block and cell configuration that was used for the computation of the HOG descriptors and was chosen among other block and cell configurations was a rectangular block with 4 non-overlapping cells, as the one presented in Fig. 3.13(a). Further experimentation, however, reveals that a rectangular block with 6 overlapping blocks improves significantly the performance of a HOG-LBP building detector. The cell configuration of the proposed rectangular block consists of 4 non-overlapping cells, in the same fashion as the previously employed rectangular block, and two new cells that are placed horizontally and vertically to the rectangular block and overlap with the other cells and with each other. These new cells have twice the size of the other cells and half the size of the entire rectangular block and they are presented in Fig. 3.13(b). The cell configuration of the proposed rectangular block is therefore the concatenation of the cell configurations presented in Fig. 3.13(a) and 3.13(b).

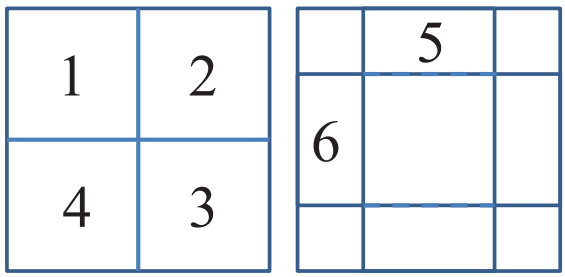



Figure 3.13: The previously employed cell configuration is shown in (a), while the new cells are shown in (b) and are added to the cell configuration of (a) to form the new cell configuration.

The results of applying the proposed differentiations on the HOG-LBP building detector are presented in Table 3.8. The conclusions that can be drawn from the results is that the addition of the Cr channel to the computation of the HOG part of the HOG-LBP descriptors leads to an improvement on the performance of the HOG-LBP building detector on the Athens test set by about 1.8% as far as $F_1$-score is concerned. This improvement is based on false alarm reduction that validates our claim that the Cr channel holds valuable information for the discrimination of buildings. The introduction of the new rectangular block for the computation of the HOG and LBP descriptors gives a strong boost to the per-

formance of the proposed HOG-LBP building detector. More specifically, the utilisation of the new rectangular block leads to an increase in the number of detected buildings by 12.5%, while it also manages to improve the precision of the HOG-LBP detector by almost 7%. As a result, the performance of the HOG-LBP building detector on the Athens test set is improved by almost 7.6% as the metric of $F_1$-score increases from 0.727 to 0.782, when the new rectangular block is employed. Although the proposed differentiations improve the accuracy of the HOG-LBP building detector, an increase in the standard deviations of all employed metrics is observed, meaning that the robustness of the new method is reduced. This increase can be attributed to the fact that the Cr colour channel is sensitive to the spectral resolution differences present in the satellite images of the Athens dataset.

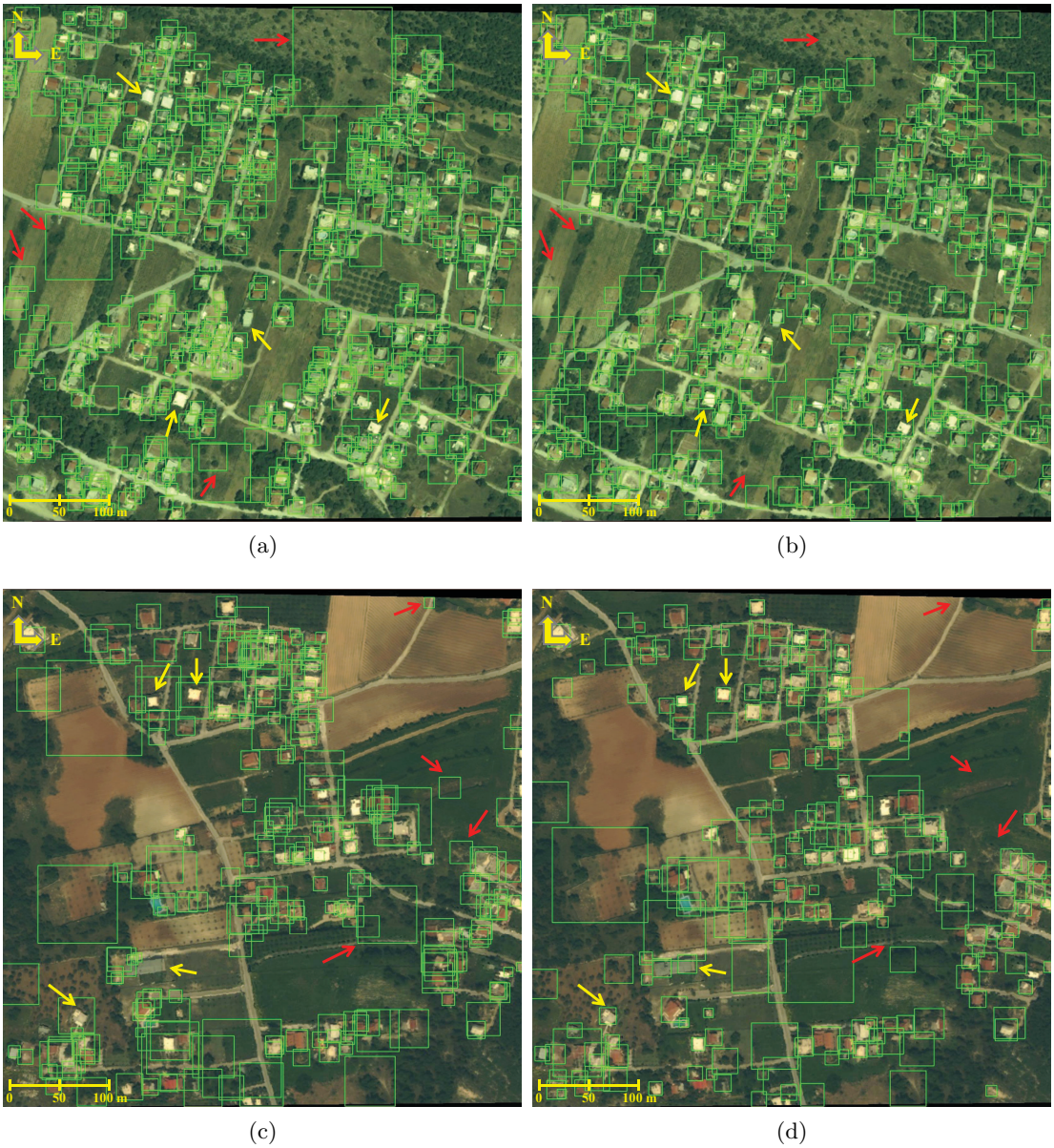
Table 3.8: Effect of proposed differentiations on the performance of the HOG-LBP building detector on the Athens dataset. The results are presented in the format: mean ± std.

| Differentiations | Recall | Precision | $F_1$-score |
|---|---|---|---|
| Initial HOG-LBP | 0.86 ± 0.057 | 0.61 ± 0.115 | 0.714 ± 0.077 |
| HOG-LBP + Cr colour | 0.856 ± 0.059 | 0.632 ± 0.116 | 0.727 ± 0.078 |
| HOG-LBP + Cr colour + New rectangular block | **0.963 ± 0.073** | **0.676 ± 0.155** | **0.782 ± 0.108** |

The total size of the HOG-LBP descriptor that is fed to the SVM classifier for training and testing can be computed based on the number of colour channels, the number of cells in an image block and the size of the HOG and LBP histograms. The HOG part of the HOG-LBP descriptor utilises 5 colour channels (R,G,B,NIR and Cr) and each image block consists of 6 cells, where the HOG histogram in each cell has 36 bins. This means that the total size of the HOG part of the HOG-LBP descriptor is equal to 1080. On the other hand, the LBP part of the HOG-LBP descriptor utilises 4 colour channels (R,G,B and NIR) and each image block consists of 6 cells, where a LBP histogram with 256 bins is computed for each cell. This means that the total size of the LBP part of the LBP descriptor is equal to 6144. As a result, the final HOG-LBP descriptor consists of 7224 features.

A comparison of the proposed HOG-LBP building detector and the improved HOG detector, presented in Section 3.2, is made in Fig. 3.14 for two satellite images of the Athens test set and illustrates the advantages of employing the LBP features and the two proposed differentiations. Examples of previously undetected buildings that are now detected and buildings, whose boundaries are more accurately delineated, are pointed out with yellow arrows on the images of Fig. 3.14. Furthermore, examples of false detections that are now discarded are shown by red arrows on the images. Some new false alarms are introduced by the new algorithm, but they are fewer than those removed as it is reported by the increase in the precision of the proposed HOG-LBP algorithm in the Athens test set. Moreover, the proposed HOG-LBP building detector achieves excellent building detection performance by managing to recognise almost every building in the Athens test set. As a result, the proposed HOG-LBP building detector performs better on the Athens test set

of multi-spectral images than any other building detector that has been tested so far.




(a) (b)

(c) (d)

Figure 3.14: Comparison of detections from (a), (c) improved HOG building detector and (b), (d) HOG-LBP building detector on two images of the test set. Buildings that are detected or better delineated by the HOG-LBP detector are pointed out by yellow arrows, while false alarms rejected by the HOG-LBP detector are pointed out by red arrows.

## 3.4 Enhanced HOG-LBP building detector

One of the most significant contributions of this thesis is the introduction of a new distance function for the training and classification of the HOG-LBP descriptors. The new distance function has the ability to suppress the effect of noise and thus it can lead to a more accurate and robust building detection methodology. We demonstrate the effect of the

new distance function in the building detection framework and evaluate and compare the performance of the proposed enhanced HOG-LBP building detector on the Athens dataset of multi-spectral images. The block diagram of the proposed enhanced HOG-LBP building detector is essentially the same as the block diagram of the HOG-LBP building detector, depicted in Fig. 3.12.

### 3.4.1 Cosine-based distance function

The cosine-based distance function was initially introduced by Fitch et al. in [185] in order to replace the $l_2$-norm function in the computation of displacement between video frames. It is known that the $l_2$-norm distance function has a significant flaw, namely, it is sensitive to outliers. The cosine-based distance function is proved to be more robust to noise and outliers, thus being a more suitable distance function than the $l_2$-norm. Similarly, the $l_1$-norm is robust to outliers as it does not raise the differences in the power of two and this is why the $l_1$-norm sometimes replaces the $l_2$-norm for the computation of distances between feature vectors. However, contrary to the $l_2$-norm distance function that offers stable and unique solutions, the $l_1$-norm distance function sometimes does not. In this thesis, we take advantage of the properties of the cosine-based distance function and we propose the novel introduction of the function to the SVM classifier for the computation of the distances between the HOG-LBP descriptors. In this way, we attempt to improve the robustness of a classifier in the presence of noise or outliers.

Given that $\mathbf{x_i}$ and $\mathbf{x_j}$ are two feature vectors of length $L$, an ideal distance function can be approximated by a limited number $P$ of sinusoidal terms, giving rise to the cosine-based distance function

$$d(\mathbf{x_i}, \mathbf{x_j}) \approx \sum_{p=1}^{P} \sum_{l=1}^{L} b_p (1 - \cos(a_p \pi (\mathbf{x_i}(l) - \mathbf{x_j}(l)))) \tag{3.27}$$

Because of the high computational cost of determining the coefficients $a_p$ and $b_p$ for more than one sinusoidal terms since several hundreds of classifiers need to be trained depending on required precision, we employ the cosine-based distance function with a single term ($P = 1$). Given also the constraint $\sum_{p=1}^{P} b_p = 1$ that bounds the values of the coefficients $b_p$, we end up with a distance function, where only the variable $a_1$ is unknown. This parameter, simply noted $\alpha$, represents the frequency of the cosine and it is a regularisation parameter that defines the behaviour of the cosine-based distance function in the presence of noise. Therefore, the parameter $\alpha$ can be optimised so that the cosine-based distance function, shown in Eq. (3.28), works best for the building detection task.

$$d(\mathbf{x_i}, \mathbf{x_j}) = \sum_{l=1}^{L} (1 - \cos(\alpha \pi (\mathbf{x_i}(l) - \mathbf{x_j}(l)))) \tag{3.28}$$

The cosine-based distance function for various values of the parameter $\alpha$ and the $l_1$-norm and $l_2$-norm distance functions are presented and compared in Fig. 3.15. One may observe that for small values of $\alpha$ (i.e. smaller or equal to 0.5) and differences in the range $[-1, 1]$, the cosine-based distance function has a parabolic shape similar to the shape of the $l_2$-norm curve. This means that for such values of $\alpha$ the cosine-based distance function behaves like the $l_2$-norm and the distance between two vectors increases as their absolute difference increases. Furthermore, for really small values of $\alpha$ (i.e. smaller or equal to 0.1), the behaviour of the cosine-based distance function resembles the linear behaviour of the $l_1$-norm. On the other hand, for larger values of $\alpha$, the cosine-based distance function presents a sinusoidal shape that is quite different from the $l_2$-norm curve. In such cases, large differences between vectors lead to smaller distances computed by the cosine-based distance function, while medium differences between vectors correspond to maximum distances between them. As a result, such values of $\alpha$ reduce the effect of large differences between vectors that can be attributed to noise or outliers. These statements hold because the cosine-based distance function for values in the range $[0, 1]$ has an influence function (i.e. derivative) that is equivalent to Andrew's M-Estimate [185, 186]

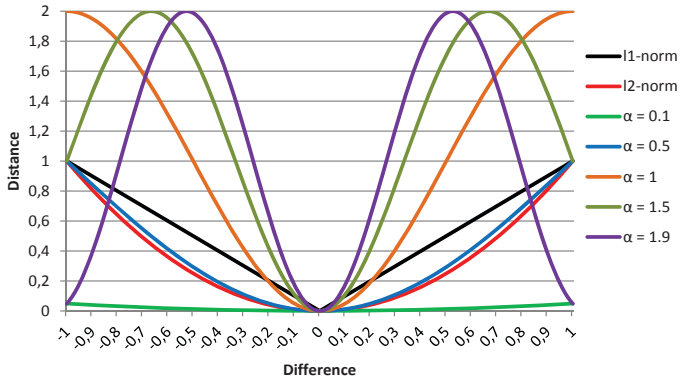


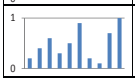

Figure 3.15: Comparison of the $l_1$-norm, $l_2$-norm and the cosine-based distance function for various values of parameter $\alpha$.

$$\psi(r) = \begin{cases} sin(\pi r) & \text{if } -1 \leq r \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.29}$$

The Andrew's M-Estimate belongs to a category of outlier rejection techniques, called re-descending m-estimators. The cosine-based distance function is considered outlier suppressor because it is not a monotonically increasing function as the difference between two vectors increases, but it redescends smoothly towards zero for large difference values. An example of outlier suppression is presented in Fig. 3.16, where the histograms in the second and third rows are compared with the histogram of the first row using the $l_1$-norm, $l_2$-norm and the cosine-based distance function for various values of $\alpha$. The only difference between the first row and the second row histograms is located in the last bin, which could

be attributed to noise, while the third row histogram is completely different than the first row histogram. As one may notice, the $l_2$-norm fails to capture the relations between the histograms, while this is not the case for the $l_1$-norm and the cosine-based distance function with increased values of $\alpha$.

Figure 3.16: Computation of distances between first row histogram and second and third row histograms using the $l_1$-norm, $l_2$-norm and the cosine-based distance function for various values of $\alpha$.

| Histograms | $l_1$-norm | $l_2$-norm | Cosine-based dissimilarity measure | | | | |
|---|---|---|---|---|---|---|---|
| | | | α = 0.1 | α = 0.5 | α = 1 | α = 1.5 | α = 1.9 |
|  | - | - | - | - | - | - | - |
|  | 0.8 | 0.64 | 0.0314 | 0.691 | 1.809 | 1.809 | 0.9372 |
|  | 2.2 | 0.64 | 0.0316 | 0.7714 | 2.8772 | 5.7598 | 8.1357 |

The cosine-based distance function, presented in Eq. (3.28), is employed as a novel special de-noising distance measure for the computation of distances between the HOG-LBP descriptors fed to the SVM classifier. One of the main advantages of the cosine-based distance function is the fact that it can be adapted to any problem at hand and suppress the correct amount of noise thanks to its parameter $\alpha$. This means that the cosine-based distance function can behave like $l_1$-norm ($\alpha \leq 0.1$), $l_2$-norm ($\alpha \in (0.1, 0.5]$), ideal outlier suppressor ($\alpha$ close to 1) and even as an oscillator ($\alpha \geq 1.5$). To take advantage of the power of the cosine-based distance function, the HOG-LBP descriptors should have values in the range $[0, 1]$. To achieve this, we divide a HOG-LBP descriptor with the number of pixels used for its computation. This is possible because the gradient magnitudes have values in the range $[0, 1]$ for pixel values in the range $[0, 1]$ and the LBP histograms are formed by counting the number of pixels belonging to the corresponding histogram bin. The optimal value of $\alpha$ is determined after experimentation in a validation set in order the cosine-based distance function suppresses the correct amount of noise in the dataset.

For the training of the HOG-LBP descriptors, the RBF kernel is chosen for the SVM classifier. However, instead of the $l_2$-norm, the cosine-based distance function is employed as shown in Eq. (3.30). We also experiment with a linear kernel for the SVM classifier that employs the $l_1$-norm. A SVM model is trained by determining the optimal parameters (i.e. cost of outliers and RBF kernel parameter $\gamma$) for which the cross-validation estimate of the test error is minimal. The trained SVM model is used for the classification of the HOG-LBP descriptors in the testing phase of the proposed enhanced HOG-LBP building detector.

$$K(x_i, x_j) = e^{-\gamma \sum_{l=1}^{L} (1 - \cos(\alpha\pi(x_i(l) - x_j(l))))}, \gamma > 0 \tag{3.30}$$

Furthermore, Platt scaling [175] is employed to transform the SVM scores that define the distances of the descriptors from the SVM separating hyperplane to probabilities. Platt identified that class-conditional densities between the margins are exponential and two exponentials suggest the use of a parametric form of a sigmoid function [175]. Given that $x_i$ is the SVM score for a descriptor, the probability of this descriptor $p_i$ was defined in Section 3.2.1 and it is also presented here for quicker reference

$$p_i = \frac{1}{1 + e^{Ax_i + B}} \tag{3.31}$$

The idea behind the transformation of the SVM scores to probabilities is two-fold; namely to enhance the meaning of the output of the SVM classifier as instead of being the unbounded distance of a sample from the SVM separating hyperplane, it now expresses the probability of a sample to belong to the positive ("building") class and also to use a new probabilistic threshold to differentiate between building and non-building regions, instead of the zero threshold that the standard SVM classifier assumes. These two additions give rise to the enhanced HOG-LBP building detector with or without Platt scaling.

### 3.4.2 Comparison of HOG-LBP building detector variants

The HOG-LBP building detector that was proposed in Section 3.3 and employs the $l_2$-norm function for the computation of distances between the HOG-LBP descriptors and the SVM scores for the descriptor classification is denoted simply as HOG-LBP detector (M1). The introduction of the cosine-based function for the computation of the distances between the HOG-LBP descriptors gives rise to the Enhanced HOG-LBP detector (M4). Moreover, the use of Platt scaling leads to two more HOG-LBP detector variants; the HOG-LBP detector with Platt scaling (M2) and the Enhanced HOG-LBP detector with Platt scaling (M5). Finally, we also acquire the HOG-LBP building detector that employs the $l_1$-norm and Platt scaling and is denoted as Linear HOG-LBP detector with Platt scaling (M3). These HOG-LBP building detector variants are tested in the satellite images of the Athens dataset and their performances are analysed and compared. All methodologies employ searching windows of multiple sizes (scales) ranging from $20 \times 20$ to $200 \times 200$ pixels. These sizes are adequate for the detection of buildings with sizes in the interval $50 - 8000m^2$, based on the resolution of satellite images. The displacement between two consecutive searching windows (i.e. stride) is equal to 5 pixels, while the ratio between two consecutive scales is selected to be equal to 1.1.

The employment of the Platt scaling does not only transform the scores of a SVM classifier from unbounded distances from the SVM separating hyperplane to probabilities, but also allows the definition of a new threshold to differentiate between positive and negative detections. The new threshold, which is equal to 0.5, implies that we accept only the detections that have larger probability to belong to the "building" class than to the "non-

building" class. This threshold value is considered as the optimal Bayes threshold value, given equal loss for false positives and false negatives according to [175] and has the advantage that it is not required to be determined manually by experimentation with the dataset.

The Platt parameters A and B, defined in Eq. (3.31), are computed using cross-validation on the training set. These parameters for the proposed HOG-LBP, Linear HOG-LBP and Enhanced HOG-LBP building detectors are presented in Table 3.9. A careful examination of the parameters reveals that the Platt parameter A should always be negative so as higher SVM scores are transformed to higher probabilities. On the other hand, the Platt parameter B can be either positive or negative. The three sigmoid functions that transform the SVM scores of the HOG-LBP, Linear HOG-LBP and Enhanced HOG-LBP building detectors to probabilities are depicted in Fig. 3.17. From the equation of the sigmoid function, it can be observed that the threshold of 0.5 corresponds to a SVM score of $-B/A$. This means that in the case of Platt scaling, the SVM separating hyperplane can be considered as transposed from 0 to 0.313, 0.338 and 0.152, in the case of the HOG-LBP, Linear HOG-LBP and Enhanced HOG-LBP detector respectively. As a result, detections that were previously accepted as positives are now discarded as negatives.

Table 3.9: Values of Platt parameters for the proposed HOG-LBP building detectors.

| Building detectors | A | B |
|---|---|---|
| HOG-LBP | -4.859 | 1.521 |
| Linear HOG-LBP | -3.747 | 1.268 |
| Enhanced HOG-LBP | -5.217 | 0.794 |

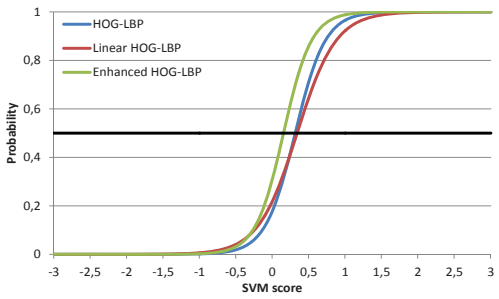
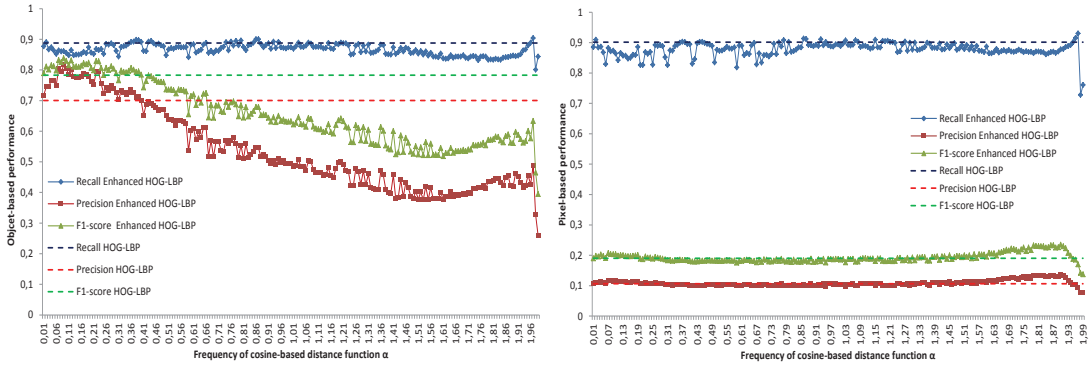



Figure 3.17: Computed sigmoid functions for transformation of SVM scores to probabilities in the cases of HOG-LBP, Linear HOG-LBP and Enhanced HOG-LBP detectors. The threshold of 0.5 is shown with a black line.

Experiments are conducted to determine the optimal value of the frequency $\alpha$ of the cosine-based distance function in order to suppress the correct amount of noise. The value of the parameter $\alpha$ varies from $[0.01, 1.99]$ with a step of 0.01. The performance of method M5 on the Athens "image" validation set, defined in Section 2.5, at the object and pixel level for all values of $\alpha$ is presented in Fig. 3.18. It can be concluded from the results that the performance of method M5 improves only for values of $\alpha$ smaller than 0.4. The optimal value of 0.09 for the parameter $\alpha$ that can be deduced from the object-based performance

of method M5 means that the amount of noise in the training set is not significant and slight variations in the computed HOG-LBP descriptors are essential for the differentiation between positives and negatives. Moreover, the optimal value of the parameter $\alpha$ reveals that the cosine-based distance function should behave similar to the $l_1$-norm in order to achieve improved robustness for the building detection task.




(a) (b)

Figure 3.18: Performance of the proposed Enhanced HOG-LBP building detector with Platt scaling (M5) on the Athens "image" validation set for different values of parameter $\alpha$. (a) Effect of frequency $\alpha$ on object-based performance. (b) Effect of frequency $\alpha$ on pixel-based performance.

The HOG-LBP building detector variants (M1-M5) are tested in the test images of the Athens dataset minus the images used for the validation of the optimal parameter $\alpha$ of the cosine-based distance function. The performance of the methods at object and pixel level are summarised in Table 3.11, while the average object-based recall, precision and $F_1$-score and their corresponding standard deviations in the test set are presented in Fig. 3.20 for a better comparison of the methods. The standard deviations are employed as indicators of the robustness of the tested methods in the images of the Athens dataset. Since these images vary significantly in illumination, sun position, spectral and spatial resolution, we believe it is important to compute such values in order to "predict" the performance of the methods on other datasets that could be tested on.

From the results in the Athens dataset, it can be deduced that the method M5 outperforms the other proposed HOG-LBP detectors. More specifically, the cosine-based distance function boosts the performance of the proposed M4 method by 6.5% with respect to the measure of $F_1$-score. An additional 4.1% increase in $F_1$-score is achieved by Platt scaling. Although a slight drop in recall is observed, when the cosine-based distance function and Platt scaling are employed, there is a significant increase in precision that leads to an overall better performance of method M5. A comparison between the method M3 and the HOG-LBP detector variants that employ the other kernels (M2 and M5) reveals that although the $l_1$-norm is more robust to outliers than the $l_2$-norm, its performance is inferior with respect to the other two detectors. This can be attributed to the fact that the M2 and M5 methods employ the RBF kernel that allows a higher-dimensionality mapping of the input feature space. A higher-dimensional space usually allows the classes to be well

separable and thus an optimal solution can be more easily identified and obtained. On the other hand, the method M3 with the $l_1$-norm presents limited classification abilities due to the difficulty in achieving a linear separation between the "building" and "non-building" classes in the input feature space.

The standard deviations of the object-based metrics verify the results of the comparison between the HOG-LBP detector variants. More specifically, the method M3 achieves the lowest standard deviation as far as $F_1$-score is concerned. This means that the method M3 achieves the highest robustness in the challenging Athens dataset, where intensity and spectral and spatial resolution variations exist. However, its inferior performance with respect to the other HOG-LBP detector variants renders it a sub-optimal classifier. The method M5 achieves the second lowest standard deviation in $F_1$-score, outperforming the other algorithms by at least 8.4%. As a result, the cosine-based distance function that is employed in the method M5 enables the suppression of the effect of noise and outliers without sacrificing the high-dimensionality mapping. A more detailed overview of the performance of the HOG-LBP variants in the test set of the Athens dataset with respect to the metric of $F_1$-score is presented in Table 3.10, where the best values for each multi-spectral image is shown in bold. As one may notice the method M5 outperforms the other HOG-LBP detector variants (M1-M4) in 16 out of 24 satellite images as far as $F_1$-score is concerned.

Table 3.10: Comparison of the HOG-LBP building detector variants on the Athens dataset.

| Method / Image | M1 | M2 | M3 | M4 | M5 |
|---|---|---|---|---|---|
| area_1a_2006 | 0.617 | 0.773 | 0.678 | 0.669 | **0.823** |
| area_1b_2006 | 0.701 | 0.78 | 0.751 | 0.773 | **0.824** |
| area_1c_2006 | 0.768 | 0.837 | 0.806 | 0.826 | **0.908** |
| area_2a_2006 | 0.568 | 0.743 | 0.667 | 0.67 | **0.761** |
| area_2b_2006 | 0.584 | 0.678 | 0.658 | 0.659 | **0.767** |
| area_2c_2006 | 0.692 | 0.787 | 0.74 | 0.738 | **0.849** |
| area_1a_2007 | **0.931** | 0.901 | 0.831 | 0.897 | 0.852 |
| area_1b_2007 | 0.964 | **0.966** | 0.89 | 0.963 | 0.961 |
| area_1c_2007 | **0.95** | 0.947 | 0.828 | 0.932 | 0.93 |
| area_2a_2007 | **0.918** | 0.877 | 0.798 | 0.798 | 0.794 |
| area_1c_2009 | 0.809 | 0.864 | 0.839 | 0.879 | **0.92** |
| area_2a_2009 | 0.766 | 0.822 | 0.739 | 0.787 | **0.882** |
| area_2b_2009 | 0.696 | 0.786 | 0.709 | 0.771 | **0.827** |
| area_2c_2009 | 0.766 | 0.845 | 0.792 | 0.836 | **0.892** |
| area_1c_2010 | 0.833 | 0.849 | 0.816 | **0.92** | 0.903 |
| area_2a_2010 | 0.732 | 0.763 | 0.723 | 0.798 | **0.84** |
| area_2b_2010 | 0.644 | 0.756 | 0.651 | 0.751 | **0.792** |
| area_2c_2010 | 0.731 | 0.791 | 0.751 | **0.857** | 0.831 |
| area_1a_2011 | 0.879 | 0.911 | 0.697 | 0.945 | **0.959** |
| area_1b_2011 | 0.872 | 0.902 | 0.76 | **0.933** | 0.926 |
| area_1c_2011 | 0.896 | 0.918 | 0.811 | 0.937 | **0.941** |
| area_2a_2011 | 0.816 | 0.902 | 0.756 | 0.887 | **0.922** |
| area_2b_2011 | 0.765 | **0.867** | 0.669 | 0.839 | 0.865 |
| area_2c_2011 | 0.873 | 0.909 | 0.806 | 0.907 | **0.932** |
| Mean $F_1$-score ± std | 0.782±0.108 | 0.835±0.083 | 0.757±0.064 | 0.833±0.09 | **0.865±0.076** |

As far as pixel-based performance is concerned, the method M3 achieves significantly

higher performance than the other HOG-LBP detector variants, which is attributed to the higher pixel-based precision, although lower pixel-based recall. A careful observation of the results reveals that the higher pixel-based precision is due to more precise building boundary identification, which can be attributed to the robustness of the $l_1$-norm as a distance function. Furthermore, the method M5 achieves similar performance to the method M2 and slightly better performance than the cases, where no Platt scaling is introduced (M1 and M4). Consequently, the additions of the cosine-based distance function and Platt scaling are successful in their attempt to remove noise and scale the SVM outputs respectively in a way to reduce false alarms and thus increase precision in the building detection task. These conclusions are further backed up by the comparison of the HOG-LBP detector variants in Fig. 3.21(b)-(f) that depict problematic cases of buildings with spectral variations in their rooftops and buildings occluded by vegetation. The method M5 manages to reject most of the false alarms present at road segments, which the other HOG-LBP detector variants accept, while it also identifies building boundaries very precisely. The method M3 achieves the highest precision in the identification of boundaries around buildings with respect to the other HOG-LBP detector variants. However, the method M3 fails to identify a problematic case of a building partially covered by vegetation (see lowest red circle in Fig. 3.21(d)). On the contrary, all other HOG-LBP detector variants (M1, M2, M4 and M5) are capable of identifying problematic cases of buildings with spectral variations in their rooftops (see yellow circles in Fig. 3.21(b)-(f)) or buildings partially covered by vegetation (see red circles in Fig. 3.21(b)-(f)).

### 3.4.3 Comparison with other techniques

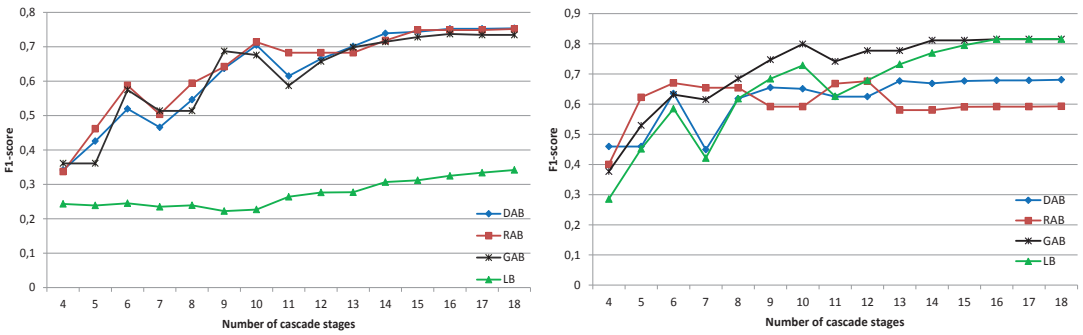
To further demonstrate the excellent performance of the proposed Enhanced HOG-LBP building detector with Platt scaling, we implement state-of-the-art object-based building detection methodologies and evaluate and compare their performance on the satellite images of the Athens test set. These building detection methods employ various features for the object detection task and they are listed below:

- *Cascade classifier with HAAR features (M6):* A cascade of boosted classifiers is a structure of several simple classifiers, whose votes are aggregated to increase the performance of a detector. Such a cascade classifier that employs HAAR features was initially proposed in [69] and was later improved in [187]. The cascade classifier is trained and used for the detection of buildings in the Athens dataset. Four types of boosting techniques are tested, namely the Discrete Adaboost [188], the Real Adaboost [189, 190], the Logitboost [190] and the Gentle Adaboost [190]. The main differences among the boosting techniques lie in the fact that the Discrete Adaboost classifies a sample based on a weighted majority vote among the classifiers, the Real Adaboost predicts a real-valued probability that a given sample belongs to a class, the Logitboost uses an additive logistic model to produce good regression fits, while

the Gentle Adaboost provides a more reliable and stable ensemble than the Real Adaboost by putting less emphasis on outliers. For a more thorough analysis of the boosting techniques, a reader is urged to read chapter 2 of [191]. The cascade HAAR classifier is tested for increasing number of cascade stages in order to identify which parameter configuration achieves the best performance on the building detection task. The results are presented in Fig. 3.19(a), where one can deduce that the Discrete Adaboost classifier with 16 cascade stages achieves slightly better results than the Real Adaboost with 16 stages and much better results than other parameter configurations.

- *Cascade classifier with LBP features (M7):* This classifier employs the same idea as the previously presented cascade classifier, with the only difference residing in the use of LBP features instead of HAAR features. Similar to the HAAR cascade classifier, experiments are conducted for all types of boosting techniques and several cascade stages in order to identify which parameter configuration achieves the best performance on the building detection task. The results are presented in Fig. 3.19(b), where it can be deduced that the Logitboost classifier with 17 cascade stages achieves slightly better results than the Gentle Adaboost with 17 stages and much better results than other parameter configurations.

- *Sirmacek fusion (M8) [3]:* A building detection methodology by fusing various distinctive features was proposed in [3]. The authors employed Harris and FAST corner features, GMSR regions and Gabor-filtering local features to identify building centres. Unlike the other algorithms, this is the only method that cannot detect building candidate regions but only building centres. Therefore, the performance of this methodology can only be evaluated at the object and not at the pixel level.




(a) (b)

Figure 3.19: Presentation of the achieved $F_1$-score for the tested configurations of cascade adaboost classifiers on the Athens "image" validation set. (a) Cascade HAAR classifier. (b) Cascade LBP classifier. DAB: Discrete Adaboost, RAB: Real Adaboost, LB: Logitboost, GAB: Gentle Adaboost

The performances of the previously described methodologies (M6-M8) are evaluated at the object and pixel level in the Athens dataset and the average recall, precision and $F_1$-score, along with their corresponding standard deviations, are summarised in Table 3.11.

Furthermore, the object-based metrics of recall, precision and $F_1$-score are visualised in Fig. 3.20 for a better comparison with the other tested building detectors.

The comparison of the proposed Enhanced HOG-LBP building detector with Platt scaling (M5) with methods M6-M8 shows the superiority of our method M5 over the other methods. More specifically, the proposed method M5 achieves an increase in object-based $F_1$-score by 10.3% over the cascade classifier with LBP features (M7) and by more than 26% over the cascade classifier with HAAR features (M6) and the Sirmacek fusion (M8). The improved performance of method M5 over the methods M6-M8 is translated to a higher number of detected buildings and a lower number of false detections. The method M5 achieves comparable standard deviation to the method M5 with respect to the object-based metric of $F_1$-score and slightly increased with respect to the method M6 that exhibits a standard deviation of 0.067. This means that the methods M4, M5 and M6 show the greatest robustness among the other tested methods in the building detection task with respect to illumination variations, spectral differences and spatial resolution changes present in the images of the Athens dataset.

A comparison of the pixel-based performance of methods M5-M8 reveals that the method M6 achieves the best precision and $F_1$-score. However, the metric of recall is much lower than that of the other methodologies, making it an inferior building detector in the Athens dataset. Similarly, the pixel-based metric of recall of method M7 is lower than that of our proposed method M5, meaning that the method M5 is able to correctly identify most of the pixels that belong to the "building" class. The reason behind the low pixel-based precision of the tested methodologies is attributed to the fact that these methods detect regions where buildings are likely to exist and they cannot recognise and extract building boundaries.

A study of the tested methods M5-M8 and their corresponding results in the building detection task reveals their weaknesses. The method M6 employs HAAR features that sum up intensities in rectangular regions and thus, they are sensitive to illumination changes, affecting significantly the results of building detection in images being under various lightning conditions. The method M8 relies on the fusion of corner, texture and gradient features. The images of the Athens dataset suffer from noise and low contrast of buildings to background and therefore, this method fails to extract enough corners to successfully identify building centres. Finally, the method M7 is successful in discriminating buildings from background, but its performance is still outmatched by the complementary use of both HOG and LBP features employed by the proposed method M5.

Fig. 3.21 compares the detections of the tested methodologies in a satellite image, where buildings with spectral variations in their rooftops (i.e. buildings within the yellow circles) and buildings partially occluded by vegetation (i.e. buildings within the red circles) exist. Results reveal that the cascade classifiers (M6 and M7) have no problem detecting such problematic cases of buildings. Unfortunately, the cascade classifiers seem to miss a few

Table 3.11: Performance of the tested methodologies at object and pixel level in the Athens dataset. The results are presented in the format: mean ± std. Best values are shown in bold.

| Method | Object-based | | | Pixel-based | | |
|---|---|---|---|---|---|---|
| | Recall | Precision | $F_1$-score | Recall | Precision | $F_1$-score |
| M1 | **0.963±0.073** | 0.676±0.155 | 0.782±0.108 | **0.966±0.023** | 0.118±0.044 | 0.207±0.069 |
| M2 | 0.94±0.108 | 0.771±0.122 | 0.835±0.083 | 0.945±0.041 | 0.125±0.05 | 0.217±0.075 |
| M3 | 0.886±0.06 | 0.676±0.125 | 0.757±0.064 | 0.842±0.062 | **0.231±0.078** | **0.353±0.088** |
| M4 | 0.93±0.095 | 0.772±0.135 | 0.833±0.09 | 0.938±0.064 | 0.135±0.05 | 0.232±0.074 |
| M5 | 0.934±0.114 | **0.823±0.102** | **0.865±0.076** | 0.945±0.066 | 0.135±0.055 | 0.23±0.079 |
| M6 | 0.755±0.089 | 0.637±0.102 | 0.685±0.075 | 0.745±0.076 | 0.218±0.052 | 0.332±0.056 |
| M7 | 0.9±0.043 | 0.701±0.1 | 0.784±0.067 | 0.881±0.044 | 0.136±0.048 | 0.231±0.069 |
| M8 | 0.516±0.091 | 0.53±0.15 | 0.5±0.08 | - | - | - |

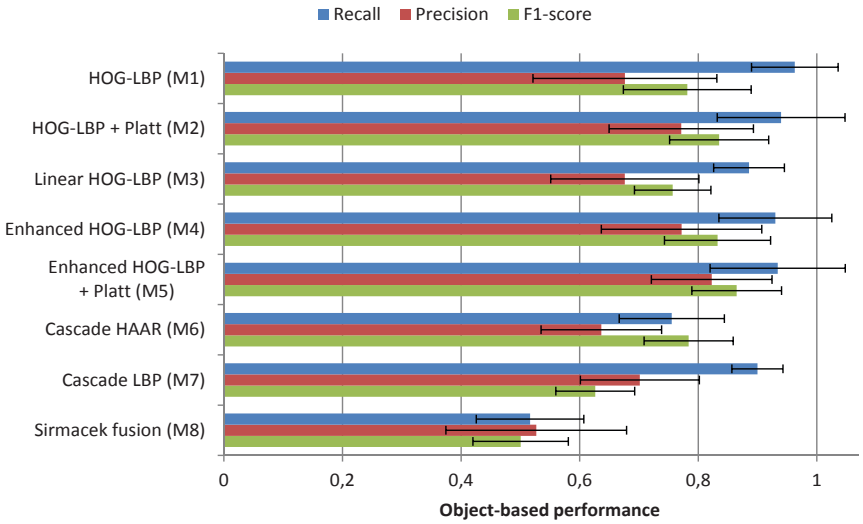



Figure 3.20: Average values of recall, precision and $F_1$-score and their corresponding standard deviations for the tested methodologies (M1-M8) at the object level in the Athens dataset.

"regular" buildings or inaccurately identify them (i.e. significant portion of a detection is covered by background). Sirmacek fusion (M8), on the other hand, fails in identifying one of the buildings with spectral variations in its rooftop (see lowest yellow circle in Fig. 3.21(i)), while it also fails to detect several "regular" buildings. Finally, all HOG-LBP variants have no problem detecting problematic cases of buildings. However, the method M5 produces the fewest false detections, thus being the optimal building detector among those tested.

## 3.5 Towards a pixel-based building detector

Although the Enhanced HOG-LBP building detector with Platt scaling achieves accurate and robust performance on the multi-spectral images of the Athens dataset at the object level, its performance at the pixel level is inadequate for pixel-based building detection. This is attributed to the fact that the proposed detector is an object-based classifier that extracts features from image regions and classifies these regions based on the likelihood

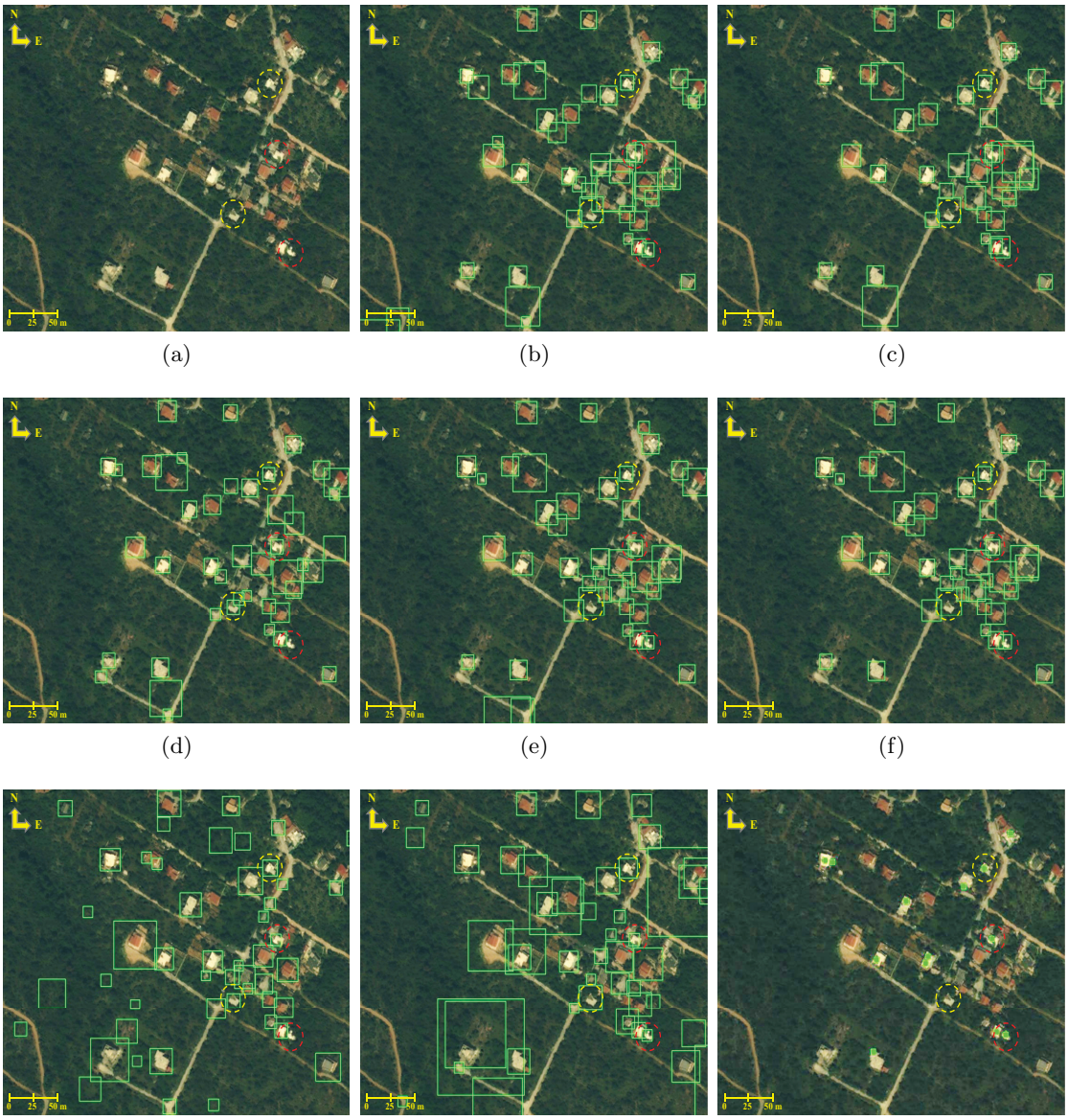


Figure 3.21: Comparison of the tested methodologies in a satellite image depicting buildings with spectral variations in their rooftops (yellow circles) and buildings covered partially by vegetation (red circles). The initial image is presented in (a), while the results of algorithms M1-M8 are shown in (b)-(i) respectively.

that buildings exist in the regions. To improve the pixel-based performance of the proposed enhanced HOG-LBP building detector and generally any object-based building detector, a methodology is proposed that consists of a set of region refinement processes. These processes employ the results of the object-based building detector and by operating at the pixel level, attempt to identify and delineate building boundaries.

The proposed pixel-based building detection methodology can be split in two modules: the object-based building detector and the region refinement procedure. The object-based building detector is concerned with the feature extraction, model training and classification

of images by detecting candidate image regions, where buildings are likely to exist. In our case, we employ the enhanced HOG-LBP detector with Platt scaling (M5), defined in Section 3.4, as it is found to outperform the other tested object-based building detectors. The region refinement procedure that follows is not limited to the specific type of classifier but any type of classifier, capable of producing candidate building regions with scores that describe the confidence of the classifier in its predictions, can be employed.

The purpose of the region refinement procedure is the successful transition from the candidate rectangular building regions, that the HOG-LBP detector produces, to the accurate delineation of building boundaries. This procedure should be able to remove background pixels around the buildings, while maintaining most of the pixels that belong to the "building" class. By employing the region refinement procedure, we want to avoid the processing and segmentation of the entire image and concentrate only on the image regions detected by the building detector. However, we notice that except for the local information that exists in each of the detected image regions, global information is also necessary in order to discard large regions that belong to fields or roads.

As a result, the region refinement procedure initially segments the image to homogeneous regions based on spectral information by employing an unsupervised pixel clustering technique. Afterwards, pixels that belong to the "vegetation" or "water" classes are identified by employing a well-known vegetation and a novel water index respectively. By identifying vegetation and water, we can at a later stage discard spurious regions that contain a significant portion of "vegetation" or "water" pixels. Finally, we take advantage of the output of the HOG-LBP detector to deduce the most probable to represent buildings image regions. These initial building candidates are further processed to form the final building candidates. These processing stages are thoroughly analysed and examined below. The proposed pixel-based building detector that is briefly described above is presented in Fig. 3.22, where the two modules along with their in-between interactions, are illustrated. The training and testing phases of the algorithm are also depicted.

### 3.5.1 Image segmentation

The proposed Enhanced HOG-LBP building detector cannot recognise building boundaries, but it produces a set of candidate rectangular image regions, where buildings are likely to exist. To make the transition from the extraction of these rectangular detections to the final delineation of building boundaries, an accurate image segmentation is necessary. The purpose of the proposed image segmentation procedure is the partitioning of an image to a set of connected-component regions, whose pixels share common multi-spectral properties. This procedure is extensively described below, while an outline of the proposed image segmentation approach is presented in Fig. 3.23.

In our approach, we employ the Expectation-Maximisation (EM) algorithm [192], since

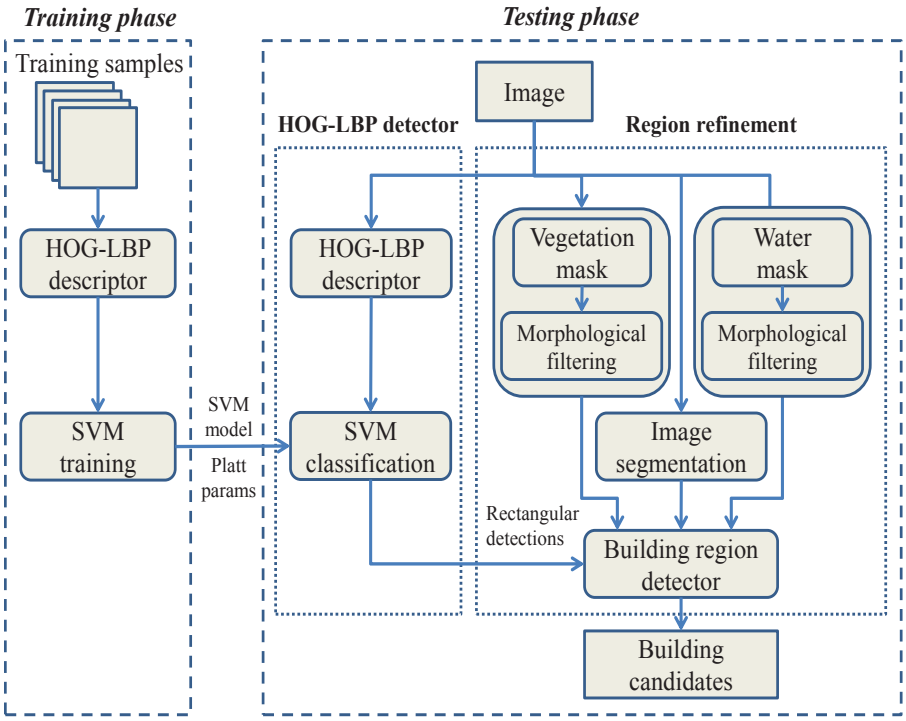



Figure 3.22: Block diagram of the proposed pixel-based building detector, along with the defined modules and their interactions.

it produces more homogeneous regions than the k-means algorithm. We considered other image segmentation approaches as well, such as mean-shift and super-pixels, but due to their dependence on parameters different from the number of classes, a parameter that we can define both in an optimal and automatic fashion, we did not apply them. Since a fixed number of clusters cannot describe adequately the spectral variations in all multi-spectral images of a dataset, we propose the use of gap statistic [193] to define in an unsupervised manner the optimal number of clusters for each tested image separately. The gap statistic can be considered as the statistical formalisation of the "elbow" heuristic technique, which considers as optimal the number of clusters that leads to an abrupt fall to the sum of point-to-cluster distances. Given that the image pixels $x_i$, $i = 1...n$ are split into $k$ clusters $C_r$, $r = 1...k$ and each cluster contains $n_r$ pixels, the within-dispersion measure (i.e. sum of point-to-point distances in a cluster over all clusters) is

$$W_k = \sum_{r=1}^{k} \frac{1}{2n_r} \sum_{i,i' \in C_r} (x_i - x_{i'})^2 \tag{3.32}$$

The gap statistic is then computed using the within-dispersion measure of the clustered image pixels, evaluated by Eq. (3.32), and the observed within-dispersion measures $W_{kb}$, $b = 1...B$ computed as in Eq. (3.32) from $B$ generated subsets of the image pixels, drawn uniformly from the initial set of image pixels.

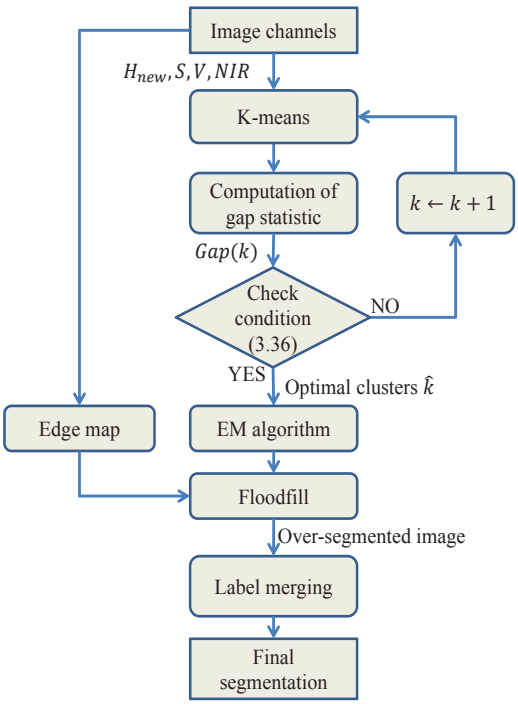



Figure 3.23: Block diagram of the proposed unsupervised image segmentation procedure.

$$Gap(k) = \frac{1}{B}\sum_{b=1}^{B} log(W_{kb}) - log(W_k) \tag{3.33}$$

The use of the logarithm in the within-dispersion measures gives an interpretation as log-likelihood. The generation of the $B$ subsets of pixels leads to the formation of an expected reference distribution, to which the initial set of pixels is compared. The larger the number of subsets $B$ is, the more accurate the reference distribution will be. Given the mean and the standard deviation of the observed within-dispersion measures

$$m_k = \frac{1}{B}\sum_{b=1}^{B} log(W_{kb}) \tag{3.34}$$

$$sd_k = \sqrt{\frac{1}{B}\sum_{b=1}^{B}(log(W_{kb}) - m_k)^2} \tag{3.35}$$

the optimal number of clusters $\hat{k}$ corresponds to the smallest value of $k$, for which it holds

$$Gap(k) \geq Gap(k+1) - sd_k\sqrt{(1+1/B)} \tag{3.36}$$

As a result, the optimal number of clusters is equal to the smallest value of $k$, for which

$log(W_k)$ drops the farthest below the expected reference distribution [193]. The optimal number of clusters $\hat{k}$ is determined by employing the gap statistic on the output of the k-means algorithm for increasing values of $k$. The k-means algorithm is computationally faster than the EM algorithm especially for large values of $k$, thus it is our selected clustering approach at this stage. Afterwards, we use the optimal number of clusters $\hat{k}$ to train $\hat{k}$ Gaussian models for the EM algorithm, which produces the final image segmentation. Each pixel is labelled based on their colour values, where instead of the Red-Green-Blue (RGB) colour space, the Hue-Saturation-Value (HSV) colour space is employed for two main reasons; unlike the RGB colour space, the components of the HSV colour space are independent of each other and the Hue ($H$) and Saturation ($S$) channels are intensity and shadow invariant, thus the segmentation procedure based on the HSV colour space produces more homogeneous regions within the underlying object boundaries. The Near-infrared (NIR) is also employed, along with the colour channels of the HSV colour space and participates in the image segmentation procedure. Due to the fact that the $H$ channel is expressed in degrees and has values in the range $[0, 360]$, the transformation of Eq. (3.37) is proposed so that the new values are in the range $[0, 1]$, as is the case for the other three spectral channels ($S$, $V$, $NIR$).

$$H_{new} = 0.5 - 0.5cos(\frac{\pi H}{180}) \tag{3.37}$$

The set of regions that the EM algorithm outputs are further processed by applying a flood-fill operation that connects neighbouring pixels with the same label and stops at edge pixels detected by the Canny algorithm. Such a procedure helps towards the accurate and robust identification of object boundaries, since objects are usually distinguished by their edges. Unfortunately, this procedure results in the over-segmentation of some image regions. To counter this, a region merging procedure is afterwards applied, where each region with size smaller than a threshold $T_{small}$ is merged with one of its neighbouring regions, with which it has the best colour similarity. The threshold $T_{small}$, which represents the area of the smallest region we wish to retain, should be set to a value quite smaller than the smallest building that should be detected in order to retrieve even non-homogeneous in colour buildings that were split in more than one regions. The purpose of $T_{small}$ is to get rid of the really small regions that increase the complexity of the problem, while being insignificant for the building detection task. The remaining image regions make up the pool of the candidate building regions from which our proposed novel building region detector selects.

### 3.5.2 Vegetation mask

One of the most accurate and well-known methods to detect vegetation in multi-spectral images is the NDVI index [194]. The NDVI index has been successfully applied in nu-

merous remote sensing applications as an indicator of vegetated areas. This index takes advantage of the NIR band that most satellites provide to identify where vegetation exists. The rationale behind the use of NDVI is based on the fact that light is better absorbed from vegetation than man-made structures. Consequently, this index produces high positive values for vegetated regions and low positive or negative values for non-vegetated areas. NDVI is computed using the near-infrared $\rho_{NIR}$ and red $\rho_R$ channels as shown below

$$NDVI = \frac{\rho_{NIR} - \rho_R}{\rho_{NIR} + \rho_R} \tag{3.38}$$

NDVI is computed for each image pixel and then an optimal threshold is automatically determined using Otsu's method [195] in order to identify the pixels belonging to vegetation. The threshold is computed based on the minimisation of the intra-class variance of the NDVI values between the pixels belonging to the "vegetation" and "non-vegetation" classes. Pixels with values higher than the value of threshold are labelled as vegetation. As a result, a binary mask is formed, where vegetation pixels are highlighted. A morphological opening, followed by a morphological closing operation, is finally applied to remove small "holes" or "islands" produced in the binary vegetation mask.

### 3.5.3 Water mask

Water extraction can be proved really useful for the task of building detection. An urban area may not depict a significant number of water bodies, nevertheless the water in urban environments is usually concentrated in swimming pools. Since swimming pools usually have rectangular shapes, a building detection method that relies on shape features can mistakenly consider swimming pools as candidate buildings. As a result, the detection of water bodies and the removal of swimming pools can lead to false alarm reduction of a building detection methodology. A commonly applied index that can differentiate the water class from other classes is the Normalised Difference Water Index (NDWI) [196]. The NDWI index is computed using the green $\rho_G$ and near-infrared $\rho_{NIR}$ channels as shown below

$$NDWI = \frac{\rho_G - \rho_{NIR}}{\rho_G + \rho_{NIR}} \tag{3.39}$$

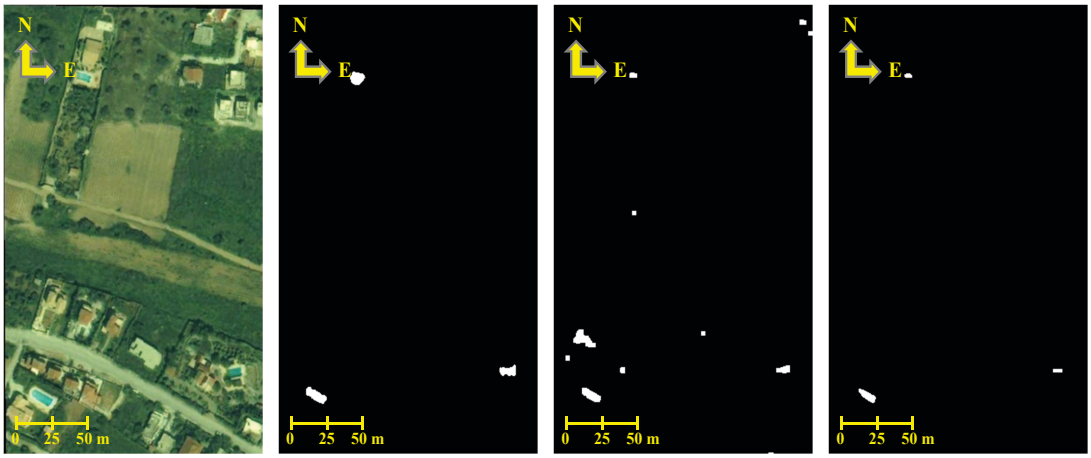
A low positive threshold, such as 0.3, can be set and pixels with NDWI values over this threshold can be labelled as water [196]. However, we notice that a single threshold cannot work in certain images of the Athens dataset as there are pixels that do not belong to water pools, but are mistakenly recognised as such. Furthermore, we test Otsu thresholding of the NDWI mask, but the results reveal that it cannot sufficiently differentiate the "water" and "non-water" classes, as many "non-water" pixels are identified to belong to the "water"

class. We believe that this happens because water pixels are usually too few in number with respect to the other class and this causes bias to the separation of the two classes using the Otsu's method.

To properly identify water in multi-spectral images, we propose a novel method that is based on the observation that water bodies usually show much higher values for the green and blue channels, rather than the red and near-infrared channels. Given the red $\rho_R$, green $\rho_G$, blue $\rho_B$ and near-infrared $\rho_{NIR}$ channels, the proposed water mask is defined using the following set of binary operations

$$\begin{aligned} WaterMask = &(\rho_G > 2\rho_R) \text{ AND } (\rho_G > 2\rho_{NIR}) \text{ AND} \\ &(\rho_B > 2\rho_R) \text{ AND } (\rho_B > 2\rho_{NIR}) \end{aligned} \tag{3.40}$$

Although the proposed method of water extraction seems to miss a few pixels belonging to the "water" class, it rejects most of the false positives that the NDWI index produces. Since our proposed methodology discards the regions that belong to the "water" class, a water detection method that presents a lower recall but a higher precision is preferable. The result of applying Eq. (3.40) to a multi-spectral image is a binary mask, where water pixels are highlighted. A morphological opening, followed by a morphological closing operation, is finally applied to remove small "holes" or "islands" in the above mentioned binary water mask, in the same fashion as in the vegetation extraction procedure. A problematic case of an image of the Athens dataset, where the NDWI index produces many false alarms that our proposed water index avoids, is presented in Fig. 3.24.




(a) (b) (c) (d)

Figure 3.24: Comparison of water detection methods in a problematic image of the Athens dataset. (a) Original image. (b) Ground truth of water locations. (c) Water mask produced by the NDWI index and threshold 0.3. (d) Water mask produced by our proposed water index using Eq. (3.40).

### 3.5.4 Building region detector

The purpose of the proposed novel building region detector is the selection of the image regions that have the highest likelihood to correspond to buildings based on the output of the enhanced HOG-LBP building detector, their colour difference to their neighbouring regions and their measure of rectangularity. The proposed pixel-based building region detector aims at achieving a transition from the detections obtained using the HOG-LBP descriptors to the accurate pixel-based delineation of the building boundaries in a multi-spectral image. A block diagram of the building region detector along with an example illustrating the employment of the proposed method on a HOG-LBP detection is presented in Fig. 3.25. What follows is a thorough explanation of the steps that the proposed building region detector consists of.

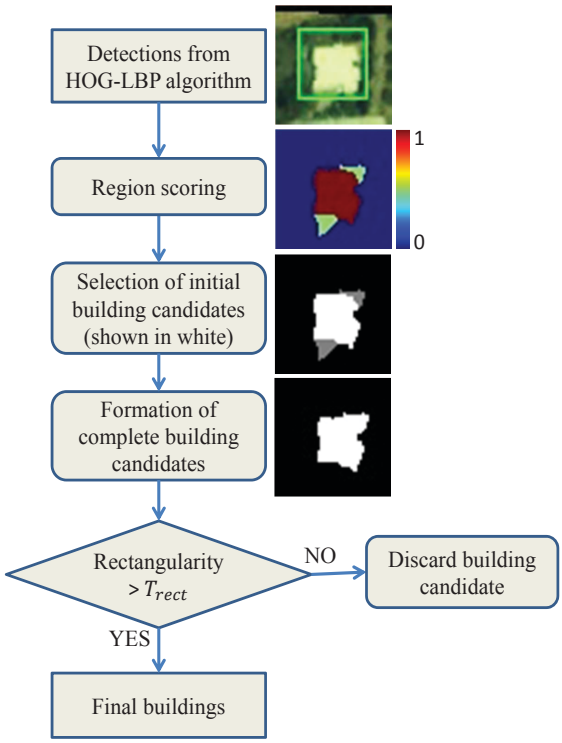



Figure 3.25: Block diagram of the building region detector, along with corresponding results of running the algorithm on a HOG-LBP detection.

The first step of the proposed methodology investigates the scoring of the regions that are formed by the image segmentation procedure. Let's assume that the proposed enhanced HOG-LBP building detector is an ideal object detector, which possesses the following properties:

1. Each detection corresponds to one and only object.

2. The detected area is of rectangular shape and its centroid is approximately the same as the centroid of the object (i.e. building).

3. The size of the object is associated to the scale in which the detection is found. This means that larger objects are detected in larger scales of the searching window.

Taking these properties into consideration, we develop a novel approach for the scoring of image regions and the selection of the best image regions as building candidates. From each HOG-LBP detection, represented as a rectangular area $A_D$ with width and height $l$ (width equals height in our case) and centroid $O_D$, a single image region $A_R$, being the result of the image segmentation procedure described in Section 3.5.1, with mean multi-spectral colour vector $C_R$, centroid $O_R$ and bounded by a rotated rectangle $A_B$ is selected as initial building candidate region based on the addition of four terms. These terms are the amount of overlap $M_{over}$ between a region and the detection, the region's rectangularity $M_{rect}$, the region's colour difference $M_{cdiff}$ to its neighbours and the distance $M_{dist}$ of the region's centroid to the centroid of the HOG-LBP detection. The purpose of employing and summing all these terms is to alleviate problems, where buildings may not have exact rectangular shapes or their contrast with the background is weak. Next, we present how the above mentioned terms are computed and why they are selected to form a scoring metric for the selection of an image region as building candidate.

- *Overlap with HOG-LBP detection $M_{over}$:* The overlap of an image region with a HOG-LBP detection is equal to the area intersection between the image region and the detection divided by the union of the areas of the image region and the detection. Thus, the overlap term $M_{over}$ is equal to $A_R \cap A_D / A_R \cup A_D$. The use of this term is based on the notion that an ideal building detector should produce a detection large enough to cover the area of a true building and small enough to include minimum background. As a result, ground areas or roads, whose sizes can be large with respect to the sizes of the corresponding HOG-LBP detections, are devalued.

- *Rectangularity $M_{rect}$:* The term rectangularity is defined as the ratio of the area of an image region divided by the area of the smallest rotated rectangle that bounds the image region. Consequently, the rectangularity term $M_{rect}$ can be computed by the ratio $|A_R|/|A_B|$. High values of rectangularity mean that the shape of an image region resembles a rectangle. Since the rectangle is the most common shape of buildings, this measure is an important term to differentiate between true buildings and false alarms.

- *Colour difference $M_{cdiff}$:* The colour difference refers to the square root of the Euclidean distance between the mean colour of an image region $C_R$ and the average weighted by size colour $C_N$ of its neighbouring image regions, which is defined by the region adjacency graph computed during the image segmentation procedure. Hence, the colour difference term $M_{cdiff}$ is defined as $\|C_R - C_N\|_{l_2}/N$, where $N$ is the dimensionality of the colour vectors $C_R$ and $C_N$. The largest the colour difference is, the more likely it is that an image region distinguishes from its surroundings and it may represent a true building.

- *Distance from detection's centroid $M_{dist}$:* This term is based on a property of an ideal object detector stating that the detections should be centred on the objects. As a result, the distance term $M_{dist}$ is equal to $e^{-2\|O_R - O_D\|_{l_2}/l}$. This means that the distance between the centroid of an image region that ideally represents a true building and the centroid of a detection should be as small as possible.

The image regions, whose centroids are located inside each rectangular HOG-LBP detection are considered. Afterwards, scores are computed for these regions based on the weighted sum of the terms defined above, using Eq. (3.41). The image region with the highest score is selected as initial building candidate. The weight vector $\mathbf{w} = [w_1, w_2, w_3, w_4]$ can be optimised by employing a grid search and evaluating the pixel-based building detection results on a validation set so that the contribution of the most reliable to describe buildings terms is boosted.

$$score = \mathbf{w} \begin{bmatrix} M_{over} \\ M_{rect} \\ M_{cdiff} \\ M_{dist} \end{bmatrix} = w_1 \frac{A_R \cap A_D}{A_R \cup A_D} + w_2 \frac{|A_R|}{|A_B|} + w_3 \frac{\|C_R - C_N\|_{l_2}}{N} + w_4 e^{-\frac{2\|O_R - O_D\|_{l_2}}{l}} \tag{3.41}$$

The next step of the proposed methodology concerns the processing of the initial building candidates to form complete building candidates. This processing step is essential so that non-homogeneous rooftops that have been split during segmentation in more than one homogeneous regions, are merged in a single building candidate. The construction of complete building candidates is based on a novel iterative procedure that is applied to each initial building candidate separately. However, before the iterative procedure initiates, each initial building candidate is checked in order to find whether a significant percentage of its pixels (i.e. more than a threshold $T_{over}$) belongs to either vegetation or water by employing the previously computed vegetation and water masks. In such case, the initial building candidate is discarded as false alarm without further processing. Consequently, larger values of the threshold $T_{over}$ decrease the tolerance of the algorithm to accept as building candidates regions with significant amount of pixels classified as vegetation or water. Otherwise, the iterative procedure is executed during which, the regions neighbouring the building candidate that do not belong to either vegetation or water are searched and the one with the closest colour similarity with the building candidate is selected. If the merging of the two regions forms a region that achieves a larger measure of rectangularity than the initial region, then the region merging is performed and the merged region becomes the new initial building candidate for the next iteration of the algorithm. Otherwise, the algorithm terminates and the region $A_R$ that has been formed so far is checked under two conditions, shown in Eq. (3.42), before it is accepted as a

complete building candidate.

$$|A_R| \geq T_{small} scale_0^2 \tag{3.42a}$$

$$|A_R| \leq \frac{|A_w|}{F_{dev}} scale_0^2 \tag{3.42b}$$

The use of these conditions is based on the fact that the HOG-LBP detector does not only provide information about the location of a building, but also about the size of the building. This information is stored at the scales of the detections and is employed in order to discard candidate buildings that are less likely to correspond to true buildings. Given the scale $scale_0$ of the largest HOG-LBP detection that encloses region $A_R$ (i.e. the centroid of the region lies inside the HOG-LBP detection) and the initial searching window $A_w$, the first condition of Eq. (3.42) states that the area of the region $A_R$ should be larger or equal to the size of the smallest region that is retained from the image segmentation procedure, which is equal to $T_{small}$, multiplied by $scale_0^2$. The second condition of Eq. (3.42) states that the size of the region $A_R$ should be smaller or equal to the size of the largest rectangular HOG-LBP detection that encloses the region, divided by a factor of deviation $F_{dev}$. The factor of deviation has values in the range $[0, 1]$ and controls the size of the accepted building candidates with respect to the size of a HOG-LBP detection, hence a large factor $F_{dev}$ limits the range of building sizes the algorithm accepts. The final output of the building candidate processing is a set of complete building candidates and their corresponding measures of rectangularity that are employed in the final part of the methodology. An outline of the algorithm that forms the complete building candidates is presented in Fig. 3.26.

Finally, the complete building candidates are tested whether their rectangularity is over a user-defined threshold $T_{rect}$. Ideally, we would like to retain building candidates with clear rectangular shapes (i.e. high values of rectangularity) and hopefully, most falsely detected building candidates will be discarded due to their low measures of rectangularity. The building candidates with rectangularity higher than the threshold $T_{rect}$ form the final output of our proposed pixel-based building detector.

### 3.5.5 Sensitivity to parameters

Experiments are conducted in order to determine how the parameters of the pixel-based building detector affect its performance on the multi-spectral images of the Athens dataset. The number of generated data subsets $B$ that affects the accuracy of the computed gap statistic is set equal to 20. Experiments reveal that the proposed methodology is robust to changes to the number of generated data subsets $B$ since higher values affect slightly the accuracy of the gap statistic, while they increase significantly the computational time. Furthermore, the morphological operations applied during the formation of the final veg-

**Algorithm: Computation of complete building candidates**

Input: Image regions $r_i$ with positive score, scale $scale_i$ and rectangularity $rect_i$ , Minimum size of building $T_{small}$ , Factor of deviation $F_{dev}$ , Size of the initial searching window of HOG-LBP detector $|A_w|$.

Output: Complete building regions $b_j$ , $j = 1..B$.

For each region $r_i$ do:
  $scale_0 = 0$
  While do
    - get neighbours of region $r_i$
    - select neighbour $r_s$ with largest colour similarity to $r_i$
    - merge regions $r_i$ and $r_s$ and compute rectangularity $rect_{is}$
    - If $rect_{is} < rect_i$
        return $r_i$
    - Else
        $r_i \leftarrow$ merged($r_i$ , $r_s$)
        $scale_0 = \max(scale_0 , scale_i)$
      End
  End
  If $Area(r_i) \geq T_{small} scale_0{}^2$ AND $Area(r_i) \leq \frac{|A_w|}{F_{dev}} * scale_0{}^2$
    - $b_j \leftarrow r_j$
End

Figure 3.26: Outline of the algorithm that forms the complete building candidates written in pseudo-code.

etation and water masks are performed using a square structuring element of size $5 \times 5$ in pixels. This size is large enough to suppress small details and "holes" in the masks and small enough to avoid significant misclassification errors. Moreover, the frequency of the cosine-base distance function $\alpha$ that affect the performance of the enhanced HOG-LBP building detector with Platt scaling was examined in Section 3.4.2 and it is set equal to the optimal value of 0.09. The parameters that affect the performance of the proposed pixel-based building detector and should be properly defined are the factor of deviation $F_{dev}$, the minimum area of accepted image regions in the image segmentation procedure $T_{small}$, the percentage of building candidate's pixels that should belong to either vegetation or water so that the building candidate is discarded $T_{over}$, the weight vector $\mathbf{w}$ and the rectangularity threshold $T_{rect}$. These parameters are defined by experimentation on the 5 multi-spectral images of the Athens "image" validation set and they are chosen based on the optimisation of the pixel-based performance of our proposed building detector.

The minimum area of an image region $T_{small}$ should be set quite smaller than the size of the smallest building that the algorithm should detect in an image, in order to take into account non-homogeneous rooftops. Since we employ a HOG-LBP building detector with minimum and maximum searching windows, there is an indication of what sizes of buildings are searched in an image. As a result, $T_{small}$ can be set to a value proportional to the size of the initial searching window of the HOG-LBP detector. The weight vector $\mathbf{w}$ is determined by a grid search for component values in the range $[0, 1]$ with a step of 0.05. This is possible because we are only interested in the relative score differences between the compared regions inside a rectangular HOG-LBP detection and not the absolute score values. Experiments show that the most significant scoring term is the colour difference

Table 3.12: Parameter settings for the proposed approach.

| Parameter | Value |
|---|---|
| Frequency of cosine-based measure ($\alpha$) | 0.09 |
| Data subsets for gap statistic ($B$) | 20 |
| Minimum image region ($T_{small}$) | 30 pixels |
| Factor of deviation ($F_{dev}$) | 0.45 |
| Overlap threshold for vegetation or water ($T_{over}$) | 30% |
| Weight vector $\mathbf{w}$ | [0.4,0.1,0.8,0.1] |
| Rectangularity threshold ($T_{rect}$) | 0.4 |

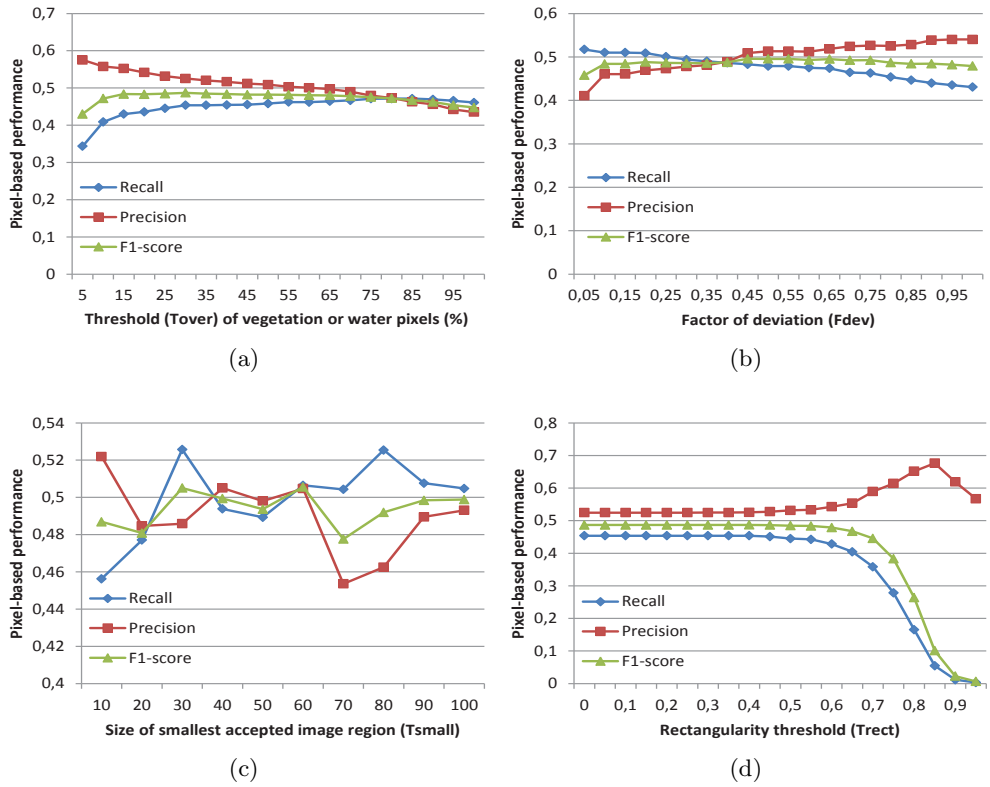



Figure 3.27: Performance of our proposed methodology on the Athens "image" validation set for different parameter settings. (a) Effect of overlap threshold with vegetation and water $T_{over}$, (b) effect of factor of deviation $F_{dev}$, (c) effect of size of smallest accepted region $T_{small}$ and (d) effect of rectangularity threshold $T_{rect}$ on pixel-based performance.

of an image region to its neighbours, followed by the overlap of an image region and the corresponding detection of the HOG-LBP detector. Table 3.12 presents the parameter values that achieve the optimal results on the Athens dataset. Fig. 3.27 illustrates the effects of varying the parameter values on the performance of the proposed pixel-based building detector on the Athens "image" validation set.

From Fig. 3.27, a few conclusions can be drawn. The proposed pixel-based building detector is quite robust in changes of the values of the parameters $T_{over}$, $F_{dev}$ and $T_{rect}$. More specifically, most of the vegetation and water areas have already been discarded during the object-based classification. The remaining areas are devalued by the region scoring

procedure due to their low contrast with the background and their low rectangularity and they are not selected as building candidates. Furthermore, too large regions, that would have been discarded by the $F_{dev}$ threshold, are already devalued by the region scoring procedure. Moreover, most of the regions with low rectangularity scores have already been rejected during the previous steps of the proposed pixel-based building detector and increasing the rectangularity threshold does not lead to a significant increase in the accuracy of the algorithm. In cases where the rectangularity threshold is set to too high values, a drop in the pixel-based performance of the proposed pixel-based building detector is observed. Finally, the size of the smallest accepted image region seems to affect significantly the performance of the pixel-based building detector, as shown in Fig. 3.27 (c). This can be attributed to the fact that the proposed pixel-based building detector is quite sensitive to the quality of the image segmentation, requiring an as accurate as possible image segmentation procedure.

### 3.5.6 Evaluation

The proposed pixel-based building detector (M9) is tested on the multi-spectral images of the Athens dataset and its performance is compared to another pixel-based building detector, proposed by Chaudhuri et al. in [58]. The building detector in [58] is based on Internal Gray Variance (IGV) and morphology operations and it is referred from now on as IGV building detector (M10). According to the methodology of the IGV building detector, a tested image is initially enhanced by a combination of morphological opening and closing operations with a structuring element or template that is extracted from a representative building in the image. The purpose of the image enhancement is to improve the contrast of the target objects (i.e. buildings) from the background, while smoothing irrelevant details. Afterwards, IGV features that evaluate the intensity variance of pixels in their neighbourhoods are extracted and used in order to define seed points. From the seed points, clusters of edge pixels are formed that are candidates for representing borders of buildings. These borders are then thinned using filtering operations. The authors noticed that roads are also detected as man-made objects and thus a shadow detection technique is employed to reject road segments and other falsely detected regions. Finally, an automatic thresholding technique based on the Otsu's method is employed to compute the final building regions.

We implement the IGV building detector with the exception that their proposed shadow detection technique is replaced by the algorithm of Liu et al. [197] for panchromatic images. This substitution is based on the fact that the description of their shadow detection technique was not clear enough to be implemented and their experiments with other shadow detection techniques revealed that the algorithm of Liu et al. is a worthy substitute of their own shadow detection technique by performing slightly better than their method in some scenarios and slightly worse in others. Furthermore, we experiment with

various parameter values in order to find out that the parameter values they propose, yield the best results in the Athens dataset. Finally, experiments are conducted on the selection of optimal building candidates as structuring elements for the image enhancement step of the IGV building detector.

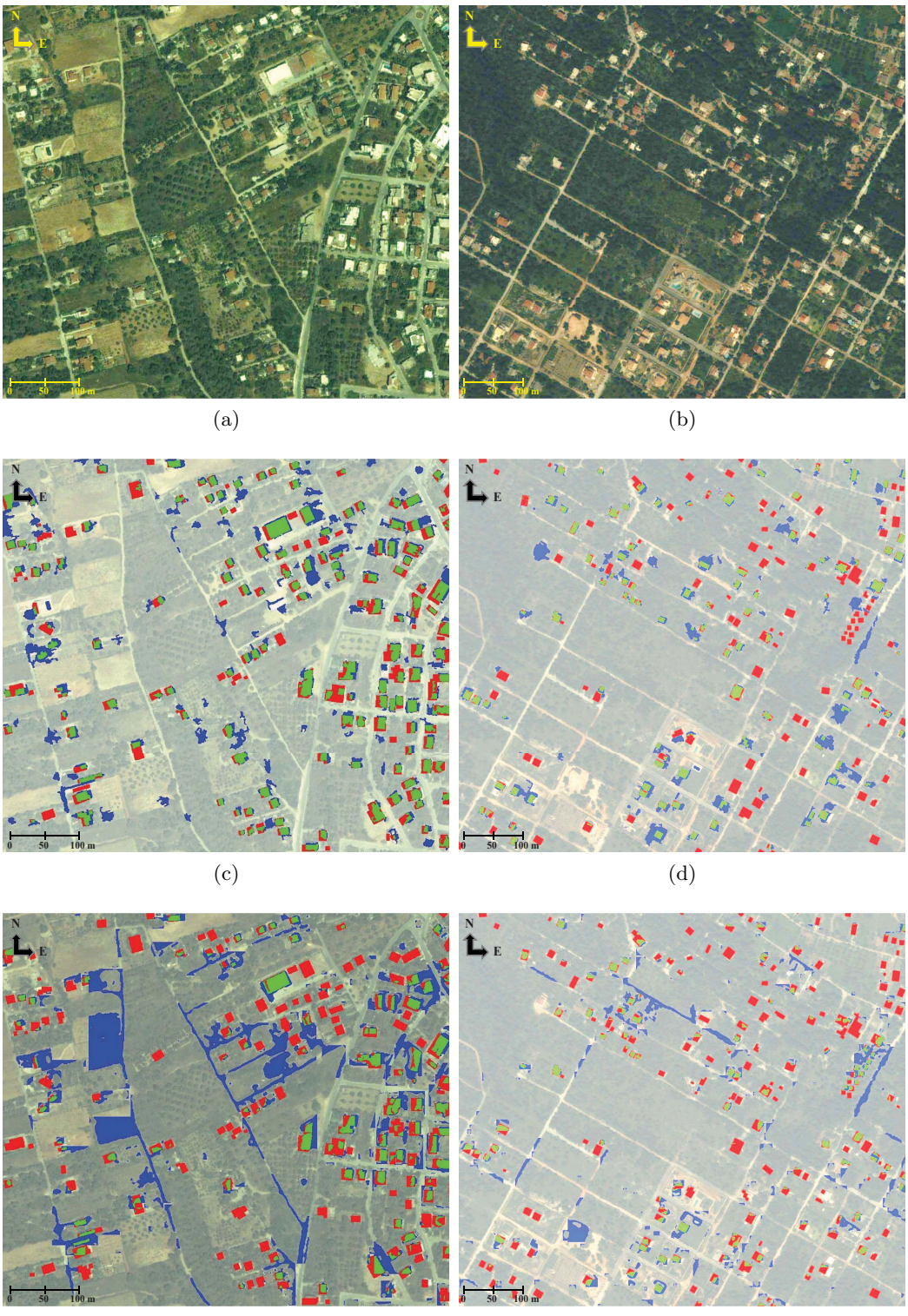
The performances of methods M9 and M10 at the object and pixel level on the Athens dataset are presented in Table 3.13, where the best values are shown in bold. The results demonstrate the superiority of our method M9 over the method M10 both at object and pixel level. More specifically, the method M9 improves the object-based and pixel-based $F_1$-score by almost 12.8% and 69% respectively, in comparison with the performance of method M10 on the Athens dataset. The method M9 is inferior only in the object-based precision, where the method M10 achieves the highest object-based precision from all tested object and pixel-based methodologies in the Athens dataset (M1-M9). The interested reader is urged to compare the results of Table 3.13 with those in Table 3.11.

Table 3.13: Results, presented in the format: mean ± std, of pixel-based building detectors on the Athens dataset.

| Method | Object-based | | | Pixel-based | | |
|---|---|---|---|---|---|---|
| | Recall | Precision | $F_1$-score | Recall | Precision | $F_1$-score |
| M9 | **0.717±0.069** | 0.73±0.1 | **0.717±0.05** | **0.556±0.06** | **0.494±0.082** | **0.516±0.043** |
| M10 | 0.476±0.233 | **0.859±0.09** | 0.588±0.2 | 0.334±0.17 | 0.314±0.122 | 0.28±0.06 |

Our proposed method M9 achieves a standard deviation with respect to the metric of $F_1$-score equal to 0.05, which is much better than the reported standard deviation of method M10 (0.2). This means that the method M9 is more robust to illumination variations and spectral and spatial resolution differences, present in the Athens dataset, than the method M10. As a matter of fact, the method M9 can be considered more robust than any object-based building detector that was presented in the previous sections (M1-M8) both at the object and pixel level. An illustration of the building detection results of methods M9 and M10 in a Quickbird and a WorldView-2 satellite image of the Athens dataset is presented in Fig. 3.28.

The inferior performance of method M10 on the Athens dataset can be attributed to three main reasons. Firstly, the method M10 is strongly affected by the selection of a suitable building template or structuring element that enhances the contrast between the buildings and other objects in a satellite image. We experiment with various templates to get optimal results, but a selection of a non-representative template can significantly affect the results. Secondly, noise and low contrast present in the Athens dataset can severely affect the extraction of building edges. Inaccurate or absent building edges lead to false building hypotheses that deteriorate the performance of method M10. Finally, in high-resolution satellite images, shadow is present not only close to buildings but also close to other objects, such as tall trees and fences. Furthermore, the position of the sun the time the image is captured can significantly affect the quality of the shadow information, making it even irretrievable. As a result, non-building edges can be wrongly assumed to




(a) (b)

(c) (d)

(e) (f)

Figure 3.28: Comparison of methods M9 and M10 in (a) a QuickBird and (b) a WorldView-2 image of the Athens dataset. (c), (d) Detected buildings from our method M9 for images (a), (b) respectively. (e), (f) Detected buildings from the method M10 for images (a), (b) respectively. True positives are shown in green, false positives are shown in blue and missed pixels are presented in red.

belong to buildings, leading to a severe drop in the performance of method M10. This phenomenon can be observed in Fig. 3.28, where roads are falsely detected and retrieved as candidate buildings due to the presence of "shadowy" edges that could not be discarded as "non-building" edges.

Next, our proposed pixel-based building detector (M9) is compared with our proposed object-based Enhanced HOG-LBP building detector with Platt scaling (M5), defined in Section 3.4. The results show that the pixel-based performance of method M9 is significantly better than the performance of method M5, as several false detections are removed and building boundaries are correctly identified. This means that the purpose of the proposed method M9, which is to make the transition from the detection of buildings, bounded in rectangular areas, to accurate and robust building boundary delineation is achieved. Unfortunately, some buildings are lost in the process, leading to a drop in the pixel-based recall and the object-based performance of method M9 on the Athens dataset. This loss of buildings can be attributed to the fact that the quality of image segmentation can be severely degraded in cases of noise in the satellite image or buildings with similar colour information to the background. Moreover, there are some buildings with irregular shapes that do not present high rectangularity values and therefore, they are discarded either during the formation of the initial building candidates or by the employment of the rectangularity threshold. Finally, the corresponding standard deviations of the employed object-based and pixel-based metrics show that the method M9 manages to improve the robustness of method M5 in the illumination variations and noise present in the Athens dataset.

Although our proposed method M9 shows superior pixel-based performance over other object-based and pixel-based building detection methods, there are still some significant limitations. Firstly, the results are strongly affected by the output of method M5, meaning that buildings lost by method M5 cannot be recovered in the next steps of the proposed method M9. Secondly, the quality of image segmentation plays a crucial role to the building detection task. Buildings that are not accurately segmented by our proposed image segmentation procedure may not be selected by the region refinement procedure, thus being discarded by the method M9. Although more than one scoring terms are aggregated, as shown in Eq. (3.41), there can still be non-building regions that achieve higher scores than the building regions. Finally, our method M9 may fail to reject false positives that describe small road segments or land patches (see blue areas in roads and land in Fig. 3.28 (d)). In such occasions, the extracted image regions can present high rectangularity and strong contrast with the background, thus mistakenly recognised as buildings. Such problems can however be alleviated if steps are taken in the direction of segmenting a satellite image accurately, since then roads and land fields will extend out of the HOG-LBP detections and they will be devalued by the scoring metric of Eq. (3.41).

## 3.6 Comparison of HOG-based building detectors

Table 3.14 summarises the object-based and pixel-based performances of the proposed HOG-based building detectors. From the object-based results, it can be concluded that the optimal building detector is the Enhanced HOG-LBP detector with Platt scaling (M5). The results reveal that this method achieves an increase in the object-based metric of $F_1$-score by 10.6% with respect to the second best building detector, which is the HOG-LBP building detector. The results show that the analysis and incorporation of more features as the chapter progresses leads to the creation of more accurate and robust object-based building detectors that are able to compete with and surpass state-of-the-art building detection methodologies, as those presented in Table 2.2.

Table 3.14: Performance of the proposed HOG-based building detectors on the Athens dataset. The results are presented in the format: mean ± std. BD: building detector

| Method | Object-based | | | Pixel-based | | |
|---|---|---|---|---|---|---|
| | Recall | Precision | $F_1$-score | Recall | Precision | $F_1$-score |
| HOG BD | 0.858±0.085 | 0.577±0.131 | 0.675±0.069 | 0.881±0.088 | 0.093±0.109 | 0.168±0.103 |
| Improved HOG BD | 0.853±0.058 | 0.606±0.12 | 0.709±0.067 | 0.875±0.083 | 0.101±0.098 | 0.181±0.091 |
| HOG-LBP BD (M1) | **0.963±0.073** | 0.676±0.155 | 0.782±0.108 | **0.966±0.023** | 0.118±0.044 | 0.207±0.069 |
| Enhanced HOG-LBP BD (M5) | 0.934±0.114 | **0.823±0.102** | **0.865±0.076** | 0.945±0.066 | 0.135±0.055 | 0.23±0.079 |
| Pixel-based BD (M9) | 0.717±0.069 | 0.73±0.1 | 0.717±0.05 | 0.556±0.06 | **0.494±0.082** | **0.516±0.043** |

As far as pixel-based performance is concerned, the proposed pixel-based building detector achieves the optimal performance, improving the pixel-based metric of $F_1$-score by 124.3% with respect to the object-based enhanced HOG-LBP building detector that it is based on. This means the use of the region refinement procedures has a beneficial effect on the pixel-based performance of the proposed building detector, although a drop in the object-based performance is noticed.

## 3.7 Summary

This chapter proposes a set of methodologies to solve the problem of building detection in high-resolution multi-spectral images by employing HOG features. Starting from an optimised to the building detection task HOG detector, we study and analyse the effect of other features, when used in conjunction with the HOG features, in an attempt to create a robust and accurate building detection algorithm. The feature analysis leads to two basic propositions; the probabilistic fusion of HOG, NDVI and FAST features and the combined extraction of HOG and LBP features. Both methods improve the performance of the initial HOG building detector by reducing the number of false detections it produces.

One of the biggest contributions of this chapter is the novel use of a cosine-based distance function. The cosine-based distance function replaces the $l_2$-norm in the computation of distances between features and allows for a suppression of the effect of noise and outliers. The noise suppressing effect of the cosine-based distance function enables the SVM classifier to train a more accurate and robust model for the classification of feature vectors and therefore, improve the performance of a building detector. Furthermore, Platt scaling allows the transformation of the SVM scores to probabilities and the definition of a new threshold for the acceptance or rejection of the feature vectors, leading to further improvement in the precision of a building detector. As a result, the final proposed object-based building detector (M5) is capable of achieving an impeccable recall, by correctly identifying most of the buildings present on the images of the Athens dataset, despite shadows, occlusions and illumination variations, while also achieving a high precision by making fewer mistakes than any other object-based building detection methodology that was tested.

Finally, a pixel-based building detector (M9) is proposed to improve the pixel-based performance of the proposed object-based building detector. A set of region refinement processes is applied to the output of the optimal object-based building detector and makes the transition from the rectangular detection of building areas to the accurate building boundary delineation possible. The region refinement processes consist of an unsupervised image segmentation technique, the extraction of vegetation and water masks and the construction of building hypotheses by employing a novel region scoring procedure. The results verify the significant improvement on the pixel-based performance of a building detector, when the region refinement processes are employed.

In this chapter we address the problem of building detection by performing a manual search on descriptive features that can improve the performance of a building detector. In the next chapter, we are still concerned with the problem of building detection, but we let the convolutional neural networks to automatically generate descriptive features that can assist in the attainment of accurate building detection results.

# CHAPTER 4

## CNN-based Building Detection

In the previous chapter, a thorough investigation of the effect of the combination of various descriptive features on the performance of a building detector was performed. In this chapter, we address again the problem of building detection, but we do not manually select discriminatory features. Instead, we employ a Convolutional Neural Network (CNN) to automatically build complex features that can be used to solve the problem of building detection.

CNNs belong to a category of machine learning algorithms that aims at modelling multiple levels of representation or abstraction. Although the theory behind them was developed three decades ago, their true potential has not been shown until recently, when several researchers demonstrated their incredible ability to achieve state-of-the-art performances in various disciplines, such as computer vision and video, audio and text processing. Their impeccable performance on object detection tasks motivated us to employ CNNs for the problem of building detection from multi-spectral images.

CNNs are a special type of feed-forward artificial neural networks that are composed of convolutional filters. These filters exploit spatially local correlations in the input data and they are suitable for processing multi-dimensional arrays, such as images. The main advantage of a CNN over other classification methods is its ability to automatically discover and extract discriminatory features without the intervention of a human expert and employ them for the chosen classification task. A CNN starts from the simplest features that can be image pixels and ends up to more complex representations as we get deeper into the network. Afterwards, these features are used in the CNN classification phase. As a result, a CNN is able to both extract meaningful features from the input data and classify the data into classes using the extracted features.

However, CNNs have a few disadvantages or limitations that prohibit their use in several applications. Firstly, CNNs need a large amount of labelled data in order to extract meaningful features that can be employed for accurate classification. In the case of small

datasets, the problem has been partially solved by employing data augmentation techniques [198, 199]. However, data augmentation is not always sufficient. Secondly, the CNN training and testing consist of computationally demanding operations that could take several days to be executed in conventional computer systems. Luckily, the breakthroughs in computing enable the use of Graphical Processing Units (GPUs) for the CNN training and testing. GPUs are computationally efficient processing units with large amount of dedicated memory that can handle computationally heavy operations. Finally, there are no guidelines to assist in the development of a general CNN architecture that can be employed in any classification task. The existence of several layers that can be combined in a CNN, each one with its own hyper-parameters that need to be optimised, creates severe problems on all attempts for the development of a general CNN architecture. As a result, a developer usually relies on his instinct and prior knowledge on the problem and behaviour of CNN layers to decide the optimal CNN architecture.

Motivated from the need of a general guideline for the development of a CNN, we propose in this chapter a shallow CNN architecture that consists of few layers and can be optimised for the building classification task. There are two main reasons behind the implementation of a shallow CNN. Firstly, the images of the datasets that we experiment on are small, thus prohibiting the use of too deep neural networks and secondly, shallow CNNs tend to perform better than deeper CNNs as the accuracy can start degrading, when the depth of a network increases. This degradation of the performance of a CNN is attributed to the fact that deeper networks are not that easy to optimise as their shallower counterparts [200, 201, 202]. However, deep CNNs are necessary when dealing with more complex problems and larger training sets. Furthermore, we propose two novel layers that increase the generalisation ability and accuracy of the proposed CNN architecture. The first layer is called Normalisation Layer (NL) and it combines the power of both raw and normalised features, while the second layer is called Euler Transform Layer (ETL) and it is concerned with the Euler transformation of features to a new representation that is more robust to noise, inspired from the cosine-based distance function that was employed in the previous chapter. The optimisation simplicity of a shallow CNN allows us to analyse in depth the effect of the novel CNN layers on the discrimination ability of the proposed CNN architecture.

All CNN algorithms and the two novel layers that are introduced in this chapter are developed using the Torch software. The Torch software is a computing framework that enables the use of the scripting language Lua for the development of machine learning algorithms and the use of GPUs for the training and testing of the algorithms. We programmed the Euler transform layer and tested it using Torch in order to be certain that its back-propagation computations are correct and accurate. Finally, a NVIDIA Tesla K40 GPU is utilised in order to increase the computational speed during the training and testing of the CNNs.

The main contributions of this chapter are pointed out below:

- A proposed shallow CNN architecture that is optimised for the building classification task.
- A novel normalisation layer that improves the generalisation power of the proposed shallow CNN.
- A novel Euler transform layer that improves the performance of the proposed CNN architecture by increasing its robustness to noise and outliers.
- A new building detection technique that employs the optimal pixel-based HOG-LBP building detector presented in the previous chapter and combines it with the discrimination abilities of the proposed CNN architecture.

The structure of the chapter is as follows. Section 4.1 presents the proposed shallow CNN architecture, called *Modular*-CNN that can be employed and optimised based on its classification abilities on the building detection task. Section 4.2 presents the novel NL layer that can be added to the *Modular*-CNN to improve its generalisation ability, while Section 4.3 presents the novel ETL layer that can be added to the *Modular*-CNN and improve its robustness to the presence of noise and outliers. Section 4.4 presents a deep CNN architecture that is employed for comparison with the proposed *Modular*-CNN on the available datasets. Section 4.5 presents the experimental procedure for the optimisation of the *Modular*-CNN and the way it is combined with the pixel-based building detector of the previous chapter for improved building detection results. Finally, Section 4.6 concludes the work done in this chapter by referring to the contributions in the image classification and object detection tasks.

## 4.1 *Modular*-CNN architecture

In this section, we present our proposed shallow CNN architecture and the procedure for its optimisation for the building classification task. This CNN architecture consists of maximum two basic units or *modules* placed in a sequential and/or parallel configuration. We call this architecture *Modular*-CNN for sake of the *modules* that it consists of. A *module* is defined as a basic CNN that consists of three layers: a spatial convolutional layer, an activation function and a pooling layer placed in a sequential order as shown in Fig. 4.1.

The convolutional layer is responsible for extracting meaningful features for the classification procedure. The use of the convolutional layer is based on the observation that the spatial correlation between pixels is local and the assumption that the local statistics are similar at different locations. As a result, a filter, whose parameters are optimised during the CNN training phase, runs over each feature map and extracts new features from spatially local features. The use of a single filter that processes an entire feature map improves the computational efficiency of the convolutional layer by reducing the pa-

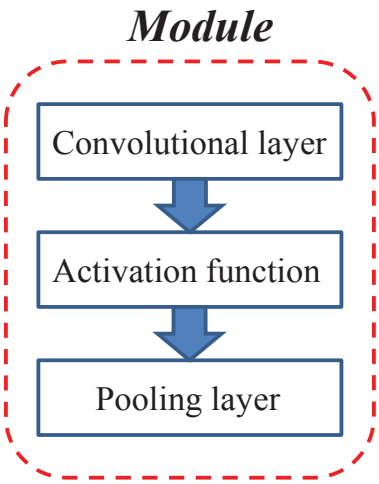



Figure 4.1: The proposed basic unit or *module* of the *Modular*-CNN architecture.

rameters that need to be optimised. The output of the convolutional layer is given by Eq. (4.1), where $h_k^{n-1}$ and $h_j^n$ are the $k$-th and $j$-th feature maps of the $(n-1)$-th and $n$-th convolutional layers of the CNN respectively and $w_{kj}^n$ is the convolutional filter or kernel of the $n$-th CNN layer that processes the $k$-th feature map in order to output the $j$-th feature map. An illustration of the computations performed in a convolutional layer of a CNN is presented in Fig. 4.2.

$$h_j^n = \sum_{k=1}^{K} h_k^{n-1} * w_{kj}^n \tag{4.1}$$

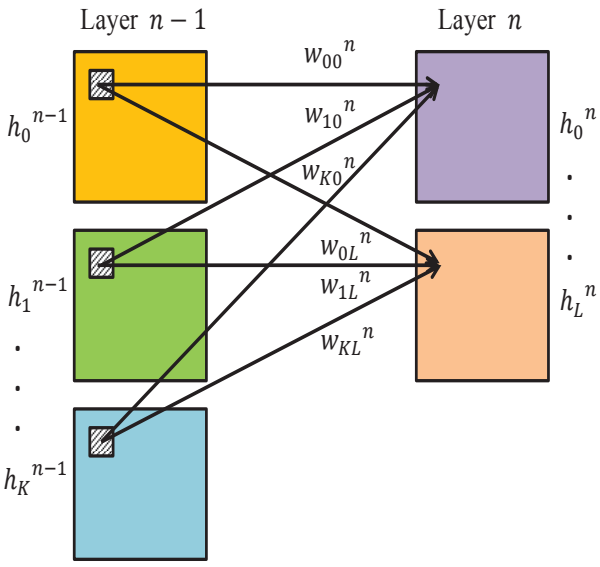
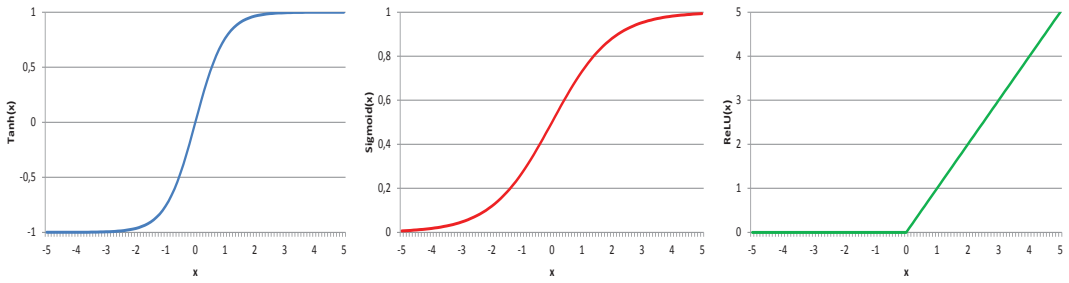



Figure 4.2: Behaviour of convolutional layer of a CNN.

The activation function is the second layer of the proposed *module* and it has as a purpose to add non-linearity in the linear operations of the convolutional layer. Common activation

functions that are employed in a CNN architecture are the hyper-tangent (tanh), sigmoid and Rectified Linear Unit (ReLU) functions. The ReLU function can be described by the operation $max(0, x)$, which means that it maps the negative input to the value of zero, while keeping the positive input unchanged. The advantages of the ReLU function over other activation functions are the faster learning, the simplification of back-propagation and the avoidance of saturation issues [11]. Plots of the hyper-tangent, sigmoid and ReLU activation functions are presented in Fig. 4.3.




(a) (b) (c)

Figure 4.3: Plots of (a) hyper-tangent (tanh), (b) sigmoid and (c) ReLU activation functions.

The final layer of the proposed *module* is the pooling layer. The pooling layer performs an operation on a neighbourhood of a feature map and returns a new feature map that has smaller spatial dimensions than the input feature map. This procedure allows the incorporation of spatial invariance in the CNN training phase and the formation of more complex patterns. Common pooling layers are the max-pooling and average-pooling layers that compute the maximum and average values respectively over the feature map neighbourhoods.

The top of the *Modular*-CNN architecture consists of two fully connected linear layers that receive as input the output of the configuration of *modules* that were previously described. The first fully connected linear layer reduces the number of features from $M$ to $N$ in order to decrease the number of parameters that needs to be optimised, while the second layer performs a linear mapping of the features without modifying the dimensionality of the feature vectors. Finally, there is a last layer that performs the classification of the feature vectors to classes. This layer can be a linear layer that performs a linear mapping of the feature vectors to classes or a Euclidean or a Cosine layer that clusters the feature vectors to classes based on the Euclidean distance or the cosine similarity respectively. The *Modular*-CNN architecture is presented in Fig. 4.4.

The proposed *Modular*-CNN architecture has several hyper-parameters that govern its classification performance and need to be carefully optimised. Except for the dimensionality of the final feature vector $N$, each layer of a *module* has its own hyper-parameters that need to be optimised. Our strategy is to optimise the *Modular*-CNN in a *module*-wise manner, meaning that *modules* are added to the CNN architecture one by one and their hyper-parameters are optimised independently of the hyper-parameters of other *modules*.

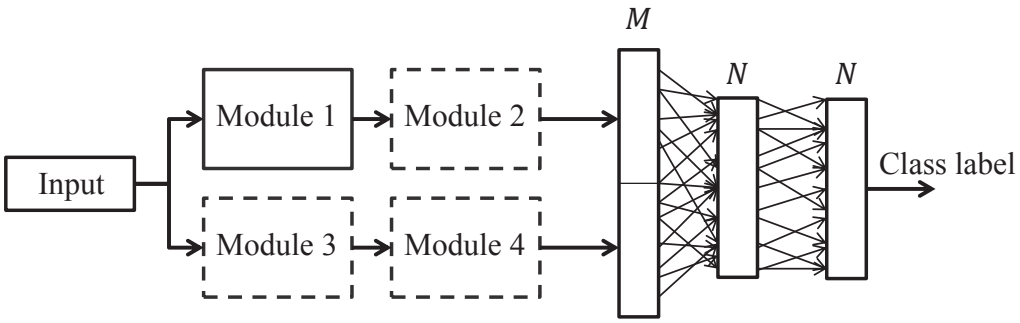



Figure 4.4: General architecture of the proposed *Modular*-CNN. Optional *modules* are presented in dashed lines

An optimised *module* is added to the CNN as long as the classification performance of the CNN increases or until the required depth or width is reached and its hyper-parameters are kept fixed, while other *modules* are added.

The training of a CNN is usually performed using stochastic gradient optimisation methods. These methods consist of rounds of forward passes, where the training samples are introduced to the CNN, labels are computed and errors with respect to the ground truth labels are evaluated and rounds of backward passes, where the computed training errors are back-propagated to the CNN layers and the weights/parameters of the CNN are adjusted.

## 4.2 Normalisation layer

Normalising the input data is a common data pre-processing method that increases the performance of a classifier, especially one that relies on stochastic gradient optimisation methods. This is true because given the random weight initialisation of a neural network, scaled features are weighted equally and allow for an improvement in the convergence speed of the network. Furthermore, too large input values can saturate some of the hidden neurons of a neural network, especially in the first training epochs. The saturation of a neuron renders the neurons of the next layers that are connected with it inactive, thus the neural network fails to learn anything. Such conditions can lead a neural network to get stuck in local optima.

Unfortunately, input normalisation is not always enough. Since the output of a CNN is a non-linear mapping of the normalised input, the effect of normalisation is in most cases lost in the feature space [203]. As a result, it is useful to normalise the computed feature vectors before they are classified to classes. On the other hand, since the raw features come from already normalised data, the neural network mapping may have led some important features to become prominent in the output and this discrimination can be lost after a new normalisation in the feature space. Based on these ideas and observations, we suggest the use of a short-cut connection before the classification of the feature vectors. The proposed

layer, called Normalisation Layer (NL), takes as input a feature vector $\mathbf{x}$ and creates a new feature vector that has twice the size of the initial vector. The first half of the new feature vector is a copy of the initial vector (i.e. $\mathbf{x}$), while the second half is a normalised by $l_2$-norm copy of the initial feature vector (i.e. $\frac{\mathbf{x}}{\|\mathbf{x}\|_{l_2}}$). A disadvantage of employing the NL layer is the fact that it outputs a new feature vector that has twice the size of the initial feature vector and consequently this leads to an increase in the computational complexity of the CNN. The NL is depicted in Fig. 4.5 and it is added at the top of a CNN architecture in order to modify the final feature vectors prior to their classification in classes.

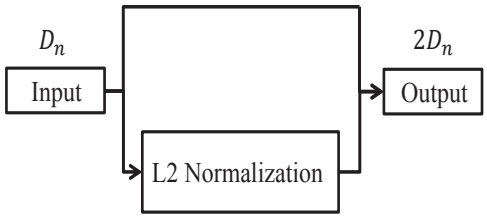



Figure 4.5: Novel NL layer. The size of the output feature vector is twice the size of the input feature vector $D_n$.

## 4.3 Euler transform layer

In Section 3.4.1, a new distance function was proposed as a replacement to the $l_2$-norm based on the observation that the $l_2$-norm is significantly affected by large values that can be attributed to noise. The cosine-based distance function is considered to be more robust to noise and outliers than the $l_2$-norm. Given two feature vectors $\mathbf{x_i}$ and $\mathbf{x_j}$ of length $L$, the cosine-based distance function was defined in Eq. (3.28) and it is also presented here for completeness

$$d(\mathbf{x_i}, \mathbf{x_j}) = \sum_{l=1}^{L} (1 - \cos(\alpha\pi(\mathbf{x_i}(l) - \mathbf{x_j}(l)))) \tag{4.2}$$

The cosine-based distance function of Eq. (4.2) is controlled by a single variable $\alpha$ that affects the response of the distance function to large differences. Small values of $\alpha$ make the cosine-based function to behave similarly to the $l_2$-norm, meaning that the distance between two feature vectors increases as their difference becomes larger. On the other hand, large values of $\alpha$ make the cosine-based distance function to suppress its response to large differences. Since large differences between feature vectors can be attributed to outliers, the cosine-based distance function attempts by regularising its control variable $\alpha$ to suppress the effect of noise and outliers.

The ability of the cosine-based distance function to suppress outliers for large values of $\alpha$ is based on the fact that it is not a monotonically increasing function as the difference

between two vectors increases, like the $l_2$-norm, but it redescends smoothly towards zero for large difference values, as shown in Section 3.4.1. This allows the cosine-based distance function to smoothly suppress large differences, which can be attributed to noise or outliers.

There are two possible ways one can take advantage of the cosine-based distance function. It can either be directly employed as an alternative to the $l_2$-norm [185, 204] or the feature vectors can be transformed to their Euler representation before they are mapped to classes. Applying the cosine-based distance function to a pair of vectors $\mathbf{x_i}$ and $\mathbf{x_j}$ of size $L$ is equivalent to transforming the feature vectors to their Euler representation $\mathbf{z_i}$ and $\mathbf{z_j}$, where $\mathbf{z_i} = \frac{1}{\sqrt{2}} e^{i\alpha\pi\mathbf{x_i}}$ and subsequently employ the $l_2$-norm function. With the use of a few trigonometric identities, it holds

$$\begin{aligned} \|\mathbf{z_i} - \mathbf{z_j}\|^2 &= \frac{1}{2} \sum_{l=1}^{L} \|(\cos(\alpha\pi x_i(l)) + i\sin(\alpha\pi x_i(l)) \\ &\quad - (\cos(\alpha\pi x_j(l)) + i\sin(\alpha\pi x_j(l))\|^2 \\ &= \sum_{l=1}^{L} (1 - cos(\alpha\pi x_i(l) - \alpha\pi x_j(l))) \end{aligned} \tag{4.3}$$

The interested reader can find the analytical proof of the equality between the Euler representation of feature vectors and the cosine-based distance function in Appendix A.

In this chapter, we choose to employ the Euler transformation method since it is easily implemented in a neural network framework. Therefore, we propose the addition of a novel layer in the *Modular*-CNN architecture. The novel layer, which is referred to as the Euler Transform Layer (ETL), is introduced before the layer that classifies feature vectors and after the NL layer that was introduced in Section 4.2. However, it is obvious that the ETL layer can be employed without the need of the NL layer and even before the NL layer. The ETL layer transforms features to their Euler representation, meaning that each feature vector $\mathbf{x}$ is now described by a cosine (i.e. $\cos(\alpha\pi\mathbf{x})$) and a sine (i.e. $\sin(\alpha\pi\mathbf{x})$) part that are concatenated together. A disadvantage of employing this layer is the fact that the Euler feature vector has twice the size of the initial feature vector and consequently the computational complexity and the parameters of the CNN are increased. The ETL layer is depicted in Fig. 4.6.

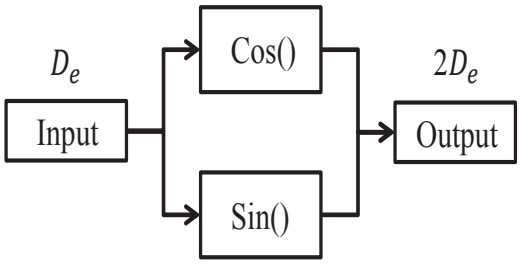



Figure 4.6: Novel ETL layer. The size of the output feature vector is twice the size of the input feature vector $D_e$.

## 4.4 PlainNet

We implement and test a deep CNN architecture, called PlainNet for two main reasons. Firstly, we want to evaluate and compare its performance against the proposed *Modular*-CNN on the building classification task. Moreover, we want to test the effect of the proposed NL and ETL layers on a deeper and different CNN architecture. PlainNet is a version of the plain-net for $n = 1$ defined for the CIFAR-10 experimentation in [14]. More specifically, the PlainNet consists of stacks of two $(3 \times 3)$ convolutional layers for each feature map size. The feature map size is progressively halved from $D$ to $D/4$, while the number of filters is doubled from 16 to 64. Instead of a pooling layer, a convolutional layer with a stride of 2 is chosen for feature map downsampling. The linear mapping is performed at the top of the PlainNet without a decrease in feature vector dimensionality. The architecture of PlainNet is presented in Fig. 4.7.

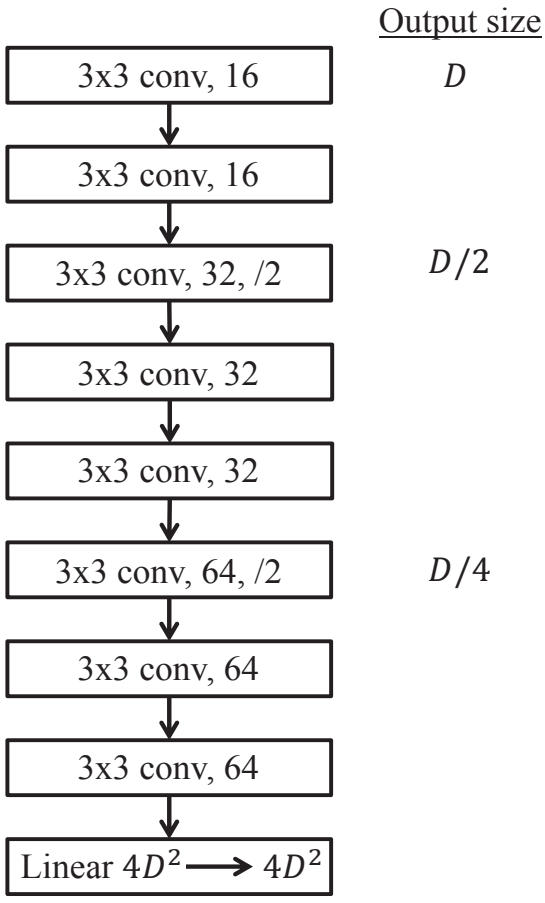



Figure 4.7: Architecture of PlainNet. In the convolutional layers, the number of output filters is shown last, while "/2" means a stride of 2. The ReLU layers after each convolutional and linear layer are omitted.

## 4.5 Evaluation of CNN architectures and novel layers

In this section, we present the implementation details and optimisation procedure of our *Modular*-CNN. Moreover, we analyse and compare the effects of our proposed layers on the performances of the *Modular*-CNN and PlainNet on the Athens remote sensing dataset. Finally, we propose the combination of the Enhanced HOG-LBP building detector, proposed in the previous chapter, and the *Modular*-CNN in order to create a very accurate and robust building detector.

### 4.5.1 Implementation details

The construction of the optimal to the building classification task *Modular*-CNN follows a *module*-based optimisation scheme. This means that each *module* is initially introduced to the current CNN and its hyper-parameters are optimised. In the case that the addition of the *module* is beneficial to the performance of the CNN, the *module* is added permanently in the CNN. Then, the hyper-parameters of the added *module* are kept fixed, while new *modules* are introduced to the CNN. In the case that the addition of the *module* leads to performance degradation, the optimisation procedure terminates and the current CNN without the added *module* is returned.

Each *module* is optimised with respect to the hyper-parameters of the basic layers that it consists of. These hyper-parameters are the convolution type, the convolutional filter size, the number of convolutional filters, the activation function, the pooling type and the pooling size and stride. Only the activation function is fixed to the ReLU unit as it has been shown in the literature that it outperforms other activation functions, while the other hyper-parameters are optimised using the hyper-parameter value pool of Table 4.1.

Table 4.1: CNN hyper-parameter value pool.

| Hyper-parameter | Value |
|---|---|
| Convolution type | Local \| Full |
| Filter size | $1 \times 3$ \| $3 \times 3$ \| $3 \times 1$ \| $1 \times 5$ \| $5 \times 5$ \| $5 \times 1$ \| $1 \times 7$ \| $7 \times 7$ \| $7 \times 1$ |
| Number of filters | 25 \| 50 \| 75 \| 100 |
| Pooling type | Max \| Average |
| Pooling size and stride | size $= 2 \times 2$, stride $= 2$ \| size $= 3 \times 3$, stride $= 1$ |

More specifically, two types of convolution are examined, the local convolution and the full convolution described in [205]. The full convolution is mostly proposed for dense predictions since it has the ability to output features of various sizes, not necessarily smaller than the input size. In our implementation of the full convolution, we choose the output size to be equal to the input size, no matter how large filters are employed by adding appropriate zero padding. We also experiment with various symmetrical and

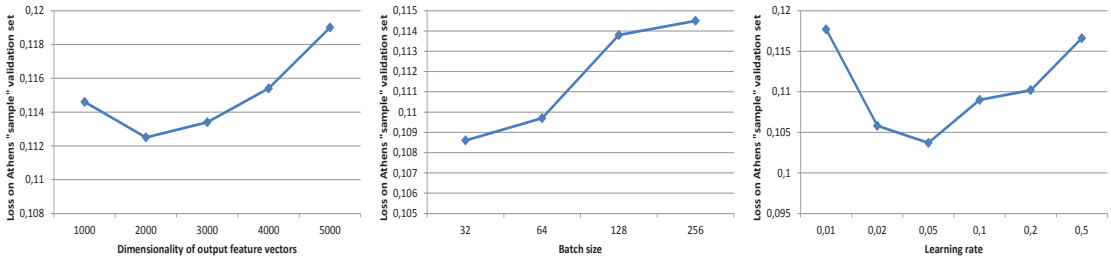
asymmetrical convolutional filters in an attempt to extract useful features from the images. Furthermore, we test both max-pooling and average-pooling using either a kernel of size $2 \times 2$ with a stride of 2 (i.e. non-overlapping pooling) or a kernel of size $3 \times 3$ with a stride of 1 (i.e. overlapping pooling). It has been shown that overlapping pooling can prevent over-fitting [11].

Other hyper-parameters that affect the *Modular*-CNN architecture are optimised with respect to the Athens "sample" validation set, as it is presented in Section 2.5.1. The dimensionality $N$ of the extracted feature vectors that are introduced to the second fully connected layer of the *Modular*-CNN is optimised to the value of 2000. Linear and Euclidean layers for the classification of feature vectors are tested and we conclude that the Euclidean layer that performs classification based on the Euclidean distances between feature vectors outperforms the linear layer. A second reason why the Euclidean layer is preferable is the fact that it employs the $l_2$-norm as a distance function. Thus, the novel ETL layer that transforms the feature vectors to their Euler representation and requires the $l_2$-norm as a distance function for the classification of the feature vectors can be directly employed. Dropout that omits the response of neurons with a certain probability by changing their values to zero is examined but it leads to sub-optimal results and therefore, it is not employed.

Moreover, there are hyper-parameters that affect the CNN training procedure and need to be optimised. The algorithm that is employed for the *Modular*-CNN training is the stochastic gradient optimisation method. The most important hyper-parameters that affect the stochastic gradient optimisation method are the size of the mini-batch, the learning rate and the momentum. The size of the mini-batch defines the number of training samples that are fed simultaneously to the CNN for training and it is optimised to the value of 32. The learning rate defines the speed with which the weights of the CNN are adjusted during back-propagation and it is optimised to the value of 0.05. Finally, the momentum is used to diminish the fluctuations in weight changes over consecutive iterations and it is fixed to the value of 0.7. Given the error function $E(\mathbf{w})$ that depends on the CNN weight vector $\mathbf{w}$, the change of weights $\Delta\mathbf{w}(t+1)$ at iteration $t+1$ can be characterised using the learning rate $\eta$ and the momentum $\mu$ as

$$\Delta\mathbf{w}(t+1) = -\eta\frac{\vartheta E}{\vartheta \mathbf{w}} + \mu\Delta\mathbf{w}(t) \tag{4.4}$$

The experimentation on the dimensionality of output feature vectors, the batch size and the learning rate is performed on the Athens "sample" validation set and it is presented in Fig. 4.8. The training of the CNN with each hyper-parameter configuration is performed for a maximum number of iterations or epochs, which we set equal to 100. In each epoch, the training set is introduced to the CNN in mini-batches for training and then the CNN is evaluated on the Athens "sample" validation set. During the evaluation, the loss on the Athens "sample" validation set, which is equal to the average negative log-likelihood of




(a) (b) (c)

Figure 4.8: Experimentation on the Athens "sample" validation set with respect to (a) the dimensionality of output feature vectors, (b) batch size and (c) learning rate.

each validation sample to belong to the correct class, is computed. The training phase is terminated when the loss on the validation set does not decrease for 5 consecutive epochs. Afterwards, the performance of the CNN for the epoch, in which the minimum loss was achieved, is preserved. This strategy is employed in order to prevent over-fitting of the CNN to the training data, since after a few epochs the loss on the training set keeps decreasing, while the loss on the validation set starts increasing. This typical behaviour of a neural network is reported during our experiments with the *Modular*-CNN and is presented in Fig. 4.9.

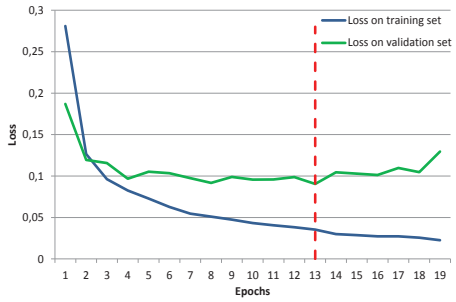



Figure 4.9: Evolution of training and validation loss during the *Modular*-CNN training. Over-fitting is prevented by stopping the CNN training in the epoch shown with a red dashed vertical line.

The CNN training is repeated for 3 rounds, where a new initialisation/reset of the weights is performed in the beginning of each new round. The average performance of the CNN on the Athens "sample" validation set is computed and used for the selection of the optimal hyper-parameter configuration. This procedure enables us to draw more reliable conclusions about the accuracy of the tested CNN on the current classification task.

The optimisation of the *Modular*-CNN is performed without the novel NL and ETL layers. These layers are added afterwards to the CNN architecture and the optimisation of the parameter $\alpha$ of the ETL layer is performed over the already optimised *Modular*-CNN, whose hyper-parameters are kept fixed.

A disadvantage of our proposed *Modular*-CNN that needs to be reported is the large number of parameters that need to be optimised during the training phase. The large

number of parameters is attributed to the fully connected linear layers of the proposed CNN architecture as the dimensionality of the feature vector that the convolutional part of the CNN outputs is quite large. This is because of the shallowness of the *Modular*-CNN that does not allow a significant downsampling of the input feature maps and this is the main reason why the first linear layer is employed (i.e. reduce the dimensionality of the computed feature vectors).

### 4.5.2 *Modular*-CNN optimisation on Athens dataset

This section presents the experiments for the optimisation of the *Modular*-CNN based on its performance on the building classification task. More specifically, we employ the Athens training set and "sample" validation set in order to train and optimise the hyper-parameters of the *modules* of the *Modular*-CNN architecture respectively. The small size of the images of the Athens training set (i.e. $20 \times 20$ pixels), makes it a perfect candidate for the testing of the proposed shallow *Modular*-CNN.

A successful training of a CNN usually requires thousands of training images. Since the Athens training set is small (i.e. 2100 images), it is augmented by taking horizontal and vertical flips of the image patches. The images of this dataset consist of 4 spectral channels, namely the red, green, blue and near-infrared channels. In our experiments, we employ the YUV colour space and the near-infrared colour channel, since this spectral configuration yields the best classification results in the validation set. Generally, we observe that the CNN performance is strongly affected by the selected colour channels, which makes the selection of an optimal colour representation for the images critical to the performance of a CNN. The dataset is normalised to have zero mean and unity variance before it is fed to the CNNs for training and testing.

Experiments are performed in order to optimise the *modules* that the *Modular*-CNN consists of. The optimal performance on the Athens "sample" validation set is achieved by the *Modular*-CNN architecture depicted in Fig. 4.10. The addition of a fourth *module* leads to a drop in the generalisation ability of the trained *Modular*-CNN. Moreover, we examine the effect of the novel NL and ETL layers, when added to the *Modular*-CNN and PlainNet. The average loss on the validation set of the proposed *Modular*-CNN and PlainNet with the addition of the NL and ETL layers for values of $\alpha$ that lie in the range $[0 - 1.9]$ is depicted in Fig. 4.11. Furthermore, Table 4.2 summarises the performance of the *Modular*-CNN and PlainNet with the addition of the proposed NL and ETL layers on the validation set.

From Table 4.2 and Fig. 4.11, it can be concluded that the addition of the novel NL and ETL layers to the *Modular*-CNN and PlainNet leads to an improvement in their performance on the Athens "sample" validation set. More specifically, a drop in the error on the validation set by about 10.4% and 8.2% in the cases of PlainNet and *Modular*-CNN

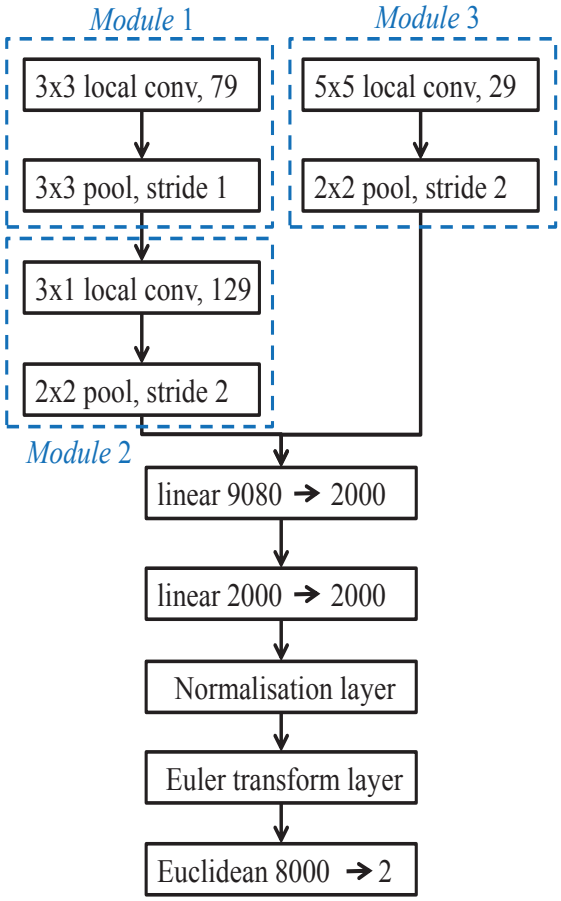
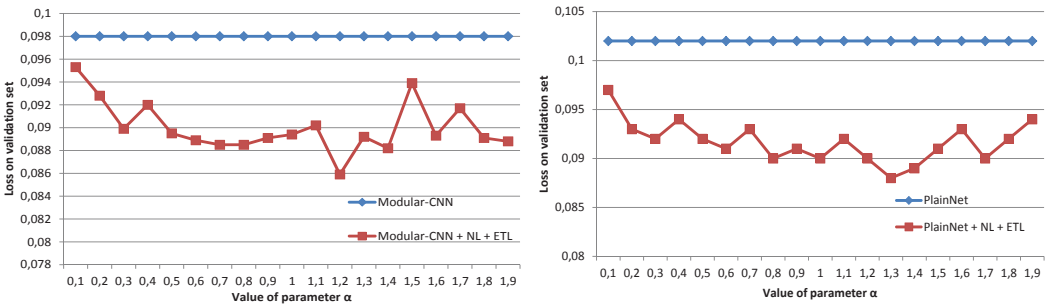



Figure 4.10: Optimal *Modular*-CNN architecture for the Athens remote sensing dataset. In the convolutional layers, the number of output filters is shown last, while in the fully connected blocks, the transition from $M$ to $N$ features is depicted. The ReLU units after each convolutional and linear layer are omitted.




(a) (b)

Figure 4.11: Performance of the *Modular*-CNN and PlainNet with and without the novel NL and ETL layers with respect to the average loss on the Athens "sample" validation set.

respectively is noticed. Furthermore, the addition of the ETL layer reduces significantly the average loss on the validation set, thus increasing the generalisation power of the *Modular*-CNN and PlainNet. The optimal value of the parameter $\alpha$ for the ETL layer is included next to the corresponding error in Table 4.2. A comparison between the Plain-

Table 4.2: Classification performance on the Athens dataset. The results are presented in the format: mean ± std.

| Method | Error (%) | # params |
|---|---|---|
| PlainNet [14] | 3.404 ± 0.066 | ≈ 2.685M |
| PlainNet+NL | 3.118 ± 0.084 | ≈ 2.688M |
| PlainNet+NL+ETL | 3.049 ± 0.075 ($\alpha$=1.3) | ≈ 2.695M |
| *Modular*-CNN | 3.123 ± 0.062 | ≈ 22.2M |
| *Modular*-CNN+NL | 2.871 ± 0.079 | ≈ 22.21M |
| *Modular*-CNN+NL+ETL | 2.866 ± 0.071 ($\alpha$=1.2) | ≈ 22.22M |

Net and the *Modular*-CNN reveals that our proposed *Modular*-CNN slightly outperforms PlainNet with and without the addition of the novel NL and ETL layers. The improved performance of the *Modular*-CNN with respect to the PlainNet can be attributed to the increased number of parameters of the *Modular*-CNN that can significantly enhance its learning abilities.

In order to compare our proposed *Modular*-CNN and PlainNet and evaluate the effect of the proposed NL and ETL layers on different classification tasks, we employ two benchmark datasets, namely the CIFAR-10 and MNIST datasets. The interested reader can see the results of the experimentation and the conclusions that were drawn in Appendix B.

### 4.5.3 *Modular*-CNN testing on Athens dataset

In an attempt to take advantage of the state-of-the-art performance of a CNN on image classification tasks and its ability to automatically generate discriminatory features, we employ the *Modular*-CNN with the novel NL and ETL layers for the detection of buildings in the satellite images of the Athens test set. Employing the proposed *Modular*-CNN in the Athens "image" validation set, we notice that the *Modular*-CNN can work complementary with the optimal object-based HOG-LBP building detector with Platt scaling (M5) that was presented in the previous chapter as the locations of the false detections differ between the two algorithms. Applying the *Modular*-CNN directly to the test images of the Athens dataset leads to sub-optimal results, as it is reported on Table 4.3. The sub-optimal performance of the *Modular*-CNN on the building detection task can be attributed to the small training set used for the *Modular*-CNN training. Although data augmentation is employed, it is inadequate to lead to state-of-the-art performance on the building detection task. However, allowing the proposed *Modular*-CNN to classify only the positive detections returned by the method M5, we avoid the computational overhead of having to classify overlapping image patches at multiple scales and we also suppress the appearance of new false detections.

Thus, a new building detection methodology is proposed that takes advantage of both the power of the HOG-LBP features and the discrimination abilities of the *Modular*-CNN. A flowchart of the proposed building detection methodology is presented in Fig.

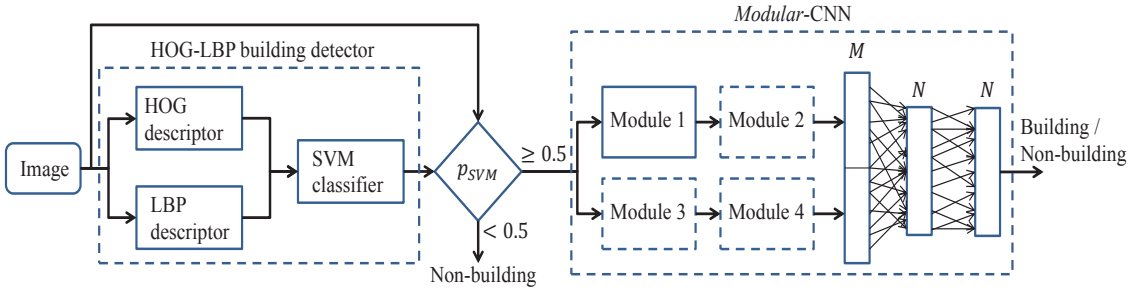



Figure 4.12: Flowchart of the proposed building detection methodology.

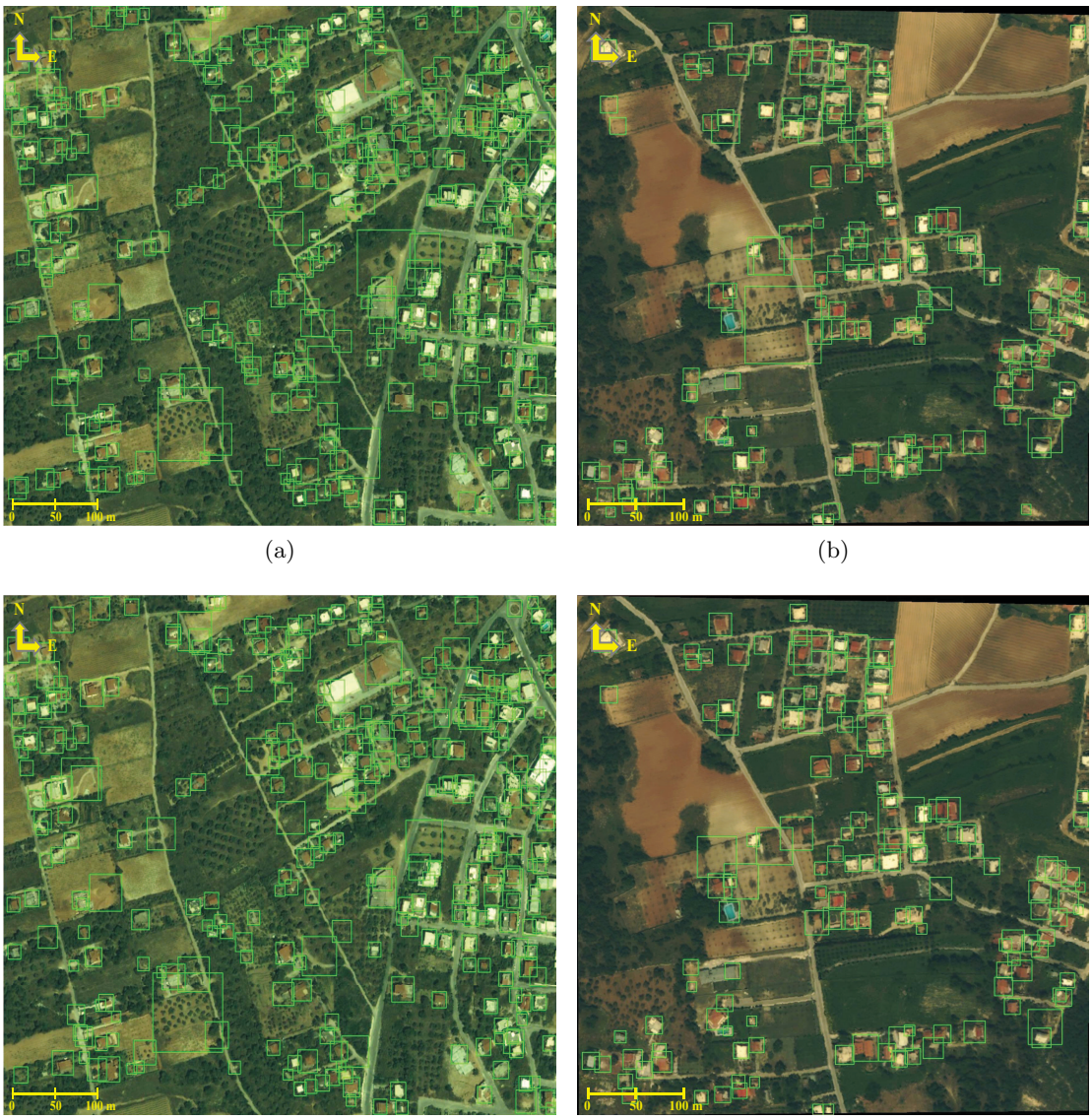
4.12. The combined building approach initially employs the HOG-LBP building detector (M5) to extract candidate building regions, which are then classified using optimised for the building detection task *Modular*-CNN. The combined building approach achieves an average object-based precision of 0.867 in the Athens test set, as it is shown in Table 4.3, allowing for a reduction of the false alarms produced by the HOG-LBP detector by 5.3% and an improvement in the metric of object-based $F_1$-score by almost 2.2%.

Furthermore, we employ the pixel-based HOG-LBP building detector (M9) that was defined in the previous chapter and it consists of the method M5 and a set of region refinement processes. The goal is to evaluate the effect of the *Modular*-CNN on the performance of the method M9 on the Athens test set. The results are summarised in Table 4.3, where it can be concluded that an improvement in the metric of object-based $F_1$-score by 1.1% and a slight drop in the metric of pixel-based $F_1$-score by almost 0.4% is observed, when the *Modular*-CNN is employed.

Table 4.3: Performance of building detectors on Athens test set. The results are presented in the format: mean ± std. M-CNN: *Modular*-CNN

| Method | Object-based | | | Pixel-based | | |
|---|---|---|---|---|---|---|
| | Recall | Precision | $F_1$-score | Recall | Precision | $F_1$-score |
| M5 | **0.934±0.114** | 0.823±0.102 | 0.865±0.076 | **0.945±0.066** | 0.135±0.055 | 0.23±0.079 |
| M-CNN | 0.898±0.066 | 0.721±0.134 | 0.789±0.072 | 0.861±0.063 | **0.248±0.077** | **0.376±0.086** |
| M5+M-CNN | 0.917±0.121 | **0.867±0.079** | **0.884±0.078** | 0.92±0.108 | 0.157±0.055 | 0.261±0.073 |
| M9 | 0.717±0.069 | 0.73±0.1 | 0.717±0.05 | 0.556±0.06 | **0.494±0.082** | **0.516±0.043** |
| M9+M-CNN | **0.726±0.074** | **0.738±0.095** | **0.725±0.05** | **0.56±0.06** | 0.485±0.075 | 0.514±0.045 |

We demonstrate the ability of the combined *Modular*-CNN and HOG-LBP object-based building detector (M5 + *Modular*-CNN) to suppress false detections in two images of the Athens test set in Fig 4.13. Furthermore, we illustrate the performance of the method M9 with and without the use of the proposed *Modular*-CNN in Fig. 4.14. The conclusion that can be drawn is that the addition of the proposed *Modular*-CNN in the object-based and pixel-based building detectors introduced in the previous chapter has a beneficial effect on their performances on the Athens test set as it manages to reduce the number of false detections.




(a) (b)

(c) (d)

Figure 4.13: Detections shown as green bounding boxes in two test images of the Athens dataset from (a), (b) the method M5 and (c), (d) the combined approach of our *Modular*-CNN and the method M5.

## 4.6 Summary

This chapter proposes a shallow CNN architecture, called *Modular*-CNN, that can be employed for the building detection task. When a dataset consists of small images, a shallow CNN is adequate to correctly classify the dataset. However, for datasets with larger images, the *Modular*-CNN can be extended with the addition of more parallel or sequential *modules*.

Furthermore, two novel layers (i.e. NL and ETL) are proposed that can be added to the *Modular*-CNN architecture in order to improve its performance. The NL layer harnesses the power of both raw and normalised features with the purpose to improve the generali-




(a) (b)

(c) (d)

Figure 4.14: Detected building regions from (a), (b) the method M9 and (c), (d) the combined approach of our *Modular*-CNN and the method M9 in two images of the Athens test set. True positives are shown in green, false positives are shown in blue and missed pixels are presented in red.

sation power of the *Modular*-CNN, while the ETL layer takes advantage of a cosine-based distance function to transform the feature vectors to their Euler representation. The Euler transformed feature vectors are shown to improve the robustness of the *Modular*-CNN to the effects of noise and outliers.

Experiments are conducted and verify that the performances of the proposed shallow *Modular*-CNN and a deeper CNN, called PlainNet, on the challenging Athens dataset are improved, when the novel NL and ETL layers are introduced. However, the results are not conclusive as the high number of parameters of the *Modular*-CNN improves its learning capacity but renders the optimisation of the *Modular*-CNN more difficult. Moreover, the

fact that the *Modular*-CNN is a shallow network means that it can usually be optimised more easily and achieve a better performance than a deeper one.

Finally, we propose a new building detector based on the complementary behaviour of the HOG-LBP building detector (M5), introduced in the previous chapter, and the *Modular*-CNN. The new building detector applies the classification abilities of the optimised to the building detection task *Modular*-CNN on the positive detections of the method M5 in order to avoid unnecessary computational overhead and the detection of additional false alarms. The experimentation on the Athens dataset suggests that the object-based building detection method M5 and the pixel-based building detection method M9 with the addition of the *Modular*-CNN consist the optimal object-based and pixel-based building detection methodologies respectively that are developed in the framework of this thesis.

In Chapters 3 and 4, we address the first task of urban change monitoring, namely the building detection task. In the next chapter, we deal with the second task of urban change monitoring, which concerns the detection of building changes. As we show in the next chapter, the building masks that the optimal building detection methodologies of Chapter 4 provide, can be proved valuable for the identification of building changes.

# CHAPTER 5

## Change Detection

In the previous two chapters, we dealt with the problem of building detection in order to acquire accurate and robust building masks (BMs) that depict the locations of buildings in an image. In this chapter, we address the other task of urban change monitoring, namely change detection. We want to take advantage of the building masks that our proposed optimal pixel-based building detector outputs in order to acquire more accurate building change detection results.

Change detection in computer vision is generally concerned with the problem of identifying the type and magnitude of changes between two images depicting the same scene and acquired at different temporal periods. Change detection can find application in several fields that require the detection and modelling of changes between two or more images, such as urban mapping, surveillance, manufacturing (quality control) and medical diagnostics. Depending on the problem that needs to be solved, some changes are important and need to be modelled, while other changes are irrelevant and they should be discarded. In the framework of this thesis, we are concerned with the problem of detecting changes in the building infrastructure between two or more images. Building change detection can be a very challenging task, especially in remote sensing applications as noise, local illumination variations, low contrast, distortions and spectral and spatial resolution differences that are caused from the sensors that capture the compared images set significant obstacles in an accurate building change detection.

We approach the problem of change detection from two main directions. The first direction is the top-down approach, where the objects of interest (i.e. buildings) are initially detected in the images before the comparison of the extracted objects is performed. Top-down approaches are usually hard to implement as the objects of interest need to be accurately detected before their comparison. Objects lost during the detection procedure cannot be recovered afterwards. The optimal pixel-based building detector that was proposed in Chapter 4 computes and provides the building masks in order to enable the comparison of the building masks for change detection purposes. Therefore, in the top-down approach,

we study and analyse how the extracted building masks can assist in the development of an accurate building change detector.

The second direction that we follow is the bottom-up approach. In this approach, changes are initially detected between the images and then these changes are analysed and modelled and only those that correspond to buildings are preserved, while the others are discarded. Change detection techniques can be directly applied to images in order to detect general changes, but the identification of the type of changes is usually a challenging task. In this framework, we investigate pixel-based change detection techniques due to their computational efficiency. Unfortunately, these techniques suffer from two significant limitations; the need for an optimal threshold that differentiates true changes from false ones and their sensitivity to noise. To counter the effect of noise and improve the robustness of the pixel-based change detection methods, we propose a novel Change Amplification (CHAM) algorithm that compares two images and amplifies their phase differences. This algorithm is inspired by the work of [206] in the magnification of small motions in video sequences. We demonstrate the beneficial effect of the CHAM algorithm on the Athens remote sensing dataset and the Aerial Imagery Change Detection (AICD) dataset.

Finally, we take advantage of the state-of-the-art performance of Convolutional Neural Networks (CNNs) on several applications and we propose an accurate and robust building change detector based on CNNs. More specifically, the proposed CNN Building Change Detector (CNN-BCD) compares pixels between two images by considering pixel neighbourhoods around them and evaluates the magnitude and type of changes. We evaluate the performance of the CNN-BCD algorithm on the Athens dataset and we compare it with the other tested pixel-based change detection techniques.

The main contributions of this chapter are pointed out below:

- The novel CHAM algorithm that amplifies changes between images by transforming them to the Fourier domain and magnifying their phase differences.
- The new robust CNN-BCD algorithm that identifies building changes based on the power and discrimination abilities of CNNs.

The structure of the chapter is as follows. Sections 5.1 and 5.2 present the top-down and bottom-up approaches respectively in order to achieve accurate building change detection results. Section 5.3 presents the proposed CNN-BCD algorithm, while Section 5.4 presents various illumination normalisation techniques for improved change detection results, including the theory behind the proposed novel CHAM algorithm that allows the enhancement of differences between the compared images. In Section 5.5, we evaluate and compare the effect of the tested illumination normalisation techniques and the proposed CHAM algorithm on the performance of the pixel-based change detection methods and the CNN-BCD algorithm. Moreover, we compare the change detection performance of the top-down and bottom-up approaches on the Athens and the AICD datasets. Finally, in

Section 5.6, we conclude the work by summarising the main contributions of this chapter in the change detection task.

## 5.1 Top-down approach

A top-down approach consists of two significant processing steps. The first step is the detection of the objects of interest, whose changes we want to model. Any type of image segmentation algorithm can be employed in this step in order to acquire a binary mask with the locations of the objects of interest. An accurate and robust segmentation of the objects of interest is usually a challenging task and cases of under- or over-segmentation can significantly affect the extracted binary masks. Furthermore, objects that are not detected during this phase cannot be recovered in a later stage, thus leading to sub-optimal change detection results. As a result, the accurate detection of the objects of interest is of paramount importance in the framework of a top-down approach. The second processing step of a top-down approach is the actual change detection. Given the successful and accurate detection of the objects of interest, this processing step can be trivially solved by simply subtracting the binary masks that depict the objects of interest. The diagram of a top-down approach, in the case of building change detection that concerns this thesis, is presented in Fig. 5.1.

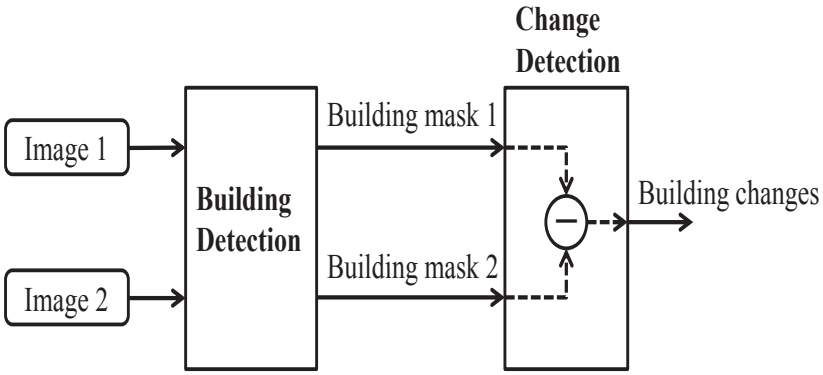



Figure 5.1: The diagram of a top-down approach applied in the building change detection task.

In the previous chapters, we analysed and compared several building detection methodologies and came up with a pixel-based HOG-LBP building detector with the addition of the *Modular*-CNN that achieves the optimal results in the building detection task. In this chapter, we employ the proposed pixel-based building detector as the first processing step of a building change detection procedure based on the top-down approach. The purpose is to extract accurate binary building masks from two images, whose changes should be detected and modelled and subtract the building masks in order to acquire a binary change mask that depicts the differences in the building infrastructure between the two images. The binary change mask is then filtered using morphological closing and opening operations so as to remove small regions that do not correspond to true changes and "holes" in

the change mask.

## 5.2 Bottom-up approach

A bottom-up change detection approach can be split in two main processing steps. The first step is change detection and concerns the employment of any change detection methodology that compares two images and quantify their differences. A change detection algorithm that deals directly with the images faces significant limitations. Noise and local illumination variations can severely affect the change detection procedure, leading to sub-optimal results. Algorithms that pre-process images in order to make their illumination even are usually employed prior to the image comparison. The second processing step of a bottom-up change detection approach, which is actually the most challenging step, concerns the modelling of changes in order to identify which changes are important with respect to the task that needs to be solved. The diagram of a bottom-up approach in the case of building change detection is presented in Fig. 5.2.

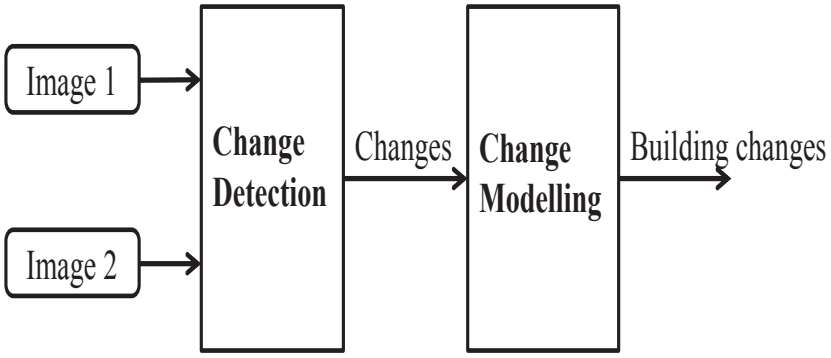



Figure 5.2: The diagram of a bottom-up approach applied in the building change detection task.

For the first processing step of a bottom-up building change detection approach, we consider and analyse several pixel-based change detection methodologies. The purpose is to identify all true changes, while managing to discard the largest amount of false changes. Although pixel-based methods evaluate the magnitude of changes, most methods give no information on the type of changes. Given a set of $n$ multi-spectral images $I_{ms}^{(1)}, I_{ms}^{(2)}, ..., I_{ms}^{(n)}$ and their conversion to grayscale images $I_g^{(1)}, I_g^{(2)}, ..., I_g^{(n)}$, the task is to identify the differences between the images taken in pairs, meaning that the $I^{(n-1)}$ image is compared with the chronologically subsequent image $I^{(n)}$. The pixel-based change detection techniques that are evaluated in this chapter are:

- **Image differencing:** The magnitude of changes between a pair of images is evaluated based on the absolute differences of the grayscale pixel values of the compared images.

$$D^{(n-1)} = |I_g^{(n)} - I_g^{(n-1)}| \tag{5.1}$$

- **Image regression:** This method performs regression between the corresponding pixels of the compared images before the computation of changes. In our work, we choose to employ linear regression. This means that the parameters $a$ and $b$ of the linear equation $y = ax + b$ are optimised based on how well the grayscale pixel values of image $I_g^{(n)}$ fit the pixel values of image $I_g^{(n-1)}$. The result of the linear regression is a new image $\hat{I}_g^{(n)} = aI_g^{(n)} + b$ and the change detection is performed based on the absolute differences of the intensity values of images $I_g^{(n-1)}$ and $\hat{I}_g^{(n)}$.

$$D^{(n-1)} = |\hat{I}_g^{(n)} - I_g^{(n-1)}| \tag{5.2}$$

- **Median background subtraction:** This algorithm detects changes by subtracting the images from a background image [121]. The background image is formed by considering the corresponding pixels in every location of the compared images and computing the median values. In the case of $n$ compared images of size $K \times L$ in pixels, the background image $B$ is formed by the following equation, where the operation "*med*" defines the median operation:

$$B(k,l) = med(I_g^{(1)}(k,l), I_g^{(2)}(k,l), ..., I_g^{(n)}(k,l)),\ k = 1...K,\ l = 1...L \tag{5.3}$$

  As a result, the background image is formed based on the information from more than two images. In the special case, where only two images are employed, the median operation of Eq. (5.3) is transformed to a "*min*" operation and therefore, the background image contains the smallest values of the corresponding pixels of the two compared images. Finally, the magnitude of changes in the median background subtraction algorithm is given by

$$D^{(n-1)} = |I_g^{(n-1)} - B| \tag{5.4}$$

- **Change Vector Analysis (CVA):** The CVA algorithm detects changes by exploiting all multi-spectral information of images. Given that each pixel at position $(k,l)$ of a multi-spectral image $I_{ms}^{(n)}$ of size $K \times L$ is described by a spectral vector of the form $\mathbf{x}^{(n)}(k,l) = [x_0^{(n)}(k,l), x_1^{(n)}(k,l), .., x_C^{(n)}(k,l)]$, where $C$ the number of multi-spectral channels of an image, the magnitude of change between the corresponding pixels of the pair of images $I_{ms}^{(n-1)}$ and $I_{ms}^{(n)}$ is computed by taking the square root of the Euclidean distance of their spectral vectors

$$D^{(n-1)}(k,l) = \|\mathbf{x}^{(n)}(k,l) - \mathbf{x}^{(n-1)}(k,l)\| = \sqrt{\sum_{c=1}^{C}(x_c^{(n-1)}(k,l) - x_c^{(n)}(k,l))^2} \quad (5.5)$$

The angles (see Eq. 5.6) that the spectral vectors form, can hold valuable information about the type of change, but in this work we decide not to exploit such information.

$$\theta_c(k,l) = \arctan\left(\frac{x_c^{(n)}(k,l) - x_c^{(n-1)}(k,l)}{x_{c-1}^{(n)}(k,l) - x_{c-1}^{(n-1)}(k,l)}\right) \text{ for } c = 2...C \quad (5.6)$$

- **Normalised Cross-Correlation (NCC):** NCC belongs to the subcategory of texture analysis based methods of the pixel-based change detection methods that does not consider individual pixels but compares neighbourhoods of pixels. Thus, the NCC algorithm can be more robust to noise than the other tested methods. The NCC algorithm considers a neighbourhood of $p \times p$ pixels around each pixel of the compared pair of grayscale images $I_g^{(n-1)}$ and $I_g^{(n)}$ and the value of NCC is computed as an indicator of the similarity between the two pixel neighbourhoods $I_{p\times p}^{(n-1)}$ and $I_{p\times p}^{(n)}$ by employing

$$NCC(I_{p\times p}^{(n)}, I_{p\times p}^{(n-1)}) = \frac{1}{p^2}\frac{(I_{p\times p}^{(n)} - \bar{I}_{p\times p}^{(n)})(I_{p\times p}^{(n-1)} - \bar{I}_{p\times p}^{(n-1)})}{\sigma_{I_{p\times p}}^{(n)}\sigma_{I_{p\times p}}^{(n-1)}} \quad (5.7)$$

The terms $\bar{I}_{p\times p}^{(n)}$ and $\sigma_{I_{p\times p}}^{(n)}$ represent the mean and standard deviation respectively of the pixel neighbourhood $I_{p\times p}^{(n)}$.

Special attention should be paid so as to correctly handle the range of values that a change detection methodology outputs and determine whether higher values correspond to bigger or smaller changes. Image differencing, image regression, median background subtraction and CVA output change maps with values in the range $[0, 1]$ and changes correspond to higher values. On the contrary, the NCC algorithm outputs a change map with values in the range $[-1, 1]$ and changes correspond to lower values.

Furthermore, selecting an optimal threshold that transforms the change maps to binary change masks is a challenging problem for pixel-based methodologies. A sub-optimal threshold can cause mis-detections or over-detections that can significantly affect the change detection results. In this work, we employ the Otsu's method to automatically select the optimal threshold that separates the "change" and "no-change" classes by minimising their combined spread or intra-class variance. The final binary change masks are processed using morphological closing and opening operations in order to remove small regions that cannot correspond to real changes and "holes" inside changed regions.

The second and most challenging step of a bottom-up change detection approach is change modelling that enables the identification of the type of changes. To this end, we employ the binary building masks that the optimal pixel-based building detector of Chapter 4 provides. The binary building masks are employed in order to select only the changes that correspond to building areas and form binary building change masks. A binary building change mask is finally filtered using morphological closing and opening operations so as to remove small regions that do not correspond to true changes and "holes" in the change mask.

## 5.3 CNN-BCD algorithm

Motivated from the state-of-the-art performances of CNNs on several applications and their ability to automatically generate discriminatory features without the intervention of a human expert, we propose a novel building change detection algorithm based on CNNs. The proposed CNN-BCD algorithm can be considered to belong to a category of pixel-based change detection methods that employ classification techniques. The CNN-BCD algorithm does not operate at the pixel level, but at the window/block level as it considers and compares pixel neighbourhoods around the central pixels that need to be classified. Therefore, the CNN-BCD algorithm can be considered more robust to the effects of noise and local illumination variations. The proposed CNN-BCD algorithm is presented in Fig. 5.3.

The input of the proposed CNN-BCD algorithm is the corresponding pixel neighbourhoods between two compared images and the output is the decision about whether the central pixels of the neighbourhoods present building changes. As a result, the CNN-BCD algorithm does not only evaluate the magnitude but also the type of changes as it accepts only those that correspond to changes in the building infrastructure. The architecture of the proposed CNN-BCD algorithm is based on the notion of Inception module that was initially introduced in [207] and later used in [97] for the creation of GoogleNet. The Inception module consists of several parallel processing layers in order to produce different features from the same input data. The outputs of the processing layers are then concatenated into a final output. In this architecture, we employ a parallel configuration of a $3 \times 3$ convolutional layer and a $5 \times 5$ convolutional layer with zero padding of 1 pixel in order for the output feature maps to have the same reduced dimensionality with respect to the dimensionality of the input feature maps.

Furthermore, we employ $1 \times 1$ convolutions to reduce the number of feature maps and CNN parameters in the upper layers of the CNN-BCD architecture. The use of the $1 \times 1$ convolutional layers was proposed in [13] as these layers have the advantage of increasing the depth of a CNN without adding significantly more computational complexity to the CNN architecture. The extracted feature maps from each pixel neighbourhood of an

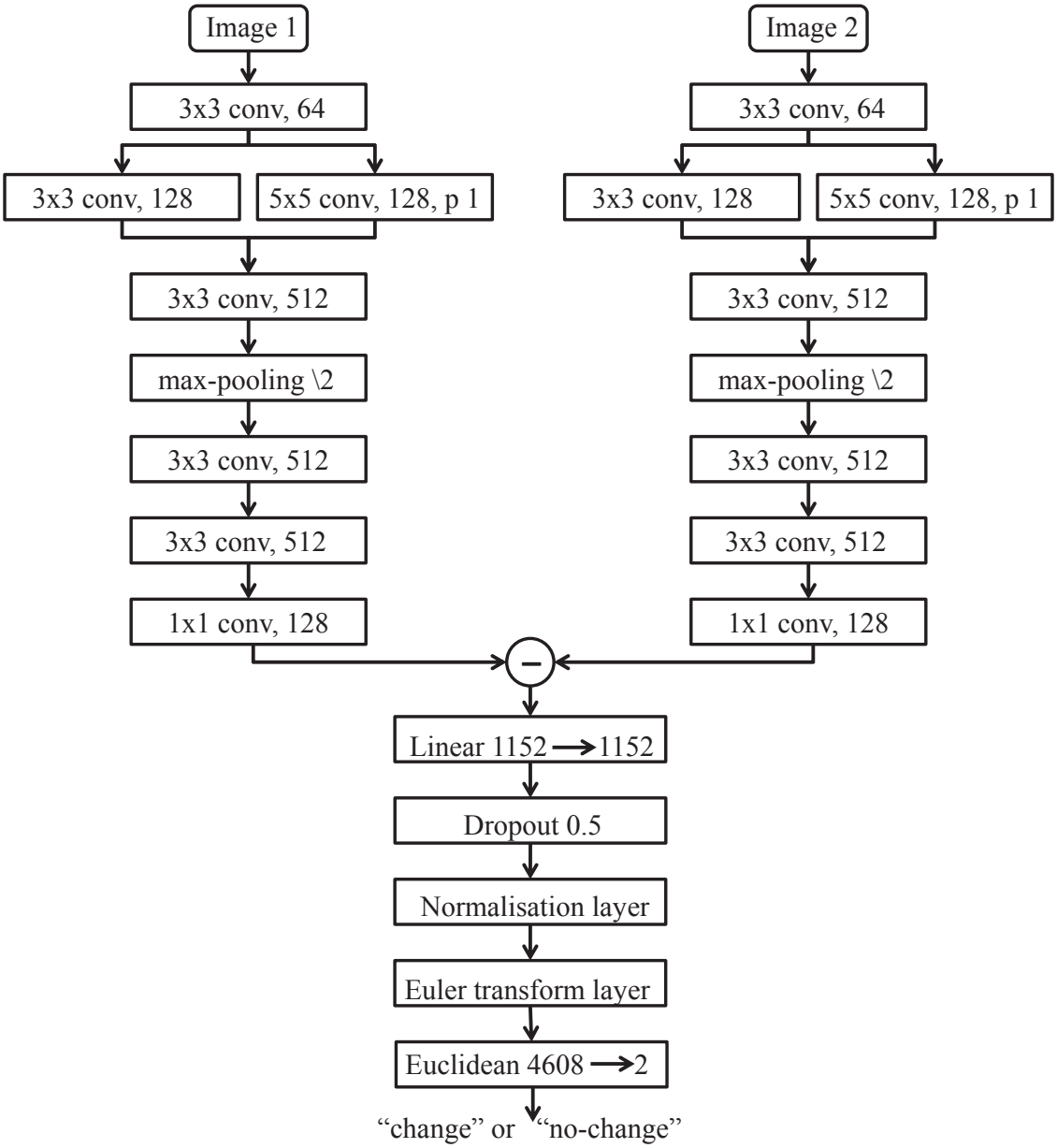



Figure 5.3: Architecture of the proposed CNN-BCD algorithm. The term "p 1" denotes zero padding with 1 pixel. ReLU units that are employed after each convolutional and linear layers are omitted.

image are subtracted before they are introduced to the linear layers. The subtraction of the feature maps is not widely used in the CNN literature, but we believe it is appropriate for the quantification of differences between the extracted feature maps from the input pixel neighbourhoods. Moreover, we employ a Dropout layer to omit the activation of random hidden units with a certain probability. Dropout has been shown to increase the generalisation ability of a CNN by preventing over-fitting during the CNN training [88]. Finally, we employ the novel normalisation (NL) and Euler transform (ETL) layers that we proposed in Chapter 4 to improve the accuracy and robustness of the CNN-BCD algorithm.

The output of the CNN-BCD algorithm is a probability that defines the confidence of the CNN-BCD algorithm that the central pixels of the compared neighbourhoods present a building change. Thus, a building change map that contains for each pixel its probability to describe a building change is formed. Otsu's thresholding is then employed to automatically define a threshold that separates the pixels that correspond to building changes from all other pixels. The binary building change mask is finally filtered using morphological closing and opening operations so as to remove small regions that do not correspond to true changes and "holes" in the building change mask.

The CNN-BCD algorithm has several advantages over the other tested change detection methodologies. Firstly, it is more robust to noise because it operates at the window level and thus it suppresses the effect of single noise-embedded pixels. The robustness of the CNN-BCD algorithm is also enhanced by the use of the ETL layer in its architecture. Furthermore, the CNN-BCD algorithm shows improved generalisation performance due to the use of the NL and the Dropout layers. Moreover, the CNN-BCD algorithm does not rely only on differences of pixel intensities or texture-based features, but it is able to create more complex features for the successful detection of building changes. Finally, the CNN-BCD algorithm is capable of detecting not only the magnitude, but also the type of changes (i.e. building infrastructure changes) due to the fact that it is trained using only the building changes as positive samples. Because of all these reasons, the CNN-BCD algorithm outperforms the other tested pixel-based methodologies, as it is shown in Section 5.5.5.

## 5.4 Image pre-processing for change detection

One of the first and most important tasks, when comparing images, is image registration. Image registration concerns the task of aligning images in a way that the same depicted objects are located exactly in the same image locations. Image registration is a hugely researched task and beyond the scope of this work and therefore, it will not be further analysed. A comprehensive review of image registration techniques is presented in Section 2.4.2. This section discusses illumination normalisation techniques that enable a more accurate detection of changes between compared images. Furthermore, the novel CHAM algorithm that pre-processes images to amplify their differences so that the changes become more distinct and easily recognisable is proposed.

### 5.4.1 Illumination normalisation

Illumination variations are the result of different sun angles during the capture of two images depicting the same scenery, but taken in different temporal periods and they can significantly affect the change detection results as they alter the intensity values of pixels.

Therefore, pixels that depict the same scenery and would otherwise have the same intensity present differences if illumination variations exist. As a result, the purpose of the illumination normalisation methods is to suppress the effect of the illumination variations.

The illumination normalisation methods that are analysed in this work are:

- **Normalisation:** The intensity values of an image are normalised so that they are in the range $[0, 1]$. Given a grayscale image $I$ and the "$min$" and "$max$" operations that compute the minimum and maximum intensity values respectively, the illumination normalised image $\hat{I}$ is the result of the following equation

$$\hat{I} = \frac{I - min(I)}{max(I) - min(I)} \tag{5.8}$$

  In the case of a colour image, the colour and intensity channels of the image are differentiated, only the intensity channel is normalised and the illumination normalised colour image is formed by merging the colour and intensity channels.

- **Standardisation:** In standardisation, the new image is formed by subtracting its mean value and dividing the result with the standard deviation. This results in an illumination normalised image that has zero mean and unity variance. Given an image $I$, its mean value $\bar{I}$ and its standard deviation $\sigma_I$, the illumination normalised image $\hat{I}$ is

$$\hat{I} = \frac{I - \bar{I}}{\sigma_I} \tag{5.9}$$

  In the case of a colour image, we apply standardisation in every channel of the image separately. Standardisation is the only illumination normalisation method that does not output pixel values in the range $[0, 1]$.

- **Histogram equalisation:** Histogram equalisation is based on the idea of improving the contrast of an image by spreading its intensity values in the allowed range [208]. As a result, a histogram of intensities is formed and then flattened. The new intensities are then mapped back to the initial image and an illumination normalised image arises. In the case of a colour image, histogram equalisation is performed only in the intensity channel of the image.

- **SVD normalisation:** This method is based on the notion of Singular Value Decomposition (SVD) [209]. Given a grayscale image $I$, the SVD operation decomposes the image to three matrices

$$I = U_I \Sigma_I V_I \tag{5.10}$$

  Matrices $U_I$ and $V_I$ are orthogonal square matrices and $\Sigma_I$ is a diagonal matrix that

contains the sorted singular values of $I$ in its main diagonal. Given a set of images $I_1, I_2, ..., I_n$ for comparison, the intensity mean $M$ of all images is computed and it is used to define a new image $I_M$ that has the same size as the compared images and it has all pixel values equal to $M$. The maximum singular value of image $I_M$ and the maximum singular value of the $i$-th image $I_i$ are employed in order to compute the ratio

$$\xi_i = \frac{max(\Sigma_{I_M})}{max(\Sigma_{I_i})} \tag{5.11}$$

Afterwards, a new diagonal matrix ($\hat{\Sigma}_{I_i} = \xi_i \Sigma_{I_i}$) is formed and is used to reconstruct the illumination normalised image $\hat{I}_i$

$$\hat{I}_i = U_{I_i} \hat{\Sigma}_{I_i} V_{I_i} \tag{5.12}$$

In the case of a colour image, SVD is applied only to the intensity channel of the image.

### 5.4.2 CHAM algorithm

In this section, we propose a novel algorithm that allows the amplification of changes between compared images. This can be considered as a vital pre-processing step towards a more accurate and robust change detection system. An outline of the proposed CHAM algorithm is presented in Fig. 5.4.

Given an image $I(x, y)$ captured at time frame $t_1$, the same image captured at a later time frame $t_2$ and translated spatially by a vector $(a, b)$ can be described as $I_2 = I(x+a, y+b)$. Our goal is the synthesis of a new image,

$$I_2^{'} = I(x + (1 + m)a, y + (1 + m)b) \tag{5.13}$$

which has its spatial translation amplified by a factor $m$. To achieve this amplification, we transform the images in the frequency domain using Fourier analysis. Applying Fourier transform to the images leads the initial or reference image at time frame $t_1$ to be equal to $F(u, v)$ and the image at time frame $t_2$ to be equal to $F(u, v)e^{i(au+bv)}$. This means that the motion magnified image at time frame $t_2$ is equal to

$$\mathcal{F}\{I_2^{'}\} = F(u, v)e^{i(1+m)(au+bv)} \tag{5.14}$$

This is in par with the basic Fourier transform property that states that a shift in the time domain corresponds to a phase multiplication in the frequency domain. Thus, motion

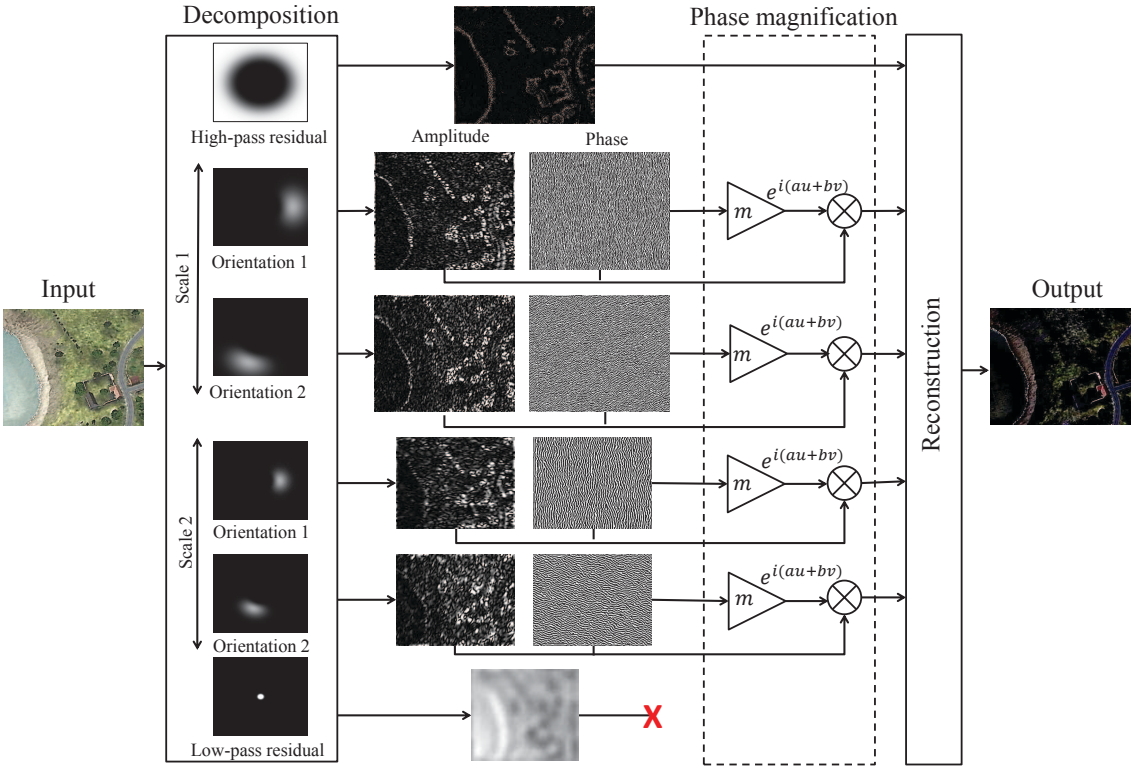



Figure 5.4: Outline of the proposed CHAM algorithm. The depicted images are presented slightly sharpened and contrast enhanced for better visualisation.

magnification in the time domain can be achieved by phase amplification in the frequency domain.

Wadhwa et al. in [206] achieved motion magnification by decomposing each video frame in sub-bands using complex-valued steerable pyramids [210] in several locations, scales and orientations, magnifying the phase differences of specific frames by employing temporal filtering and finally reconstructing the frames by adding the sub-bands. Although our task is the amplification of changes between two images, we show that the notion of motion magnification can be employed in the change detection task as well since changes can be considered as a special type of local motion between two images and they are amplified when the compared images are shifted further away in the time domain. In fact, several change detection techniques rely on the computation of local displacements between image regions to quantify their similarity [128, 211].

However, there are some significant differentiations in magnifying motion between two consecutive frames of a video sequence and amplifying changes between two images taken at completely different temporal periods that should be taken into account when designing a change amplification methodology. Video frames are captured by the same sensor and the small time interval between their capture restricts the amount of illumination variations and noise they are embedded with. On the other hand, two images with a large time interval between their capture are embedded with significant amount of noise and

global and local illumination variations due to different sun angles the time the images are captured and even different sensors that capture the images.

To counter the effects of illumination variations and noise, we propose the CHAM algorithm as a pre-processing step before the employment of any change detection technique. To this end, the image is initially transformed to the YIQ colour space in order for the intensity channel to be separated from the colour information. Afterwards, the image is decomposed in sub-bands using complex-valued steerable pyramids and the phase difference of the colour channel Q with respect to the same channel of a reference image is amplified by a factor $m$. The reference image can be the first image of the sequence of temporally different images that are compared. The amplification of the phase variations of the complex-valued steerable pyramid corresponds to magnification of local motions in spatial sub-bands of an image via the Fourier shift theorem [206].

The phase amplification of a colour channel is based on the following observation. Due to the possibly large time interval between the capture of the two compared images, the illumination variations render the intensity channel inappropriate for the modelling of changes present in the images. On the other hand, a colour channel is more robust to illumination variations and it can accurately model and enhance changes in the form of appearance or disappearance of objects between two images.

The final image is obtained by adding the phase-amplified image sub-bands during the reconstruction phase. Except for the spatial image sub-bands resulted from the pyramid filters, the steerable pyramid computes real-valued high-pass and low-pass coefficients describing residual signal components not captured by the filters [210]. In the CHAM algorithm, the Low-pass Residual (LPR) of the image is not added to the reconstructed image. The LPR contains information about the average brightness of the image. Removing the LPR during the image reconstruction phase results in a sharpened image, where the high frequencies are amplified and the low frequencies are relatively attenuated.

The exclusion of the LPR from the image reconstruction phase is based on the idea of equalising images by removing their low frequency components. The low frequency components contain background information that is not necessary for the detection of differences between images that lie in the high frequency domain. A great property of the CHAM algorithm is its ability to suppress noise. In the case of additive noise $N$, the CHAM algorithm does not magnify noise as the change amplified version $I^{'}$ of a noisy image $I + \sigma N$ in the Fourier domain is

$$\mathcal{F}\{I^{'}\} = \mathcal{F}\{I\}e^{im(au+bv)} + \sigma e^{im(au+bv)}\mathcal{F}\{N\} \tag{5.15}$$

This means that the noise is phase-shifted in the frequency domain and thus translated in the time domain. Therefore, the final result of the CHAM algorithm is a new image that has its phase differences to a reference image amplified and its LPR removed. The final

image appears darker and more detailed than the original image due to the subtraction of the average brightness caused by the removal of the LPR.

The most significant parameter that affects the performance of the CHAM algorithm is the phase amplification factor $m$. As the image features are moved by phase shifting the steerable pyramid filters that cover them, there is a limit beyond which the image features cannot be moved any more due to the limited spatial support of the steerable pyramid filters. In [206], they considered the case of a Dirac function $\delta(t)$ translated over time and proved that in the case of octave-bandwidth pyramid filters the magnification factor $m$ cannot exceed the limit

$$m\delta(t) < \frac{\lambda}{4} \tag{5.16}$$

The term $\lambda$ describes the spatial wavelength of the filters of the steerable pyramid. Eq. (5.16) denotes that the motions of the low spatial frequencies can be magnified more than those of the high spatial frequencies. Exceeding this limit is manifested as artefacts or blur as not all image pyramid components are present in their proper ratios to reconstruct the desired translated feature [206]. Unfortunately, in the change detection task, there is no knowledge about the frequency of the changes. Therefore, careful experimentation is needed to select an optimal change amplification factor $m$.

## 5.5 Evaluation of change detection techniques

In this section, we analyse and compare various illumination normalisation techniques that process images prior to their comparison. Furthermore, we optimise the parameters of the CHAM algorithm and evaluate the effect of the CHAM algorithm on the performance of the tested pixel-based change detection methodologies on the Athens and the AICD datasets. Finally, we evaluate and compare the top-down and bottom-up change detection approaches and the proposed CNN-BCD algorithm with and without the CHAM algorithm on the Athens dataset.

### 5.5.1 Comparison of illumination normalisation methods

Four illumination normalisation algorithms are compared in order to identify the one that achieves the optimal change detection results on the Athens change validation set. The image differencing method is selected as the reference change detection technique under which the illumination normalisation methods are evaluated.

The evaluation of the methods is based on two metrics, the joint entropy of the illumination normalised images and the Euclidean distance of the histograms that contain the change map values of the pixels that belong to the "change" and "no-change" classes with respect

to the ideal distributions of "change" and "no-change" classes. The ideal distributions of the "change" and "no-change" classes have only the rightmost and leftmost bins filled respectively to indicate that for the image differencing method ideally all changes should have values close to unity and all other pixels should have values close to zero.

The results of applying the tested illumination normalisation methods to the Athens change validation set is presented in Table 5.1. Furthermore, the average change values of the pixels that belong to the "change" and "no-change" classes are presented to further support the selection of the optimal illumination normalisation method.

Table 5.1: Evaluation of illumination normalisation methods.

| Method \ Metric | Joint entropy | Euclidean distance | Average "change" | Average "no-change" |
|---|---|---|---|---|
| Normalisation | 11.6257 | 1.7192 | 0.1325 | 0.06 |
| Standardisation | **5.7044** | **1.3126** | 1.3369 | 0.4891 |
| SVD normalisation | 11.6838 | 1.5764 | 0.131 | 0.047 |
| Histogram equalisation | 9.884 | 1.5669 | 0.1291 | 0.0474 |

The joint entropy between two images is calculated by counting occurrences of pairs of intensities between the corresponding pixels of the images and expresses the uncertainty about the intensity values of an image when the other image is known. As a result, this metric should be as low as possible to show that two images are correlated. Moreover, the distribution of the pixel values that belong to the "change" and "no-change" classes of a change map should be as close as possible to the ideal distributions, which means that the metric of Euclidean distance should present small values.

From Table 5.1, it can be observed that both metrics agree that the optimal illumination normalisation technique is the standardisation, which processes images so that they have zero mean and unity variance. This conclusion is also backed up by observing the difference between the average values of the "change" and "no-change" classes. When standardisation is employed, the classes are better separated since the difference of their average values is larger. An illustration of the effect of standardisation on the distribution of the change map values is presented in Fig. 5.5. It can be observed that the use of standardisation leads to a drop in the Euclidean distances of the distributions of the change map values from the ideal distributions, meaning that the "change" class is more easily differentiated from the "no-change" class.

### 5.5.2 CHAM algorithm optimisation

This section discusses the optimisation of the parameters that affect the performance of the CHAM algorithm in the detection of changes on the Athens change validation set. The CVA algorithm is chosen as the reference change detection technique for the CHAM parameter optimisation. The selection of the spectral channel, whose phase differences are

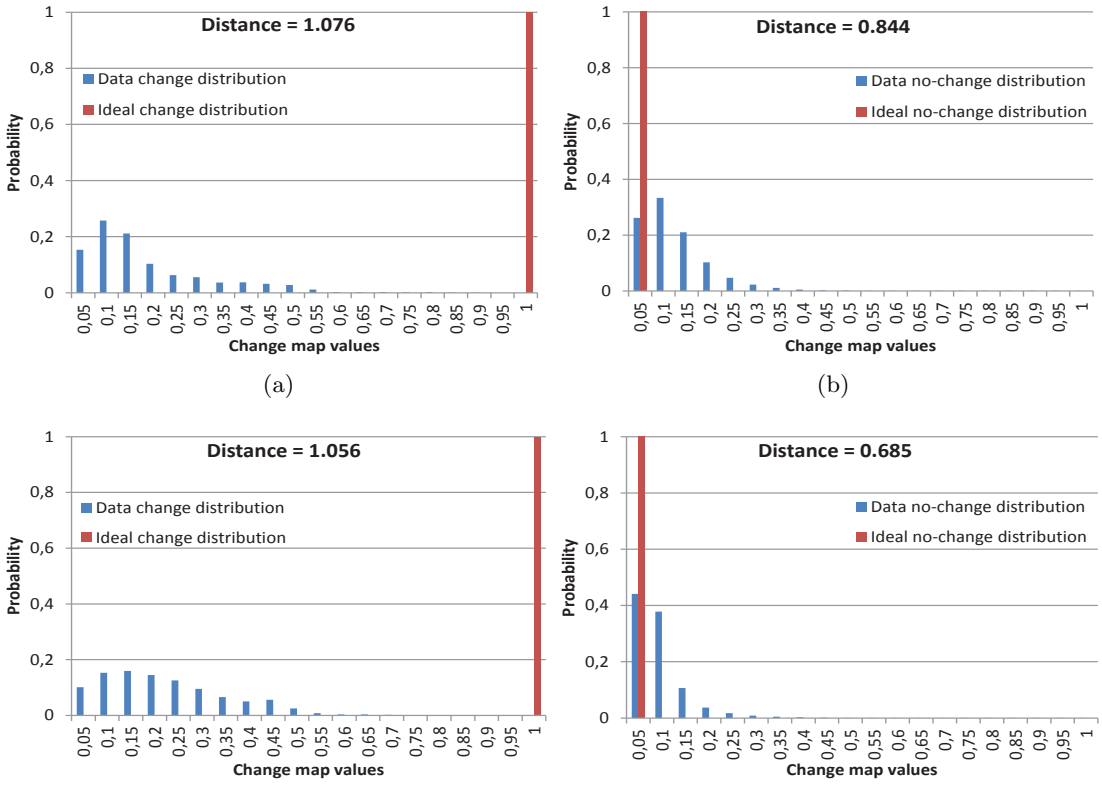



Figure 5.5: Comparison of ideal distributions (red) and distributions of change map values of "change" and "no-change" classes (blue) before (a), (b) and after (c), (d) the standardisation method. The Euclidean distances of the distributions are shown at the top of the figures.

enhanced plays a significant role in the performance of the CHAM algorithm. As it has already been noted, the intensity channel is not a reliable choice when the time interval between the capture of the compared images is large. Table 5.2 supports the previous statement and presents the results of the experimentation with various spectral channel combinations.

Table 5.2: Performance of CHAM algorithm for various phase-amplified spectral channel combinations. The results are presented in the format: mean ± std.

| Spectral channel | Recall | Precision | $F_1$-score | Kappa |
|---|---|---|---|---|
| Y | **0.405±0.108** | 0.036±0.016 | 0.065±0.028 | 0.062±0.027 |
| I | 0.391±0.089 | 0.051±0.021 | 0.09±0.037 | 0.087±0.035 |
| Q | 0.394±0.087 | **0.053±0.022** | **0.093±0.037** | **0.09±0.036** |
| IQ | 0.392±0.082 | 0.052±0.022 | 0.092±0.037 | 0.089±0.036 |
| YIQ | 0.402±0.114 | 0.038±0.017 | 0.069±0.031 | 0.066±0.029 |

Table 5.2 verifies the fact that the spectral channel Q is a more reliable choice for the amplification of changes in the remote sensing images of the Athens change validation set. In fact, the use of any combination of channels that carry colour information (I or Q) gives similar results as far as $F_1$-score and Kappa coefficient are concerned and much better results than the cases, in which the intensity channel Y is employed.

Furthermore, experiments are conducted to select the optimal bandwidth of the steerable pyramid filters. Both octave and sub-octave bandwidths are tested. Sub-octave bandwidths have the ability to allow larger magnification factors $m$ as the impulse responses of the filters are wider and thus their spatial support is larger, in expense for more overcomplete representations [206]. Our experiments reveal that the CHAM algorithm achieves similar performance no matter the bandwidth of the steerable pyramid filters. As a result, the octave bandwidth is chosen for the rest of the experimental procedure.

The next set of experiments concerns the selection of the optimal amplification factor $m$. Fig. 5.6 presents the $F_1$-score and Kappa coefficient for various values of $m$.

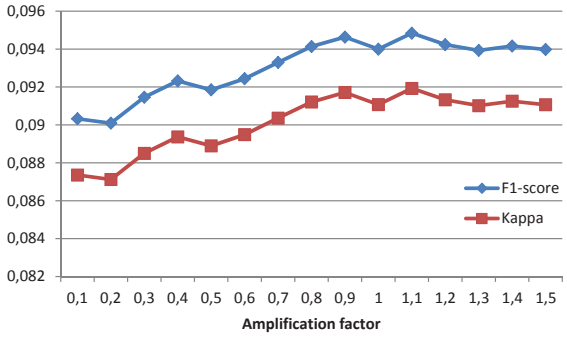



Figure 5.6: Performance of the CHAM algorithm on the Athens change validation set for various values of amplification factor $m$ with respect to the average metrics of $F_1$-score and Kappa.

The experimentation with the amplification factor $m$ reveals that the optimal value of $m$ is equal to 1.1. As the value of $m$ increases, so does the performance of the CHAM algorithm on the validation set. However, when the value of $m$ surpasses the value of 1.1, a drop in the performance of the CHAM algorithm is noticed and this performance degradation is attributed to the limited spatial support of the steerable pyramid filters that does not allow an accurate image reconstruction.

Finally, we examine the effect of the CHAM algorithm with the inclusion or not of the LPR on the performance of the CVA method on the Athens change validation set. The results are summarised in Table 5.3, where it can be concluded that the employment of the CHAM algorithm with the inclusion of the LPR (CHAM + LPR) gives an increase in the Kappa coefficient by about 31.4%, while the removal of the LPR during the image reconstruction phase (CHAM - LPR) gives an additional improvement in the performance of the CVA algorithm by about 7.6%. Unfortunately, the removal of the LPR leads to a drop in the number of detected real changes (i.e. true positives), as it is concluded from the lower recall.

An illustration of the effect that the CHAM algorithm has on the change maps that the CVA method outputs is presented in Fig. 5.7. It can be observed that the inclusion of LPR in the image reconstruction phase of the CHAM algorithm leads to suppression of few false changes. On the other hand, the removal of LPR achieves both an enhancement of

Table 5.3: Performance of CVA method with and without the CHAM algorithm. The results are presented in the format: mean ± std.

| Method | Recall | Precision | $F_1$-score | Kappa |
|---|---|---|---|---|
| CVA without CHAM | **0.404±0.028** | 0.042±0.02 | 0.074±0.031 | 0.07±0.029 |
| CVA with CHAM + LPR | 0.379±0.09 | 0.055±0.024 | 0.095±0.039 | 0.092±0.038 |
| CVA with CHAM - LPR | 0.311±0.081 | **0.061±0.023** | **0.101±0.036** | **0.099±0.034** |

the true changes and an attenuation of false changes, especially in areas that are severely affected by strong local illumination variations, such as shadows.

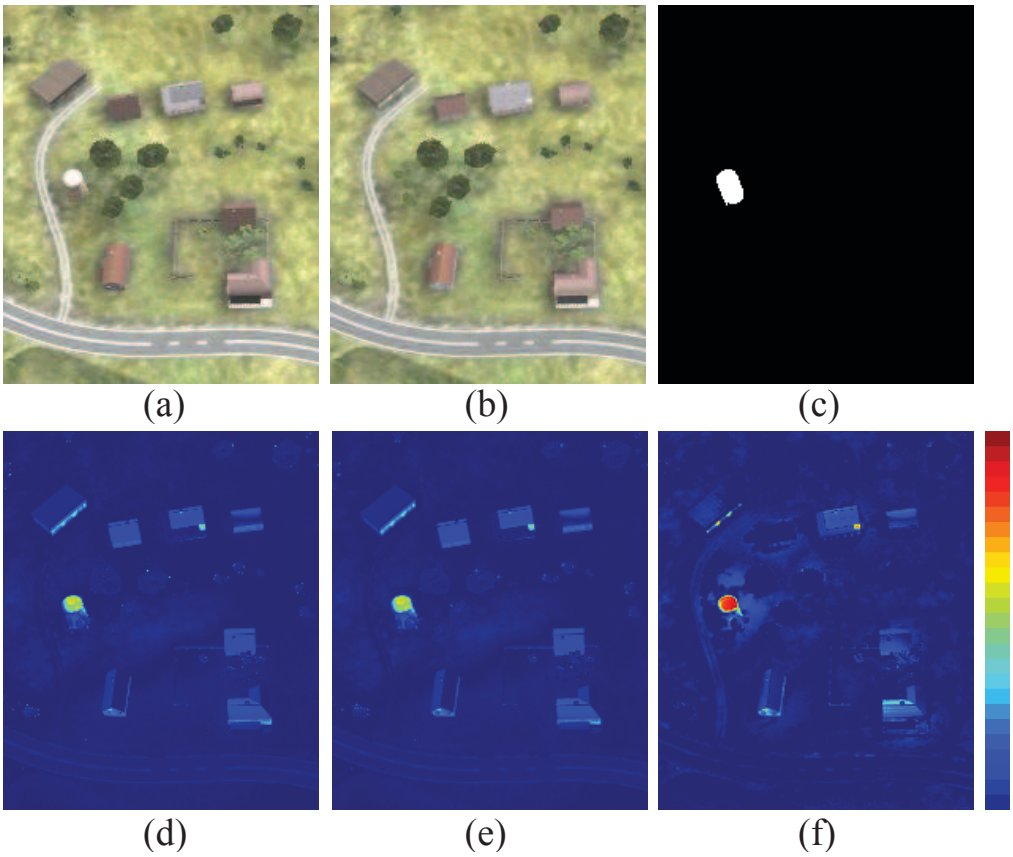



Figure 5.7: Effect of the CHAM algorithm on change maps: (a), (b) Compared images, (c) ground truth differences and change maps from (d) CVA method, (e) CVA with CHAM + LPR and (f) CVA with CHAM - LPR.

### 5.5.3 Optimisation of NCC and CNN-BCD algorithms

In this section, we examine the optimisation of the parameters that affect the performance of the tested NCC and CNN-BCD algorithms. These parameters are the size of the neighbourhood ($n \times n$) in pixels of the NCC algorithm and the parameter $\alpha$ of the ETL layer of the proposed CNN-BCD algorithm.

The results of the experimentation on the neighbourhood size of the NCC algorithm are

presented in Table 5.4, where it can be observed that the optimal performance of the NCC algorithm on the Athens change validation set with respect to the metric of $F_1$-score is achieved, when the size of the neighbourhood $n$ is equal to 8. Such a value gives the highest precision in the change detection task but a drop in recall is noticed with respect to larger neighbourhood sizes.

Table 5.4: Performance of NCC algorithm for various neighbourhood sizes $n$. The results are presented in the format: mean ± std.

| NCC neighbourhood size $n$ | Recall | Precision | $F_1$-score | Kappa |
|---|---|---|---|---|
| 8 | 0.487±0.108 | **0.01±0.005** | **0.019±0.011** | **0.016±0.009** |
| 16 | 0.721±0.101 | 0.007±0.004 | 0.013±0.007 | 0.009±0.005 |
| 32 | **0.765±0.09** | 0.006±0.003 | 0.011±0.006 | 0.008±0.004 |
| 64 | 0.755±0.084 | 0.005±0.003 | 0.011±0.006 | 0.007±0.004 |

The next experiment concerns the selection of the optimal parameter $\alpha$ that affects the behaviour of the ETL layer in the CNN-BCD architecture. The optimisation of this parameter is based on a new training and validation set that was created from the Athens change validation set.

More specifically, image patches of size $21 \times 21$ pixels are cropped from the images of the Athens change validation set. The image patches are labelled based on the ground truth change label of the central pixels of the patches. This means that if a central pixel belongs to the "change" class, then the entire patch is labelled as "change" and vice versa. From this procedure, 32600 positive ("change") and 35200 negative ("no-change") image patches are defined. Afterwards, this dataset is split in a training and a validation set by randomly selecting 70% and 30% of the image patches respectively.

Similar to the training procedure of the CNNs presented in Chapter 4, the training of the CNN-BCD algorithm is performed for a maximum number of iterations or epochs, which we set equal to 100. In each epoch, the training set is introduced to the CNN-BCD algorithm in mini-batches for training and then the CNN-BCD algorithm is evaluated on the validation set. During the evaluation, the validation loss, which is equal to the average negative log-likelihood of each validation sample to belong to the correct class, is computed. The training phase is terminated when the loss on the validation set does not decrease for 5 consecutive epochs. Afterwards, the performance of the CNN-BCD algorithm for the epoch, in which the minimum validation loss was achieved, is preserved. This strategy is employed in order to prevent over-fitting of the CNN-BCD algorithm to the training data, since after a few epochs the loss on the training set keeps decreasing, while the loss on the validation set starts increasing. The training of the CNN-BCD algorithm is performed using the stochastic gradient optimisation method with a batch size of 128 samples, learning rate equal to 0.05 and momentum equal to 0.7.

The optimisation of parameter $\alpha$ is based on the loss on the validation set. The results of the experimentation for various values of $\alpha$ are presented in Fig. 5.8, where they are

compared with the average loss on the validation set when the NL and ETL layers are not employed. The optimal performance is achieved for a value of $\alpha$ equal to 1.8. For this value of $\alpha$, a drop on the average loss on the validation set by almost 44.7% is noticed.

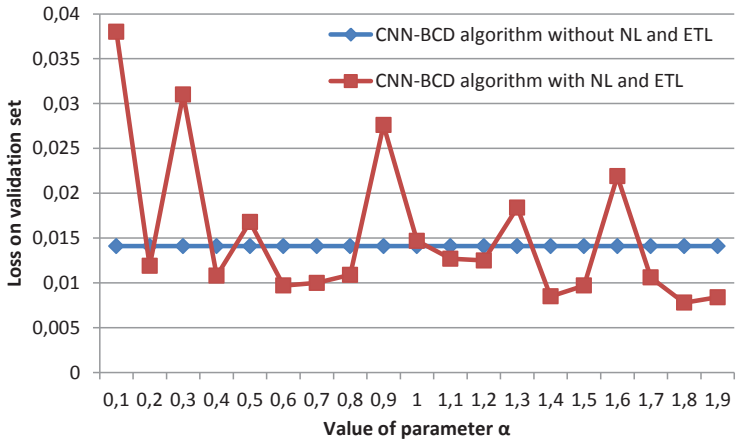



Figure 5.8: Performance of the CNN-BCD algorithm on the validation set for various values of $\alpha$.

As far as the morphological processing of the final change maps is concerned, the same morphological closing followed by a morphological opening operations are performed. The morphological operations employ a disk structuring element of a radius of 5 pixels. Such a structuring element is found to be large enough to discard false changes and small enough to allow true changes to pass through.

### 5.5.4 Effect of CHAM algorithm on change detection performance

The effect of the novel CHAM algorithm on the performances of the pixel-based change detection methods on the Athens and the AICD datasets is evaluated in this section. Furthermore, a comparison of the pixel-based change detection methods on the two datasets is performed. Finally, the performance of the CNN-BCD algorithm on the Athens dataset with and without the CHAM algorithm is presented.

Table 5.5 presents the change detection results of the tested methodologies on the Athens dataset with and without the proposed CHAM algorithm. A comparison of the change detection methods shows that the CVA algorithm achieves the best performance on the Athens dataset, followed by the image differencing technique. This means that employing all available multi-spectral information is beneficial for the change detection task. The poor performance of the NCC method can be attributed to the strong local illumination variations and the spectral and spatial resolution differences between the compared images. Furthermore, it can be concluded that all tested methodologies achieve improved change detection results when the CHAM algorithm pre-processes images. More specifically, the CVA method is improved by more than 24% with respect to the metrics of both $F_1$-score and Kappa coefficient and the other methods present even larger improvements, when the

CHAM algorithm is employed.

Table 5.5: Performance of change detection methods on Athens dataset. The results are presented in the format: mean ± std.

| Method | Without CHAM | | With CHAM | |
|---|---|---|---|---|
| | $F_1$-score | Kappa | $F_1$-score | Kappa |
| Image Differencing | 0.064±0.03 | 0.06±0.028 | 0.076±0.035 | 0.073±0.031 |
| Image regression | 0.016±0.012 | 0.012±0.01 | 0.027±0.015 | 0.023±0.011 |
| Median background subtraction | 0.028±0.009 | 0.024±0.007 | 0.038±0.013 | 0.034±0.01 |
| CVA | **0.073±0.029** | **0.07±0.027** | **0.091±0.032** | **0.087±0.03** |
| NCC | 0.011±0.009 | 0.006±0.006 | 0.014±0.008 | 0.01±0.005 |

Table 5.6 summarises the performance of the tested change detection methods on the AICD dataset with and without the use of the CHAM algorithm. The comparison of the methods reveals the superior performance of the NCC method on the AICD dataset. Operating at the window level, the NCC algorithm is more robust to global illumination variations caused from the different viewpoint angles than the other tested pixel-based change detection approaches. Furthermore, the image pre-processing using the CHAM algorithm leads to improved change detection results for all tested methodologies. More specifically, the performance of the NCC algorithm is improved by at least 33% and 34% with respect to the metrics of $F_1$-score and Kappa coefficient respectively, when the CHAM algorithm is employed. Other change detection methods present an even larger boost in their performance by at least 225% with respect to the metric of $F_1$-score, when the CHAM algorithm pre-processes images.

Table 5.6: Performance of change detection methods on AICD dataset. The results are presented in the format: mean ± std.

| Method | Without CHAM | | With CHAM | |
|---|---|---|---|---|
| | $F_1$-score ($\times 10^{-3}$) | Kappa ($\times 10^{-3}$) | $F_1$-score ($\times 10^{-3}$) | Kappa ($\times 10^{-3}$) |
| Image Differencing | 3.8±1.4 | 2.1±1.1 | 13±4.2 | 11.4±3.8 |
| Image regression | 2.4±0.9 | 0.7±0.4 | 12.2±2.8 | 11.3±2.3 |
| Median background subtraction | 2.8±1 | 1.1±0.6 | 29.2±8.9 | 27.6±7.5 |
| CVA | 5.1±1.5 | 3.4±1.1 | 21.4±4.3 | 19.8±3.9 |
| NCC | **56.9±10.3** | **55.1±9.7** | **75.8±10.6** | **73.6±9.9** |

An illustration of the performance of the tested change detection methodologies on a pair of images of the Athens dataset with and without the CHAM algorithm is presented in Fig. 5.9. The comparison of the computed change maps reveals a significant improvement in the performance of the change detection methods, when the CHAM algorithm is employed.

Fig. 5.10 presents the performance of the NCC algorithm on the AICD dataset with and without the CHAM algorithm and for different viewpoint angles. The results reveal that the proposed CHAM algorithm is robust to differences in viewpoint angles and thus, global illumination variations as the effect of the CHAM algorithm on the change detection results is beneficial (i.e. improvement of at least 28% with respect to the metric of Kappa coefficient) irrespectively of the viewpoint angle.

Finally, we evaluate the effect of the CHAM algorithm on the performance of the proposed

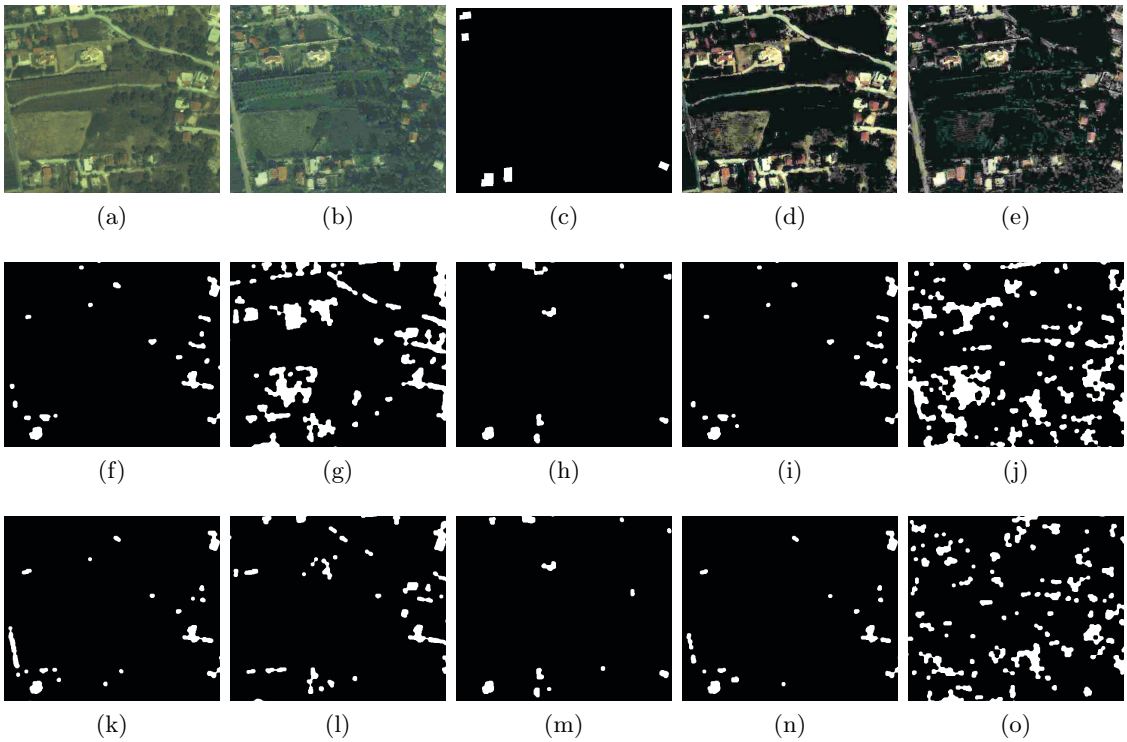



Figure 5.9: Performance of the tested change detection methods on a pair of images of the Athens dataset. The pair of images is shown in (a), (b), while their ground truth differences are shown in (c). The result of the CHAM pre-processing of images (a) and (b) is shown in (d) and (e) respectively. The change maps of image differencing, regression, median background subtraction, CVA and NCC without the CHAM pre-processing are shown in (f)-(j) respectively, while the change maps of the methods with the CHAM pre-processing are presented in (k)-(o) respectively.

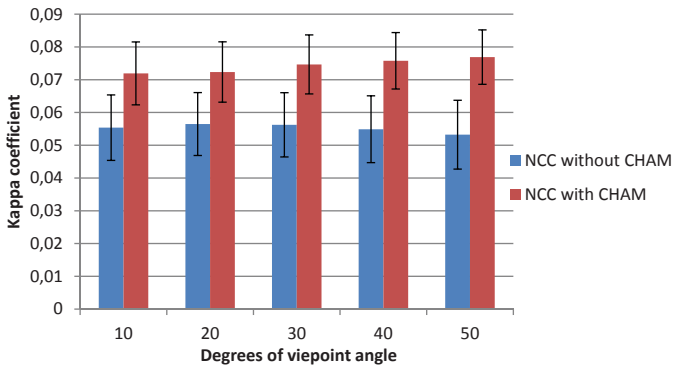



Figure 5.10: Performance of the NCC algorithm on the AICD dataset with and without the CHAM algorithm for various viewpoint angles.

CNN-BCD algorithm on the Athens dataset. To achieve a fair comparison between the CNN-BCD algorithms with and without the CHAM pre-processing, we employ for the case of the CNN-BCD algorithm with the CHAM pre-processing, the same CNN architecture with the same optimised ETL layer ($\alpha = 1.8$), as in the case of the CNN-BCD algorithm without the CHAM pre-processing. However, the images of the Athens change validation set are pre-processed using the CHAM algorithm and new training and validation sets

are formed based on the procedure described in Section 5.5.3, and they are fed to the CNN-BCD algorithm with the CHAM pre-processing for training.

Table 5.7 presents the change detection results on the Athens change test set using the CNN-BCD algorithm with and without the CHAM algorithm and compares these results with the optimal pixel-based change detection methodology (i.e. CVA algorithm) on the Athens change test set, as it was presented earlier in this section. Furthermore, Fig. 5.11 illustrates the results of the CVA and CNN-BCD algorithms with and without the use of the CHAM algorithm on two images of the Athens change test set.

Table 5.7: Performance of CVA and CNN-BCD algorithms on the Athens change test set. The results are presented in the format: mean ± std.

| Method | Recall | Precision | $F_1$-score | Kappa |
|---|---|---|---|---|
| CVA without CHAM | **0.384±0.039** | 0.042±0.019 | 0.073±0.029 | 0.07±0.027 |
| CVA with CHAM | 0.349±0.037 | 0.054±0.023 | 0.091±0.032 | 0.087±0.03 |
| CNN-BCD without CHAM | 0.207±0.127 | 0.106±0.079 | 0.129±0.089 | 0.127±0.088 |
| CNN-BCD with CHAM | 0.242±0.16 | **0.138±0.094** | **0.17±0.115** | **0.168±0.114** |

There are a few conclusions that can be drawn from Table 5.7. Firstly, the CHAM algorithm has a beneficial effect on the performance of the CNN-BCD algorithm as it improves the metrics of $F_1$-score and Kappa coefficient by 31.8% and 32.3% respectively with respect to the performance of the CNN-BCD algorithm without the use of the CHAM pre-processing. Furthermore, the comparison of the CNN-BCD and CVA algorithms shows that the CNN-BCD algorithm achieves a much better performance than the CVA algorithm on the Athens change test set with respect to the metrics of $F_1$-score and Kappa coefficient. Although a drop in recall is noticed, the much higher precision of the CNN-BCD algorithm leads to an overall better performance than the CVA algorithm on the change detection task. Finally, the performances of the CVA and CNN-BCD change detectors are mediocre with respect to the performances one would expect to get from an accurate and robust change detector. The mediocre performances of the tested change detectors can be attributed to the challenging Athens dataset, the images of which suffer from significant illumination variations, noise and spectral differences that renders the problem of identifying true changes hard (compare pair of images in Fig. 5.11(a)-(b)). Furthermore, the threshold that differentiates true changes from false ones plays a crucial role on the performances of the change detectors. We decided to employ Otsu's thresholding that automatically identifies an optimal threshold to differentiate between the "change" and "no-change" classes. However, Otsu's thresholding may be biased towards highly unbalanced classes since the pixels that belong to the "change" class are usually much fewer than the pixels that belong to the "no-change" class. Therefore, a manual threshold, accurately defined for each image of the dataset, may lead to better results.

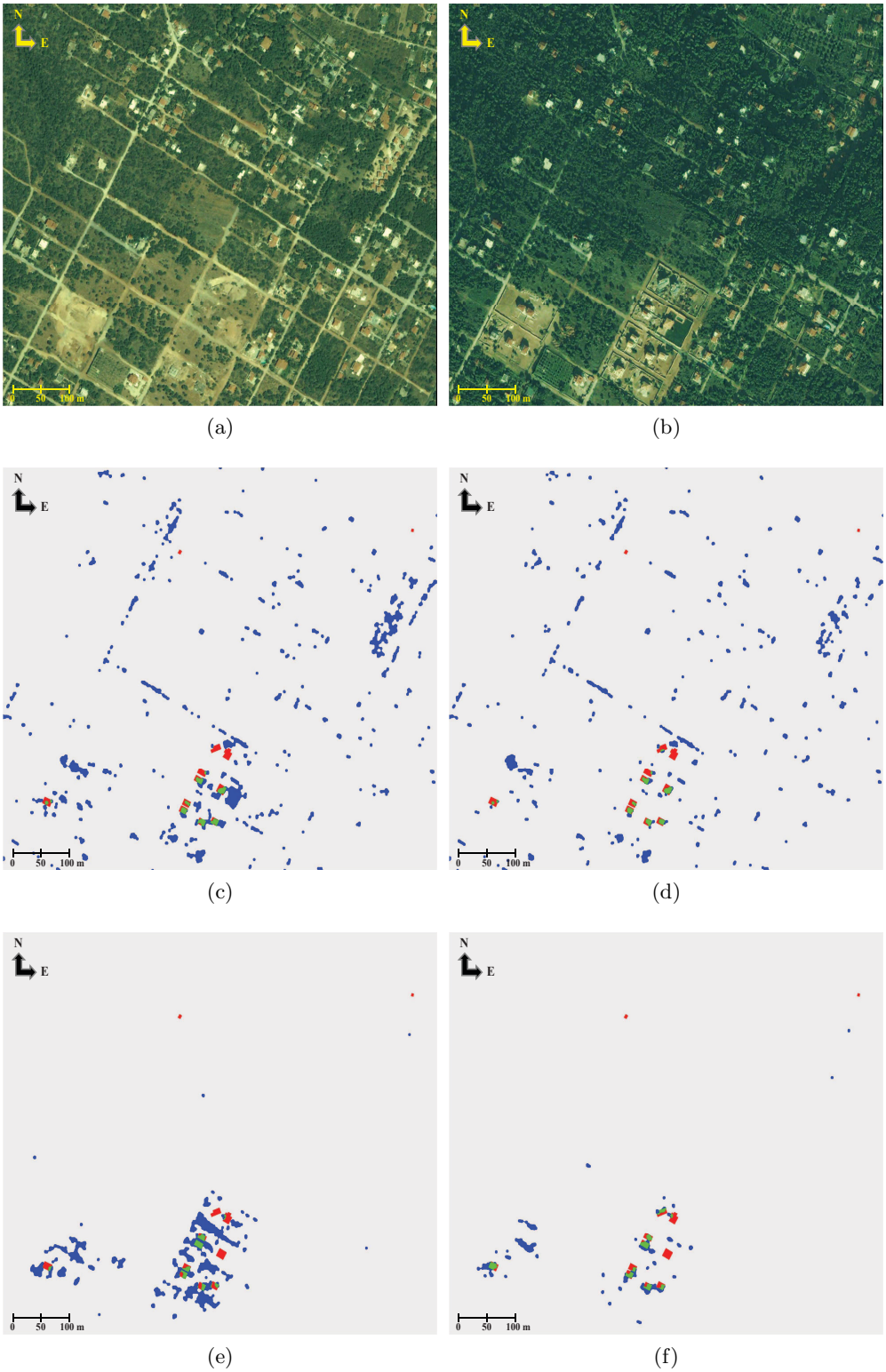



Figure 5.11: Comparison of CVA and CNN-BCD algorithms with and without the CHAM algorithm. The compared images are shown in (a) and (b), while the CVA results without and with the CHAM algorithm are presented in (c) and (d) respectively. The CNN-BCD results without and with the CHAM algorithm are depicted in (e) and (f) respectively. True, false and missed changes are shown in green, red and blue respectively.

### 5.5.5 Comparison of building change detection approaches

The purpose of this section is the comparison of the top-down and bottom-up approaches in the building change detection task. The top-down approach concerns the extraction of building masks (BMs) using the optimal pixel-based building detection methodology of Chapter 4 and the subtraction of BMs in order to identify building changes. As far as the bottom-up approach is concerned, we consider two cases. The first case includes only the changes achieved by the optimal pixel-based change detector on the Athens dataset (i.e. CVA algorithm). The detected changes in this case do not correspond only to buildings, but they are considered general changes. In the second case, the changes of the optimal pixel-based change detector are modelled and only those that correspond to buildings are preserved. To achieve this, the BMs are employed in order to select only the changes that lie inside the BMs and discard all other changes as irrelevant for the task of building change detection. The use of these two cases is to analyse the effect of the change modelling on the performance of the building change detectors.

Furthermore, we evaluate the performance of the CNN-BCD algorithm with and without the employment of the BMs. The use of the BMs in the CNN-BCD algorithm can be considered redundant as the CNN-BCD algorithm is able to detect and preserve only the changes that correspond to building infrastructure, while discarding irrelevant changes. Moreover, for both the CVA and the CNN-BCD algorithms we consider that the CHAM pre-processing has already been applied because it leads to improved change detection results, as shown in the previous section.

Table 5.8 presents the performance of the proposed building change detection methodologies on the Athens change test set, while Fig. 5.12 illustrates the results of applying the tested change detection methodologies on a pair of images of the Athens change test set. Several conclusions can be drawn from the comparison of the building change detection approaches. First of all, the accuracy of the BMs can significantly affect the performance of the top-down approach, since the change detection results, in this approach, are based on the BM subtraction. The top-down approach is able to recognise on average only 38.1% of the true changes, while also achieving a low precision. The mediocre recall is attributed to buildings lost during the extractions of the BMs. Therefore, given that the BMs are employed for the modelling of changes in the bottom-up approach, several true changes may be lost as they are not captured by the BMs. This is the reason for the drop in the metric of recall that the CVA and CNN-BCD algorithms present, when the BMs are employed. To alleviate such problems, it is imperative that the BMs are as accurate as possible.

The effect of the BMs on the CVA algorithm is beneficial as the BMs improve the precision and overall performance of the CVA algorithm on the building change detection task by almost 111% with respect to the metric of $F_1$-score. On the other hand, the performance of the CNN-BCD algorithm deteriorates, when the building masks are employed. Although

Table 5.8: Performance of building change detectors on the Athens change test set. The results are presented in the format: mean ± std.

| Method | Recall | Precision | $F_1$-score | Kappa |
|---|---|---|---|---|
| CVA | 0.349±0.037 | 0.054±0.023 | 0.091±0.032 | 0.087±0.03 |
| CVA with BMs | 0.179±0.093 | 0.246±0.037 | **0.192±0.05** | **0.19±0.05** |
| CNN-BCD | 0.242±0.16 | 0.138±0.094 | 0.17±0.115 | 0.168±0.114 |
| CNN-BCD with BMs | 0.1±0.061 | **0.372±0.122** | 0.151±0.088 | 0.15±0.087 |
| BM subtraction | **0.381±0.137** | 0.027±0.007 | 0.05±0.013 | 0.046±0.013 |

the precision of the CNN-BCD algorithm is significantly increased, when the BMs are employed, the metric of recall drops leading to an overall worse performance on the Athens change test set. This performance degradation can also be attributed to the fact that the CNN-BCD algorithm is capable of recognising building changes without the need of BMs, rendering the BMs not only unnecessary, but also harmful to the CNN-BCD building change detection performance as they introduce their own inaccuracies.

Finally, the optimal building change detection performance is achieved by the the CVA algorithm with the use of the BMs, followed by the CNN-BCD algorithm without the use of the BMs. A comparison between the top-down and bottom-up approaches reveals that the bottom-up approaches outperform the top-down approaches as they have the advantage of quantifying the magnitude of change, thus discarding several false changes. However, both approaches can be significantly improved if more accurate building masks are provided. This conclusion enhances the need for accurate detection of the objects of interest in a change detection framework.

## 5.6 Summary

This chapter addresses the task of change detection from two main approaches. In the top-down approach, the building masks that are provided using the optimal pixel-based building detector of Chapter 4 are subtracted to produce the building changes, while in the bottom-up approach, change detection algorithms are employed to compute differences, before these differences are mapped to building changes using the provided building masks.

Furthermore, this chapter analyses various illumination normalisation techniques that pre-process images prior to their comparison and they have as a purpose to improve the change detection results. To this end, we propose the CHAM algorithm that enhances the differences between the compared images. To achieve this, the CHAM algorithm amplifies the phase differences of the compared images and removes their low-pass residual in order to equalise their average intensity and highlight high frequency signals that correspond to changes. Our experiments on the challenging Athens and AICD datasets, whose images present strong local and global illumination variations, noise, low contrast and spectral and spatial resolution differences, show the robustness of the CHAM algorithm as it manages

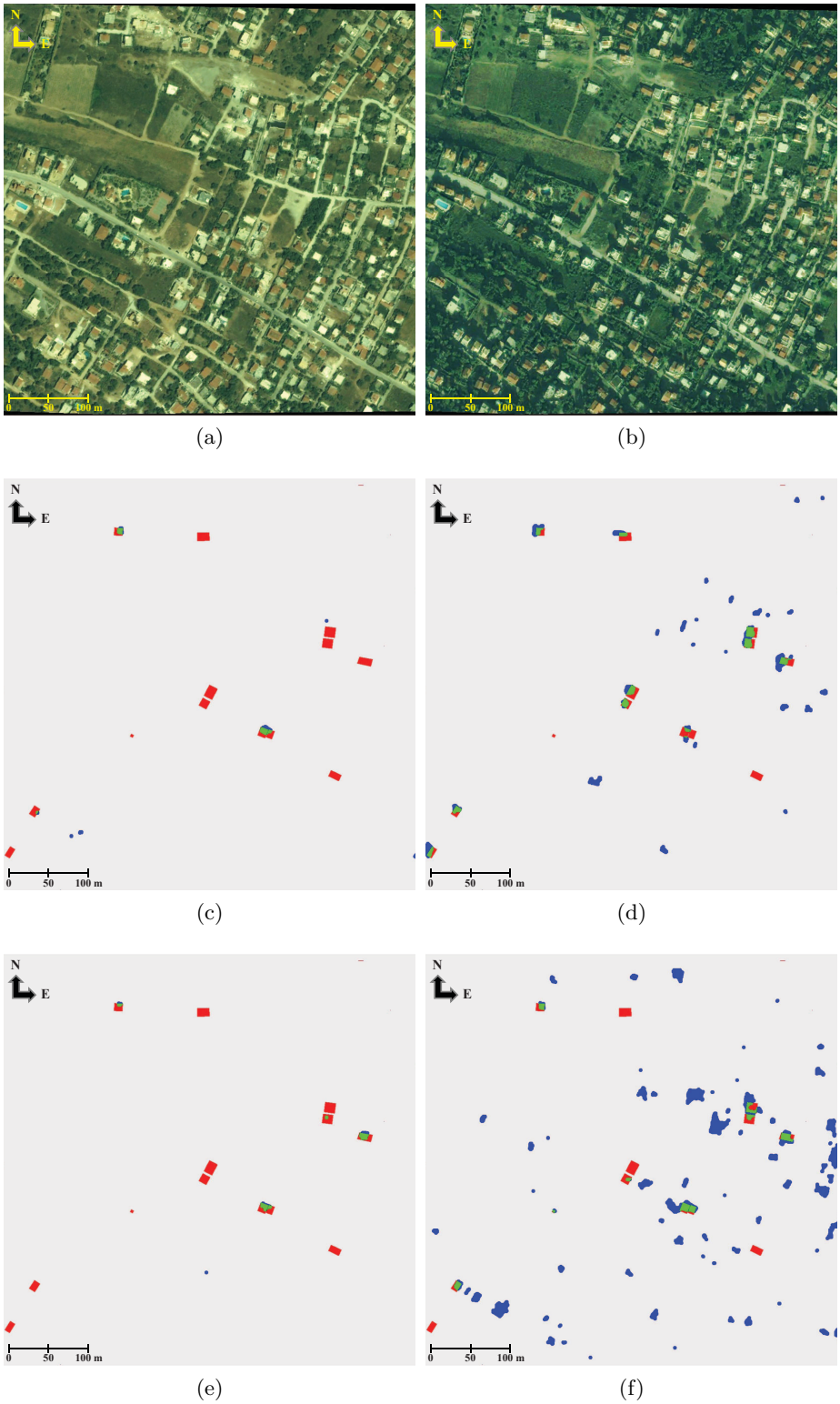



Figure 5.12: Performance of change detection approaches on a pair of images of the Athens change test set. The pair of images is depicted in (a), (b), while the building change maps computed from (c) CVA with BMs, (d) CNN-BCD, (e) CNN-BCD with BMs and (f) BM subtraction approaches are shown as well.

to improve the change detection performance of all tested change detection techniques.

Moreover, we propose a robust and accurate building change detector by employing CNNs. Our experimentation on the Athens dataset reveals the superiority of the proposed CNN-BCD algorithm over the other tested pixel-based change detection methodologies. Finally, we compare the top-down and bottom-up building change detection approaches and we recognise the need for accurate detection of buildings that can assist in the extraction of robust building masks and the correct modelling of changes, thus improving both top-down and bottom-up building change detection approaches simultaneously.

The next chapter summarises the work done in Chapters 3, 4 and 5 and proposes directions for further research in order to address the problems current methodologies on several applications, including urban change monitoring, face.

# CHAPTER 6

## Conclusions

Monitoring changes in urban infrastructure can be a really useful task for several applications that require the creation and constant update of urban maps. However, detecting and modelling urban changes can simultaneously be a challenging problem due to the spectral and spatial limitations of the sensors that capture the remote sensing images and the fact that buildings appear in various sizes and shapes and present significant colour and texture variations. Several methodologies have been developed over the years in an attempt to overcome the challenges of classifying a complex and diverse urban environment and provide accurate and robust building change detection results.

In this thesis, the problem of monitoring urban changes is decomposed to two fundamental tasks; building and change detection. Each task is addressed separately and ways to overcome the challenges of each task are proposed. The development of accurate and robust building and change detection methodologies leads to the successful addressing of the problem of urban change monitoring.

The problem of building detection is approached from two main directions. Chapter 3 examines several building detectors based on the discriminative HOG features and the classification power of the SVM classifiers. The analysis starts from a simple HOG building detector and progresses to combinations of HOG, NDVI, FAST and LBP features in an attempt to improve the performance of an object-based building detector. Furthermore, a novel cosine-based distance function is proposed that can improve the robustness of an object detection methodology to the presence of noise and outliers. The new cosine-based distance function replaces the $l_2$-norm during the SVM training and testing and we demonstrate that it improves the classification performance of a building detector. Moreover, the transformation of the SVM scores to probabilities by employing Platt scaling is proposed. Such transformation allows the definition of a new threshold that differentiates positive from negative detections. Experiments reveal that the new threshold leads to a superior performance in the building detection task.

The novelties that were previously presented allow the development of an accurate and robust object-based building detection methodology that achieves impeccable results on the challenging Athens dataset and its performance is comparable and even outperforms state-of-the-art building detection methods presented in Table 2.2. However, the object-based methodology perceives buildings as image patches and no information of their exact boundaries is provided. To overcome this problem and allow an accurate delineation of building boundaries, Chapter 3 proposes a set of novel region refinement processes. These processes include an unsupervised image segmentation technique that automatically defines the optimal number of classes, a novel region scoring algorithm that weigh image regions based on the properties of an ideal object detector in order to select the regions that are the most probable to correspond to buildings and a proposed building formation algorithm that merges regions based on the fact that the image segmentation procedure may over-segment building regions. The object-based building detection methodology with the employment of the region refinement processes leads to a pixel-based building detector capable of identifying building boundaries and extracting building masks that contain information about the size, shape and location of buildings in images.

Chapter 4 takes another approach to the building detection task and, contrary to Chapter 3 that searches manually for discriminatory features to employ, it uses Convolutional Neural Networks (CNNs). CNNs have the ability to automatically generate complex features without human intervention that are optimal for the task that needs to be solved. Urged from the need of a general CNN architecture that can work in any classification problem, the shallow *Modular*-CNN architecture is proposed that can be optimised based on the problem at hand. Furthermore, two novel layers are proposed to improve the accuracy of the *Modular*-CNN. The first layer is called Normalisation Layer (NL) and it combines the power of both raw and normalised features in order to improve the generalisation ability of the *Modular*-CNN. The second layer is the Euler Transform Layer (ETL) and it takes advantage of the robust cosine-based distance function that was proposed in Chapter 3. By transforming the feature vectors into their Euler representation, we decrease the sensitivity of the *Modular*-CNN to the presence of noise and outliers. We also test these layers as general CNN layers on other classification tasks but the results with respect to their effect on the performance of a CNN are not yet conclusive. Finally, the optimal object-based and pixel-based building detectors of this thesis are proposed by considering the positive detections of the optimal HOG-LBP building detector and applying the classification abilities of the optimised to the building detection task *Modular*-CNN on these detections.

Chapter 5 deals with the problem of building change detection. To this end, two main approaches are adopted. The first approach, called top-down approach, starts by detecting the objects of interest (i.e. buildings), whose changes should be monitored. Afterwards, change detection can simply be solved by comparing the detected buildings between pairs of images. The second approach is called bottom-up approach and it is based on the detec-

tion of general changes by employing any change detection technique. Afterwards, these changes should be modelled in order to preserve only the changes that correspond to buildings and discard irrelevant ones. The building masks that are extracted from the optimal building detector can be employed both in the top-down approach as their subtraction defines building changes and in the bottom-up approach to discard irrelevant changes. Unfortunately, noise and illumination variations can severely affect the performance of change detection methods. To suppress the effect of noise and illumination variations, we propose a novel Change Amplification (CHAM) algorithm that takes advantage of the time shifting Fourier property to amplify changes between images by magnifying phase differences. The results demonstrate a significant improvement on the change detection performance when the CHAM algorithm is employed. Finally, an accurate building change detector based on CNNs is proposed. The CNN-BCD algorithm is proved to be robust to noise, is capable of detecting only changes related to buildings and it achieves a very good performance on the building change detection problem.

## 6.1 Future prospects

This thesis investigates and proposes a universal strategy to deal with the problem of monitoring urban changes. Several of the contributions of this work lay the ground for a broader range of applications. Furthermore, the way we address the challenges of this work can direct researchers towards better approaches for successfully addressing these or similar challenges. A list of potential directions of future research is discussed below.

**Accurate unsupervised image segmentation.** The problem of image segmentation is well-known and several techniques have been proposed to overcome it. A disadvantage of such techniques is their parametrisation that needs to be meticulously optimised based on the problem at hand. The optimisation of these parameters leads to obstacles in their use in an unsupervised manner as sub-optimal parametrisation can lead to over- and under-segmentation problems. Graphical models, such as Markov and Conditional Random Fields attempt to overcome the parametrisation problem by relying on low-level features. However, noise can affect the extraction of low-level features and thus deteriorate the performance of such models. In our work, we propose an unsupervised methodology to automatically segment images, however, inaccuracies in the segmentation results exist and they can significantly affect building detection results. Thus, the introduction of accurate, robust and unsupervised image segmentation techniques for the problem of object detection and extraction is necessary.

**General CNN architectures.** The heavy use of CNNs for several applications dictates the need for the development of a general CNN architecture that can be optimised based on the problem that needs to be solved. In this thesis, an attempt is made to define a shallow CNN architecture with only a few layers that can be optimised and employed for

datasets that contain small-sized images. Likewise, research can be directed towards the development of other shallow or deeper general CNN architectures that would be able to handle a variety of classification problems.

**Handling of noise in datasets.** Improving the robustness of a classifier to the effect of noise should be a vital part of a research. In this thesis, we propose the use of the cosine-based distance function as an alternative to the $l_2$-norm. The cosine-based distance function can better handle the effect of noise and outliers and improve the accuracy of a building detector. Future work can be directed towards this direction either by improving the cosine-based distance function or by proposing new functions that are more robust to the effect of noise.

**Change amplification.** Enhancing differences between images can significantly assist towards the development of an accurate change detection methodology as noise and illumination variations can severely affect change detection results. To this end, we propose the CHAM algorithm that takes advantage of the time shifting property of Fourier transform to magnify phase differences between images in order to amplify changes. Such an algorithm can be employed as a pre-processing step in several applications that require the detection of differences between images. As a result, future work can be concentrated on improving the performance of the CHAM algorithm to be able to provide more accurate and robust to noise magnification of changes by employing, for instance, non-uniform phase amplification.

**CNN for change detection.** CNNs are able to provide state-of-the-art performances in several applications mainly due to their ability to automatically generate complex descriptive features. However, CNNs have rarely been employed for the change detection task. In this thesis, we propose the CNN-BCD algorithm that has the ability to not only detect changes but also identify the type of changes that needs to be detected. The results demonstrate the superior performance of the proposed CNN-BCD algorithm over other change detection methodologies. Therefore, CNNs are in need of further research and experimentation in order to achieve even higher performance in the task of change detection.

# Bibliography


[1] Satellite Imaging Corporation, "Satellite Image Gallery," http://www.satimagingcorp.com/gallery/.

[2] M. Ilsever and C. Unsalan, "Building detection using HOG descriptors," in *6th International Conference on Recent Advances in Space Technologies (RAST)*, June 2013, pp. 115–119.

[3] B. Sirmacek and C. Unsalan, "A Probabilistic Framework to Detect Buildings in Aerial and Satellite Images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 1, pp. 211–221, Jan 2011.

[4] A. Manno-Kovács and T. Szirányi, "Orientation based building outline extraction in aerial images," in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume I-3, 2012 XXII ISPRS Congress, Melbourne, Australia*, 2012, pp. 141–146.

[5] D. Woo, Q. Nguyen, Q. Nguyen Tran, D. Park, and Y. Jung, "Building detection and reconstruction from aerial images," in *ISPRS Congress, Beijing*, 2008.

[6] T. Hermosilla, L. Ruiz, J. Recio, and J. Estornell, "Evaluation of automatic building detection approaches combining high resolution images and LiDAR data," *Remote Sensing*, vol. 3, no. 6, pp. 1188–1210, 2011.

[7] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, June 2005, pp. 886–893.

[8] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan, "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 1491–1498.

[9] K. Shimada, R. Muto, and T. Endo, "A Combined Method Based on SVM and Online Learning with HOG for Hand Shape Recognition," *JACIII*, vol. 16, pp. 687–695, 2012.

[10] D. F. Llorca, R. Arroyo, and M. A. Sotelo, "Vehicle logo recognition in traffic images using HOG features and SVM," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, Oct 2013, pp. 2229–2234.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25*, 2012, pp. 1106–1114.

[12] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," in *Proceedings of the 23rd International Conference on Machine Learning (ICML-13)*, 2013, pp. 1319–1327.

[13] M. Lin, Q. Chen, and S. Yan, "Network In Network," in *International Conference on Learning Representations*, 2013.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[15] J. Campbell, *Introduction to remote sensing*. CRC Press, 2002.

[16] NASA, "The LandSat program," http://landsat.gsfc.nasa.gov/.

[17] Apollo Mapping, "SPOT 1-7," https://apollomapping.com/imagery/medium-resolution-satellite-imagery/spot.

[18] A. Brunn and U. Weidner, "Extracting buildings from digital surface models," in *IAPRS*, 1997, pp. 3–4.

[19] C. Wang and P. Hsu, "Building detection and structure line extraction from airborne LiDAR data," *Journal of Photogrammetry and Remote Sensing*, vol. 12, no. 4, pp. 365–379, Dec 2007.

[20] E. Baltsavias, S. Mason, and D. Stallmann, "Use of DTMs/DSMs and orthoimages to support building extraction," in *Automatic Extraction of Man-made Objects from Aerial and Space Images*. Verlag, April 1995, pp. 199–210.

[21] J. Hu, S. You, U. Neumann, and K. Park, "Building modeling from LiDAR and aerial imagery," in *Proceedings of ASPRS*, 2004.

[22] K. Karantzalos and N. Paragios, "Large-scale building reconstruction through information fusion and 3-d priors," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 5, pp. 2283–2296, May 2010.

[23] K. Hammoudi and F. Dornaika, "A featureless approach to 3d polyhedral building modeling from aerial images," *Sensors*, vol. 11, no. 1, pp. 228–259, 2011.

[24] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.

[25] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," in *Proceedings of 5th International Conference on Computer Vision*, June 1995, pp. 694–699.

[26] L. Theng, "Automatic building extraction from satellite imagery," *Engineering Letters*, vol. 13, no. 3, Nov 2006.

[27] K. Karantzalos and D. Argialas, "A region-based level set segmentation for automatic detection of man-made objects from aerial and satellite images," *Photogrammetric Engineering and Remote Sensing*, vol. 75, no. 6, pp. 667–677, 2009.

[28] V. Dey, Y. Zhang, and M. Zhong, "Building detection from pan-sharpened GeoEye-1 satellite imagery using context based multi-level image segmentation," in *International Symposium on Image and Data Fusion (ISIDF)*, Aug 2011, pp. 1–4.

[29] Q. Li, N. Mitianoudis, and T. Stathaki, "Spatial kernel K-harmonic means clustering for multi-spectral image segmentation," *Image Processing, IET*, vol. 1, no. 2, pp. 156–167, June 2007.

[30] S. Kluckner and H. Bischof, "Image-based building classification and 3d modeling with super-pixels," in *Proceedings of International Society for Photogrammetry and Remote Sensing, Photogrammetric Computer Vision and Image Analysis*, 2010.

[31] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, Sept 2004.

[32] B. Sirmacek and C. Unsalan, "Urban-area and building detection using SIFT keypoints and graph theory," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 4, pp. 1156–1167, April 2009.

[33] A. Manno-Kovács and A. O. Ok, "Building Detection From Monocular VHR Images by Integrated Urban Area Knowledge," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 10, pp. 2140–2144, Oct 2015.

[34] A. O. Ok, C. Senaras, and B. Yuksel, "Automated Detection of Arbitrarily Shaped Buildings in Complex Environments From Monocular VHR Optical Satellite Imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 3, pp. 1701–1717, March 2013.

[35] D. Singaraju and R. Vidal, "Using global bag of features models in random fields for joint categorization and segmentation of objects," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011, pp. 2313–2319.

[36] C. Zheng and L. Wang, "Semantic Segmentation of Remote Sensing Imagery Using Object-Based Markov Random Field Model With Regional Penalties," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 5, pp. 1924–1935, May 2015.

[37] C. Zheng, Y. Zhang, and L. Wang, "Semantic Segmentation of Remote Sensing Imagery Using an Object-Based Markov Random Field Model With Auxiliary Label Fields," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 5, pp. 3015–3028, May 2017.

[38] M. Golipour, H. Ghassemian, and F. Mirzapour, "Integrating Hierarchical Segmentation Maps With MRF Prior for Classification of Hyperspectral Images in a Bayesian Framework," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 2, pp. 805–816, Feb 2016.

[39] X. Li and H. Sahbi, "Superpixel-based object class segmentation using conditional random fields," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2011, pp. 1101–1104.

[40] W. Xia, C. Domokos, L. F. Cheong, and S. Yan, "Background Context Augmented Hypothesis Graph for Object Segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 4, pp. 582–594, April 2015.

[41] L. Zhou, Y. Qiao, J. Yang, and X. He, "Learning geodesic CRF model for image segmentation," in *19th IEEE International Conference on Image Processing (ICIP)*, Sept 2012, pp. 1565–1568.

[42] Y. Zhang and T. Chen, "Efficient inference for fully-connected CRFs with stationarity," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012, pp. 582–589.

[43] N. Payet and S. Todorovic, "Hough Forest Random Field for Object Recognition and Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 35, no. 5, pp. 1066–1079, May 2013.

[44] A. Kae, K. Sohn, H. Lee, and E. Learned-Miller, "Augmenting CRFs with Boltzmann Machine Shape Priors for Image Labeling," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013, pp. 2019–2026.

[45] J. Yang, S. Sáfár, and M. H. Yang, "Max-Margin Boltzmann Machines for Object Segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014, pp. 320–327.

[46] F. Chen, H. Yu, R. Hu, and X. Zeng, "Deep Learning Shape Priors for Object Segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013, pp. 1870–1877.

[47] R. Salakhutdinov and G. E. Hinton, "Deep boltzmann machines," in *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS'09)*, vol. 5, April 2009, pp. 448–455.

[48] A. Shackelford and C. Davis, "A combined fuzzy pixel-based and object-based approach for classification of high-resolution multispectral data over urban areas," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 10, pp. 2354–2363, Oct 2003.

[49] Z. Sheng-hua, H. Jian-jun, and X. Wei-xin, "A new method of building detection from a single aerial photograph," in *9th International Conference on Signal Processing (ICSP)*, Oct 2008, pp. 1219–1222.

[50] D. Chai, W. Förstner, and M. Ying Yang, "Combine markov random fields and marked point processes to extract building from remotely sensed images," in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2012, pp. 1219–1222.

[51] T. T. Ngo, C. Collet, and V. Mazet, "Automatic rectangular building detection from VHR aerial imagery using shadow and image segmentation," in *IEEE International Conference on Image Processing (ICIP)*, Sept 2015, pp. 1483–1487.

[52] T. T. Ngo, V. Mazet, C. Collet, and P. de Fraipont, "Shape-Based Building Detection in Visible Band Images Using Shadow Information," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 3, pp. 920–932, March 2017.

[53] X. Huang and L. Zhang, "A Multidirectional and Multiscale Morphological Index for Automatic Building Extraction from Multispectral GeoEye-1 Imagery," *Photogrammetric engineering and remote sensing*, vol. 77, no. 7, pp. 721–732, 2011.

[54] Q. Zhang, X. Huang, and G. Zhang, "A Morphological Building Detection Framework for High-Resolution Optical Imagery Over Urban Areas," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 9, pp. 1388–1392, Sept 2016.

[55] K. Stankov and D. C. He, "Building Detection in Very High Spatial Resolution Multispectral Images Using the Hit-or-Miss Transform," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 1, pp. 86–90, Jan 2013.

[56] ——, "Detection of Buildings in Multispectral Very High Spatial Resolution Images Using the Percentage Occupancy Hit-or-Miss Transform," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 10, pp. 4069–4080, Oct 2014.

[57] D. Singh, R. Maurya, A. Shukla, M. Sharma, and P. Gupta, "Building extraction from very high resolution multispectral images using NDVI based segmentation and morphological

operators," in *Students Conference on Engineering and Systems (SCES)*, March 2012, pp. 1–5.

[58] D. Chaudhuri, N. K. Kushwaha, A. Samal, and R. C. Agarwal, "Automatic Building Detection From High-Resolution Satellite Images Based on Morphology and Internal Gray Variance," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 5, pp. 1767–1779, May 2016.

[59] O. Benarchid, N. Raissouni, S. Adib, A. Abbous, A. Azyat, N. Achhab, M. Lahraoua, and A. Chahboun, "Building extraction using object-based classification and shadow information in very high resolution multispectral images, a case study: Tetuan, Morocco," *Canadian Journal on Image Processing and Computer Vision*, vol. 4, no. 1, Jan 2013.

[60] C. Senaras and F. T. Yarman Vural, "A Self-Supervised Decision Fusion Framework for Building Detection," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 5, pp. 1780–1791, May 2016.

[61] P. Hough, "Method and means for recognizing complex patterns," U.S. Patent 3,069,654, Dec 1962.

[62] T. Kim and J. Muller, "Development of a graph-based approach for building detection," *Image and Vision Computing*, vol. 17, no. 1, pp. 3–14, 1999.

[63] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, Nov 1986.

[64] D. Haverkamp, "Automatic building extraction from IKONOS imagery," in *Proceedings of ASPRS*, 2004.

[65] S. Vinson, L. Cohen, and F. Perlant, "Extraction of rectangular buildings in aerial images," in *Proceedings of Scandinavian Conference on Image Analysis (SCIA)*, 2001.

[66] V. Verma, R. Kumar, and S. Hsu, "3d building detection and modeling from aerial LIDAR data," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2006, pp. 2213–2220.

[67] C. Benedek, X. Descombes, and J. Zerubia, "Building Detection in a Single Remotely Sensed Image with a Point Process of Rectangles," in *20th International Conference on Pattern Recognition*, Aug 2010, pp. 1417–1420.

[68] ——, "Building Development Monitoring in Multitemporal Remotely Sensed Image Pairs with Stochastic Birth-Death Dynamics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 33–50, Jan 2012.

[69] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2001, pp. 511–518.

[70] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on Kullback discrimination of distributions," in *International Conference on Pattern Recognition*, vol. 1, Oct 1994, pp. 582–585.

[71] ——, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, Jan 1996.

[72] P. Navarrete and J. Ruiz del Solar, "Analysis and Comparison of Eigenspace-based Face Recognition Approaches," *Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, pp. 817–830, 2002.

[73] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, July 1997.

[74] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object Detection with Discriminatively Trained Part-Based Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 32, no. 9, pp. 1627–1645, Sept 2010.

[75] I. Endres, K. J. Shih, J. Jiaa, and D. Hoiem, "Learning Collections of Part Models for Object Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013, pp. 939–946.

[76] X. Wang, M. Yang, S. Zhu, and Y. Lin, "Regionlets for Generic Object Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 37, no. 10, pp. 2071–2084, Oct 2015.

[77] T. T. Pham, S. H. Rezatofighi, I. Reid, and T. J. Chin, "Efficient Point Process Inference for Large-Scale Object Detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 2837–2845.

[78] S. Mathe, A. Pirinen, and C. Sminchisescu, "Reinforcement Learning for Visual Object Detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 2894–2902.

[79] T. Liu, M. Seyedhosseini, and T. Tasdizen, "Image Segmentation Using Hierarchical Merge Tree," *IEEE Transactions on Image Processing*, vol. 25, no. 10, pp. 4596–4607, Oct 2016.

[80] E. Ohn-Bar and M. M. Trivedi, "To boost or not to boost? On the limits of boosted trees for object detection," in *23rd International Conference on Pattern Recognition (ICPR)*, Dec 2016, pp. 3350–3355.

[81] G. Hinton, P. Dayan, B. Frey, and R. Neal, "The "wake-sleep" algorithm for unsupervised neural networks," *Science*, vol. 268, no. 5214, pp. 1158–1161, 1995.

[82] J. Schmidhuber, "Learning Complex, Extended Sequences Using the Principle of History Compression," *Neural Computation*, vol. 4, no. 2, pp. 234–242, March 1992.

[83] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A Training Algorithm for Optimal Margin Classifiers," in *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory (COLT)*, 1992, pp. 144–152.

[84] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*, June 2009, pp. 248–255.

[85] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, and G. Wang, "Recent Advances in Convolutional Neural Networks," *CoRR*, vol. abs/1512.07108, 2015.

[86] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.

[87] X. Glorot, A. Bordes, and Y. Bengio, "Deep Sparse Rectifier Neural Networks," in *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS-11)*, vol. 15, 2011, pp. 315–323.

[88] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, vol. arXiv:1207.0580, 2012.

[89] S. Wang and C. Manning, "Fast dropout training," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, vol. 28, May 2013, pp. 118–126.

[90] L. J. Ba and B. Frey, "Adaptive Dropout for Training Deep Neural Networks," in *Advances in Neural Information Processing Systems 26 (NIPS)*, 2013, pp. 3084–3092.

[91] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using Convolutional Networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 648–656.

[92] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, 2015, pp. 448–456.

[93] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks," in *International Conference on Learning Representations (ICLR 2014)*, 2013.

[94] M. D. Zeiler and R. Fergus, *Visualizing and Understanding Convolutional Networks*. Springer International Publishing, September 2014, pp. 818–833.

[95] A. G. Howard, "Some Improvements on Deep Convolutional Neural Network Based Image Classification," *CoRR*, vol. abs/1312.5402, 2013.

[96] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[97] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going Deeper with Convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.

[98] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 580–587.

[99] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[100] Y. Zhining and P. Yunming, "The genetic convolutional neural network model based on random sample," *International Journal of u- and e- Service, Science and Technology*, vol. 8, no. 11, pp. 317–326, 2015.

[101] M. Paulin, M. Douze, Z. Harchaoui, J. Mairal, F. Perronin, and C. Schmid, "Local Convolutional Features with Unsupervised Training for Image Retrieval," in *IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 91–99.

[102] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer, "Discriminative Learning of Deep Convolutional Feature Point Descriptors," in *IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 118–126.

[103] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-Based Convolutional Networks for Accurate Object Detection and Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 38, no. 1, pp. 142–158, Jan 2016.

[104] I. Ševo and A. Avramović, "Convolutional Neural Network Based Automatic Object Detection on Aerial Images," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 5, pp. 740–744, May 2016.

[105] T. Kong, A. Yao, Y. Chen, and F. Sun, "HyperNet: Towards Accurate Region Proposal Generation and Joint Object Detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 845–853.

[106] J. Li, Y. Wei, X. Liang, J. Dong, T. Xu, J. Feng, and S. Yan, "Attentive Contexts for Object Detection," *IEEE Transactions on Multimedia*, vol. 19, no. 5, pp. 944–954, May 2017.

[107] S. Sáfár and M. H. Yang, "Learning shape priors for object segmentation via neural networks," in *IEEE International Conference on Image Processing (ICIP)*, Sept 2015, pp. 1835–1839.

[108] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 39, no. 6, pp. 1137–1149, June 2017.

[109] R. Girshick, "Fast R-CNN," in *IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 1440–1448.

[110] H. Zhou, J. Zhang, J. Lei, S. Li, and D. Tu, "Image semantic segmentation based on FCN-CRF model," in *International Conference on Image, Vision and Computing (ICIVC)*, Aug 2016, pp. 9–14.

[111] M. J. Shafiee, A. Wong, and P. Fieguth, "Deep Randomly-Connected Conditional Random Fields For Image Segmentation," *IEEE Access*, vol. 5, pp. 366–378, 2017.

[112] G. Lin, C. Shen, A. v. d. Hengel, and I. Reid, "Efficient Piecewise Training of Deep Structured Models for Semantic Segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 3194–3203.

[113] Y. Cao, C. Shen, and H. T. Shen, "Exploiting Depth From Single Monocular Images for Object Detection and Semantic Segmentation," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 836–846, Feb 2017.

[114] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning Depth from Single Monocular Images Using Deep Convolutional Neural Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 38, no. 10, pp. 2024–2039, Oct 2016.

[115] J. R. Jensen, *Introductory digital image processing : a remote sensing perspective*, 3rd ed. Prentice Hall, 2005.

[116] D. Lu, P. Mausel, E. Brondizio, and E. Moran, "Change detection techniques," *International Journal of Remote Sensing*, vol. 25, no. 12, pp. 2365–2401, 2004.

[117] X. Chen, J. Chen, Y. Shi, and Y. Yamaguchi, "An automated approach for updating land cover maps based on integrated change detection and classification methods," *Journal of Photogrammetry and Remote Sensing (ISPRS)*, vol. 71, pp. 86–95, 2012.

[118] M. Hussain, D. Chen, A. Cheng, H. Wei, and D. Stanley, "Change detection from remotely sensed images: From pixel-based to object-based approaches," *Journal of Photogrammetry and Remote Sensing (ISPRS)*, vol. 80, pp. 91–106, 2013.

[119] A. Singh, "Digital change detection techniques using remotely-sensed data," *International Journal of Remote Sensing*, vol. 10, no. 6, pp. 989–1003, 1989.

[120] B. Veettil, "A Comparative Study of Urban Change Detection Techniques Using High Spatial Resolution Images," in *GEOBIA 2012*, 2012, pp. 29–34.

[121] M. İlsever and C. Unsalan, "Two-Dimensional Change Detection Methods: Remote Sensing Applications," 2012.

[122] T. Sohl, "Change Analysis in the United Arab Emirates: an investigation of techniques," *Photogrammetric Engineering and Remote Sensing*, vol. 65, pp. 475–484, 1999.

[123] E. Rignot and J. van Zyl, "Change detection techniques for ERS-1 SAR data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 31, pp. 896–906, 1993.

[124] R. Wiemker, A. Speck, D. Kulbach, H. Spitzer, and J. Bienlein, "Unsupervised robust change detection on multispectral imagery using spectral and spatial features," in *Proceedings of the 3rd International Airborne Remote Sensing Conference and Exhibition*, Jul 1997, pp. 640–647.

[125] J. Chen, P. Gong, C. He, R. Pu, and P. Shi, "Land-use/land-cover change detection using improved change-vector analysis," *Photogrammetric Engineering and Remote Sensing*, vol. 69, pp. 369–379, 2003.

[126] K. Johansen, L. Arroyo, S. Phinn, and C. Witte, "Comparison of geo-object based and pixel-based change detection of riparian environments using high spatial resolution multi-spectral imagery," *Photogrammetric Engineering and Remote Sensing*, vol. 76, pp. 123–136, 2010.

[127] D.-M. Tsai and C.-T. Lin, "Fast normalized cross correlation for defect detection," *Pattern Recognition Letters*, vol. 24, no. 15, pp. 2625–2631, 2003.

[128] X. Wan, J. G. Liu, and H. Yan, "The Illumination Robustness of Phase Correlation for Image Alignment," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 10, pp. 5746–5759, Oct 2015.

[129] B. Shen and C. Chen, "Bhattacharyya distance based video scene change detection," in *Proc. SPIE*, vol. 7496, 2009, pp. 1–6.

[130] J. Aguirre-Gutiérrez, A. Seijmonsbergen, and J. Duivenvoorden, "Optimizing land cover classification accuracy for change detection, a combined pixel-based and object-based approach in a mountainous area in Mexico," *Applied Geography*, vol. 34, pp. 29–37, 2012.

[131] M. Bock, P. Xofis, J. Mitchley, G. Rossner, and M. Wissen, "Object-oriented methods for habitat mapping at multiple scalescase studies from Northern Germany and Wye Downs," *UK. Journal for Nature Conservation*, vol. 13, pp. 75–89, 2005.

[132] J. Freixenet, X. Muñoz, D. Raba, J. Martí, and X. Cufí, "Yet Another Survey on Image Segmentation: Region and Boundary Information Integration," in *Proceedings of the 7th European Conference on Computer Vision-Part III (ECCV)*. London, UK: Springer-Verlag, 2002, pp. 408–422.

[133] L. Qiu, Y. Ding, Q. Yu, W. Yu, and X. Liu, "Bayesian change detection based on space contextual information and EM algorithm," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, July 2012, pp. 2249–2252.

[134] X. Liu and R. G. Lathrop Jr, "Urban change detection based on an artificial neural network," *International Journal of Remote Sensing*, vol. 23, no. 12, pp. 2513–2518, 2002.

[135] C. Huang, K. Song, S. Kim, J. Townshend, P. Davis, J. Masek, and S. Goward, "Use of a dark object concept and support vector machines to automate forest cover change analysis," *Remote Sensing of Environment*, vol. 112, pp. 970–985, 2008.

[136] F. Bovolo, L. Bruzzone, and M. Marconcini, "A novel approach to unsupervised change detection based on a semi supervised SVM and a similarity measure," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, pp. 2070–2082, 2008.

[137] A. Makkeasorn, N.-B. Chang, and J. Li, "Seasonal change detection of riparian zones with remote sensing images and genetic programming in a semi-arid watershed," *Journal of Environmental Management*, vol. 90, pp. 1069–1080, 2009.

[138] S. Sesnie, P. Gessler, B. Finegan, and S. Thessler, "Integrating Landsat TM and SRTM-DEM derived variables with decision trees for habitat classification and change detection in complex neotropical environments," *Remote Sensing of Environment*, vol. 112, pp. 2145–2159, 2008.

[139] A. Smith, "Image segmentation scale parameter optimization and land cover classification using the Random Forest algorithm," *Journal of Spatial Science*, vol. 55, pp. 69–79, 2010.

[140] H. Nemmour and Y. Chibani, "Multiple support vector machines for land cover change detection: An application for mapping urban extensions," *Journal of Photogrammetry and Remote Sensing*, vol. 61, no. 2, pp. 125–133, 2006.

[141] K. Zong, A. Sowmya, and J. Trinder, "Building Change Detection Based on Markov Random Field: Exploiting Both Pixel and Corner Features," in *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, Nov 2015, pp. 1–7.

[142] T. Hoberg, F. Rottensteiner, R. Q. Feitosa, and C. Heipke, "Conditional Random Fields for Multitemporal and Multiscale Classification of Optical Satellite Imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 2, pp. 659–673, Feb 2015.

[143] T. Sziranyi and M. Shadaydeh, "Improved segmentation of a series of remote sensing images by using a fusion MRF model," in *11th International Workshop on Content-Based Multimedia Indexing (CBMI)*, June 2013, pp. 137–142.

[144] L. Wei, Y. Zhong, L. Zhang, and P. Li, "New Unsupervised change detection algorithm based on the MRF in remote sensing image," in *Joint Urban Remote Sensing Event*, April 2011, pp. 325–328.

[145] O. Yousif and Y. Ban, "Improving SAR-Based Urban Change Detection by Combining MAP-MRF Classifier and Nonlocal Means Similarity Weights," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 10, pp. 4288–4300, Oct 2014.

[146] P. Liu, S. Sun, G. Li, J. Xie, and Y. Zeng, "Unsupervised change detection on remote sensing images using non-local information and Markov Random Field Models," in *IEEE International Geoscience and Remote Sensing Symposium*, July 2012, pp. 2245–2248.

[147] L. Zhou, G. Cao, Y. Li, and Y. Shang, "Change Detection Based on Conditional Random Field With Region Connection Constraints in High-Resolution Remote Sensing Images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 8, pp. 3478–3488, Aug 2016.

[148] M. Möller, L. Lymburner, and M. Volk, "The comparison index: a tool for assessing the accuracy of image segmentation," *International Journal of Applied Earth Observation and Geoinformation*, vol. 9, pp. 311–321, 2007.

[149] D. Liu and F. Xia, "Assessing object-based classification: advantages and limitations," *Remote Sensing Letters*, vol. 1, pp. 187–194, 2010.

[150] J.-W. Chan, K.-P. Chan, and A.-O. Yeh, "Detecting the nature of change in an urban environment: a comparison of machine learning algorithms," *Photogrammetric Engineering and Remote Sensing*, vol. 67, pp. 213–225, 2001.

[151] G. Doxani, K. Karantzalos, and M. Tsakiri-Strati, "Monitoring urban changes based on scale-space filtering and object-oriented classification," *International Journal of Applied Earth Observation and Geoinformation*, vol. 15, pp. 38–48, 2012.

[152] J. Vrabel, "Multispectral imagery advanced band sharpening study," *Photogrammetric Engineering and Remote Sensing*, vol. 66, no. 1, pp. 73–79, Jan 2000.

[153] W. Hallada and S. Cox, "Image sharpening for mixed spatial and spectral resolution satellite systems," *International Symposium on Remote Sensing of Environment*, vol. 3, pp. 1023–1032, May 1983.

[154] S. Dawn, V. Saxena, and B. Sharma, "Remote Sensing Image Registration Techniques: A Survey," in *Proceedings of the 4th International Conference on Image and Signal Processing (ICISP)*, 2010, pp. 103–112.

[155] H.-m. Chen, M. K. Arora, and P. K. Varshney, "Mutual information based image registration for remote sensing data," *Int. J. Remote Sens*, vol. 24, pp. 3701–3706, 2003.

[156] J. N. Sarvaiya, S. Patnaik, and S. Bombaywala, "Image Registration by Template Matching Using Normalized Cross-Correlation," in *International Conference on Advances in Computing, Control, and Telecommunication Technologies*, Dec 2009, pp. 819–822.

[157] A. Alba, R. M. Aguilar-Ponce, J. F. Vigueras-Gómez, and E. Arce-Santana, *Phase Correlation Based Image Alignment with Subpixel Accuracy*. Springer Berlin Heidelberg, 2013, pp. 171–182.

[158] J. Zhang, "A study on automated image registration based on straight line features," in *Joint Urban Remote Sensing Event*, May 2009, pp. 1–6.

[159] S. Alkaabi and F. Deravi, "A new approach to corner extraction and matching for automated image registration," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 5, July 2005, pp. 3517–3521.

[160] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[161] M. Hasan, X. Jia, A. Robles-Kelly, J. Zhou, and M. R. Pickering, "Multi-spectral remote sensing image registration via spatial relationship analysis on sift keypoints," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, July 2010, pp. 1011–1014.

[162] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep., 2009.

[163] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[164] N. Bourdis, D. Marraud, and H. Sahbi, "Constrained optical flow for aerial image change detection," in *IEEE International Geoscience and Remote Sensing Symposium*, July 2011, pp. 4176–4179.

[165] J. Cohen, "A Coefficient of Agreement for Nominal Scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, 1960.

[166] M. Vakalopoulou, K. Karantzalos, N. Komodakis, and N. Paragios, "Building detection in very high resolution multispectral data with deep learning features," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, July 2015, pp. 1873–1876.

[167] NASA Earth Observatory, "Measuring Vegetation (NDVI and ENVI)," http://earthobservatory.nasa.gov/Features/MeasuringVegetation/, 2000.

[168] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *European Conference on Computer Vision*, 2006, pp. 430–443.

[169] Itseez, "Open Source Computer Vision (OpenCV)," http://opencv.org/, Latest release 2016.

[170] N. Dalal, "Finding People in Images and Videos," Ph.D. dissertation, National Polytechnique de Grenoble, July 2006.

[171] V. Vapnik, *Estimation of Dependences Based on Empirical Data*. Springer-Verlag New York, Inc., 1982.

[172] S. Di Zenzo, "A Note on the Gradient of a Multi-image," *Computer Vision Graphics and Image Processing*, vol. 33, no. 1, pp. 116–125, Jan 1986.

[173] G. Sun, S. Wang, X. Liu, Q. Huang, Y. Chen, and E. Wu, "Accurate and efficient cross-domain visual matching leveraging multiple feature representations," *The Visual Computer*, vol. 29, no. 6-8, pp. 565–575, 2013.

[174] O. Canvet and F. Fleuret, "Efficient Sample Mining for Object Detection," in *Proceedings of the 6th Asian Conference on Machine Learning (ACML)*, vol. 29, 2014, pp. 1–15.

[175] J. Platt, "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods," in *Advances in Large Margin Classifiers*. MIT Press, 1999, pp. 61–74.

[176] J. Kittler, M. Hatef, R. Duin, and J. Matas, "On combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 20, no. 3, pp. 226–239, Mar 1998.

[177] MathWorks, "MATLAB: A numerical computing environment and programming language," https://www.mathworks.com/products/matlab.html, Latest release 2016.

[178] X. Wang, T. Han, and S. Yan, "An HOG-LBP human detector with partial occlusion handling," in *International Conference on Computer Vision*, Sep 2009, pp. 32–39.

[179] J. Zhang, K. Huang, Y. Yu, and T. Tan, "Boosted local structured HOG-LBP for object localization," in *International Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011, pp. 1393–1400.

[180] F. Bianconi and A. Fernndez, "On the Occurrence Probability of Local Binary Patterns: A Theoretical Study," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 3, pp. 259–268, 2011.

[181] M. Pietikainen, A. Hadid, G. Zhao, and T. Ahonen, "Local Binary Patterns for Still Images," in *Computer Vision Using Local Binary Patterns*, ser. Computational Imaging and Vision. Springer London, 2011, vol. 40, pp. 13–47.

[182] J. Xiao and G. Wu, "A Robust and Compact Descriptor Based on Center-Symmetric LBP," in *International Conference on Image and Graphics (ICIG)*, Aug 2011, pp. 388–393.

[183] S. Junding, Z. Shisong, and W. Xiaosheng, "Image retrieval based on an improved CS-LBP descriptor," in *IEEE International Conference on Information Management and Engineering (ICIME)*, April 2010, pp. 115–117.

[184] W. Xiaosheng and S. Junding, "An Effective Texture Spectrum Descriptor," in *International Conference on Information Assurance and Security (IAS)*, vol. 2, Aug 2009, pp. 361–364.

[185] A. Fitch, A. Kadyrov, W. Christmas, and J. Kittler, "Fast Robust Correlation," *IEEE Transactions on Image Processing*, vol. 14, no. 8, pp. 1063–1073, 2005.

[186] P. Huber, *Robust Statistics*, ser. Wiley Series in Probability and Statistics. New York: Wiley, 1981.

[187] R. Lienhart and J. Maydt, "An extended set of Haar-like features for rapid object detection," in *Proceedings of the International Conference on Image Processing*, vol. 1, 2002, pp. 900–903.

[188] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[189] R. E. Schapire and Y. Singer, "Improved Boosting Algorithms Using Confidence-rated Predictions," *Machine Learning*, vol. 37, no. 3, pp. 297–336, Dec 1999.

[190] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting," *The Annals of Statistics*, vol. 28, no. 2, pp. 337–374, 2000.

[191] A. J. Ferreira and M. A. Figueiredo, "Boosting Algorithms: A Review of Methods, Theory and Applications," in *Ensemble Machine learning: Methods and Applications*, C. Zhang and Y. Ma, Eds. Springer US, 2012, ch. 2, pp. 35–85.

[192] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society*, vol. Series B 39, no. 1, pp. 1–38, 1977.

[193] R. Tibshirani, G. Walther, and H. T., "Estimating the number of clusters in a data set via the gap statistic," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, pp. 411–423, 2001.

[194] J. Rouse, R. Haas, J. Scheel, and D. Deering, "Monitoring Vegetation Systems in the Great Plains with ERTS," in *3rd Earth Resource Technology Satellite (ERTS) Symposium*, vol. 1, 1974, pp. 48–62.

[195] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, Jan 1979.

[196] S. McFeeters, "Using the Normalized Difference Water Index (NDWI) within a Geographic Information System to Detect Swimming Pools for Mosquito Abatement: A Practical Approach," *Remote Sensing*, vol. 5, pp. 3544–3561, 2013.

[197] J. Liu, T. Fang, and D. Li, "Shadow Detection in Remotely Sensed Images Based on Self-Adaptive Feature Selection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 12, pp. 5092–5103, Dec 2011.

[198] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-Supervised Nets," in *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

[199] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. A. Riedmiller, "Striving for Simplicity: The All Convolutional Net," *arXiv preprint arXiv:1412.6806*, 2014.

[200] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[201] J. Pan, E. Sayrol, X. Giro-i Nieto, K. McGuinness, and N. E. O'Connor, "Shallow and Deep Convolutional Networks for Saliency Prediction," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[202] R. Srivastava, K. Greff, and J. Schmidhuber, "Highway networks," *arXiv preprint arXiv:1505.00387*, 2015.

[203] A. B. Graf, A. J. Smola, and S. Borer, "Classification in a normalized feature space using support vector machines," *IEEE Transactions on Neural Networks*, vol. 14, no. 3, pp. 597–605, May 2003.

[204] D. Konstantinidis, T. Stathaki, V. Argyriou, and N. Grammalidis, "Building Detection Using Enhanced HOG-LBP Features and Region Refinement Processes," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. PP, no. 99, pp. 1–18, 2016.

[205] J. Long, E. Shelhamer, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440, June 2015.

[206] N. Wadhwa, M. Rubinstein, F. Durand, and W. T. Freeman, "Phase-based Video Motion Processing," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 4, pp. 80:1–80:10, July 2013.

[207] S. Arora, A. Bhaskara, R. Ge, and T. Ma, "Provable Bounds for Learning Some Deep Representations," in *Proceedings of the 31th International Conference on Machine Learning (ICML)*, 2014, pp. 584–592.

[208] R. Aedla, G. Dwarakish, and D. Venkat Reddy, "Automatic Shoreline Detection and Change Detection Analysis of Netravati-GurpurRivermouth Using Histogram Equalization and Adaptive Thresholding Techniques," *Aquatic Procedia*, vol. 4, pp. 563–570, 2015.

[209] G. Zhang, Q. Chen, and Q. Sun, "Illumination Normalization Among Multiple Remote-Sensing Images," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 9, pp. 1470–1474, Sept 2014.

[210] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger, "Shiftable multi-scale transforms," *IEEE Trans. Info. Theory*, vol. 2, no. 38, pp. 587–607, 1992.

[211] A. Hast, "Robust and Invariant Phase Based Local Feature Matching," in *22nd International Conference on Pattern Recognition*, Aug 2014, pp. 809–814.

[212] M. Liang and X. Hu, "Recurrent convolutional neural network for object recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 3367–3375.

# APPENDIX A

## Proof of Euler Transform's Equality to Cosine-based Function

This appendix presents the analytical proof of the equality between the Euler transformation of feature vectors and the cosine-based distance function as presented in Eq. (4.3).

The objective is to show that the cosine-based distance between a pair of vectors $\mathbf{x_i}$ and $\mathbf{x_j}$ of size $L$ is equivalent to applying the $l_2$-norm in their Euler representation $\mathbf{z_i}$ and $\mathbf{z_j}$ respectively, where $\mathbf{z_i} = \frac{1}{\sqrt{2}} e^{i\alpha\pi\mathbf{x_i}}$. Applying the Euler formula $e^{ia} = cos(a) + isin(a)$ and substituting $\theta_\mathbf{i} = \alpha\pi\mathbf{x_i}$, we get

$$\|\mathbf{z_i} - \mathbf{z_j}\|^2 = \frac{1}{2}\sum_{l=1}^{L} \|(\cos(\theta_i(l)) + i\sin(\theta_i(l)) - (\cos(\theta_j(l)) + i\sin(\theta_j(l))\|^2 \tag{A.1}$$

Considering the following two trigonometric identities that describe the cosine and sine differences of two angles

$$\begin{aligned} cos(a) - cos(b) &= -2sin(\frac{a+b}{2})sin(\frac{a-b}{2}) \\ sin(a) - sin(b) &= 2cos(\frac{a+b}{2})sin(\frac{a-b}{2}) \end{aligned} \tag{A.2}$$

and applying them to Eq. (A.1), along with the Euler formula and the identity $i^2 = 1$, we get

$$\begin{aligned} \|\mathbf{z_i} - \mathbf{z_j}\|^2 &= \frac{1}{2}\sum_{l=1}^{L} \|2sin(\frac{\theta_i(l) - \theta_j(l)}{2})(-sin(\frac{\theta_i(l) + \theta_j(l)}{2}) + icos(\frac{\theta_i(l) + \theta_j(l)}{2}))\|^2 \\ &= \frac{1}{2}\sum_{l=1}^{L} \|2sin(\frac{\theta_i(l) - \theta_j(l)}{2})i(cos(\frac{\theta_i(l) + \theta_j(l)}{2}) + isin(\frac{\theta_i(l) + \theta_j(l)}{2}))\|^2 \\ &= \frac{1}{2}\sum_{l=1}^{L} \|2sin(\frac{\theta_i(l) - \theta_j(l)}{2})(ie^{i\frac{\theta_i(l)+\theta_j(l)}{2}})\|^2 \end{aligned} \tag{A.3}$$

Based on the identity $|e^{ia}| = 1$, the second term of Eq. (A.3) can be removed, leaving only the sine part. Finally, taking into consideration the half-angle trigonometric identity

$$sin(\frac{x}{2}) = \pm\sqrt{\frac{1 - cos(x)}{2}} \tag{A.4}$$

and substituting it to Eq. (A.3), we get the result

$$\begin{aligned} \|\mathbf{z_i} - \mathbf{z_j}\|^2 &= \frac{1}{2}\sum_{l=1}^{L} \| \pm 2\sqrt{\frac{1 - cos(\theta_i(l) - \theta_j(l))}{2}}\|^2 \\ &= \sum_{l=1}^{L}(1 - cos(\theta_i(l) - \theta_j(l)) \\ &= 1 - cos(\alpha\pi\mathbf{x_i} - \alpha\pi\mathbf{x_j}) \end{aligned} \tag{A.5}$$

Eq. (A.5) proves and verifies the equality between the cosine-based distance function that was presented in Eq. (4.2) and the $l_2$-norm difference between the Euler representations of the feature vectors.

# APPENDIX B

## *Modular*-CNN for CIFAR-10 and MNIST classification tasks

This appendix presents additional experiments and results of the performances of the proposed *Modular*-CNN and PlainNet on two benchmark image classification datasets. More specifically, the *Modular*-CNN and PlainNet are compared with and without the proposed novel layers on the CIFAR-10 and MNIST datasets and the effect of the novel layers on these classification tasks is evaluated. The purpose is to evaluate how important the development of a general architecture, such as the *Modular*-CNN, that can be optimised with respect to the classification problem is and how effective the proposed NL and ETL layers are as general CNN layers.

It should be mentioned that the hyper-parameters of the *modules* that the *Modular*-CNN consists of are optimised with respect to the classification task that needs to be solved. However, other hyper-parameters that affect the performance of the *Modular*-CNN, such as the dimensionality of the output feature vectors, batch size, learning rate and momentum are kept fixed with the optimal values that were computed for the building detection task and they are presented in Section 4.5.1.

## B.1 Experimentation on CIFAR-10 dataset

The CIFAR-10 [162] is a well-known dataset used for multi-class image recognition tasks. The small images of the dataset (i.e. $32 \times 32$ pixels) renders it appropriate for the testing of the performance of a shallow CNN with few layers. The RGB colour images of the dataset are transformed to the YUV colour space and normalised to have zero mean and unity variance before they are fed to the CNNs for training and testing.

We optimise the *Modular*-CNN using the procedure described in Section 4.5.1. The optimisation procedure leads to the CNN architecture described in Fig. B.1, where only two sequential *modules* are employed. The experiments show that the addition of any parallel *module* leads to a degradation in the generalisation performance of the *Modular*-CNN on the CIFAR-10 dataset. We train our proposed *Modular*-CNN and PlainNet with and without the novel NL and ETL layers on the CIFAR-10 training set and without performing any data augmentation. We then evaluate their performances on the CIFAR-10 test set. The results are presented in Table B.1.

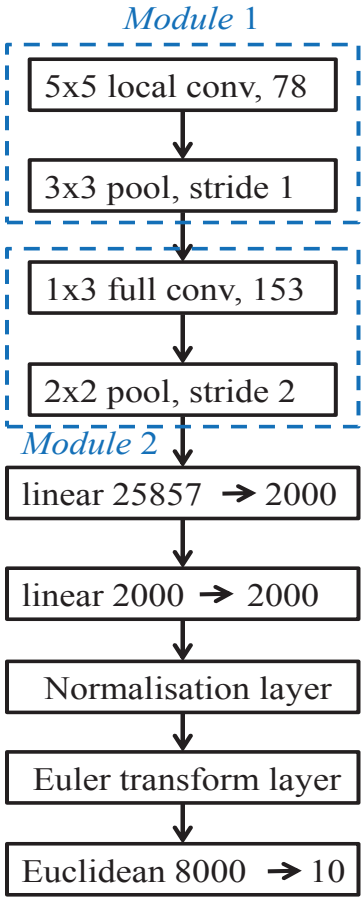


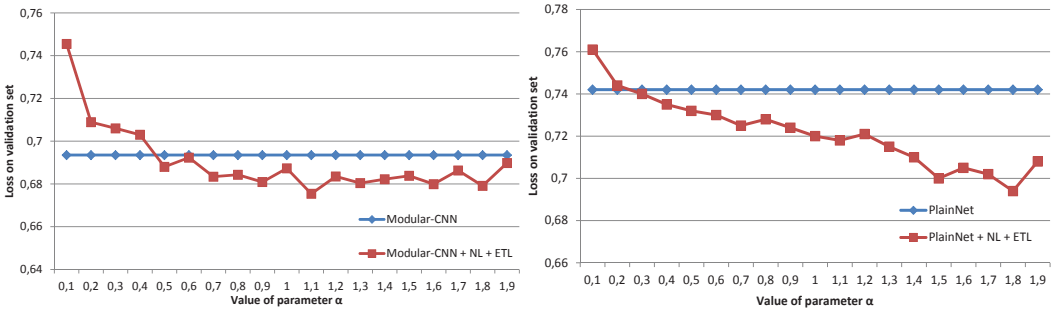

Figure B.1: Optimal *Modular*-CNN architecture for the CIFAR-10 dataset. In the convolutional layers, the number of output filters is shown last, while in the fully connected blocks, the transition from $M$ to $N$ features is depicted. The ReLU units after each convolutional and linear layer are omitted.

Table B.1: Classification performance on the CIFAR-10 dataset. The results are presented in the format: mean ± std.

| Method | Error (%) | # params |
|---|---|---|
| PlainNet [14] | 25.94 ± 0.168 | ≈ 16.94M |
| PlainNet+NL | 25.35 ± 0.177 | ≈ 16.98M |
| PlainNet+NL+ETL | 25.33 ± 0.169 ($\alpha$=1.5) | ≈ 17.07M |
| *Modular*-CNN | 21.75 ± 0.133 | ≈ 55.78M |
| *Modular*-CNN+NL | 21.29 ± 0.145 | ≈ 55.8M |
| *Modular*-CNN+NL+ETL | 21.28 ± 0.141 ($\alpha$=1.8) | ≈ 55.85M |

From Table B.1 a few conclusions can be drawn. First of all, the addition of the NL layer improves the performance of both our *Modular*-CNN and PlainNet with respect to their performance without the NL layer. From Fig. B.2, it can be concluded that the introduction of the ETL layer leads to a smaller loss on the validation set and thus, better generalisation ability of the *Modular*-CNN. However, the better generalisation ability is not reflected on the test error as the introduction of the ETL layer improves only slightly the performance of both our *Modular*-CNN and PlainNet on the test set. The value of the parameter $\alpha$ that achieves the optimal results for the ETL layer is included next to the corresponding error in Table B.1. Finally, our *Modular*-CNN with the

novel layers outperforms by about 5.4% the PlainNet with the novel layers. Although PlainNet demonstrates a larger depth, capable of learning complex features, the shallow *Modular*-CNN with a higher number of parameters can more effectively describe the CIFAR-10 dataset.




(a) (b)

Figure B.2: Performance of the *Modular*-CNN and PlainNet with and without the novel NL and ETL layers with respect to the average loss on the validation set for the CIFAR-10 dataset.

## B.2 Experimentation on MNIST dataset

The MNIST [163] dataset is a benchmark dataset for the classification of handwritten digits. The small size of the image patches (i.e. $28 \times 28$ pixels) renders the dataset suitable for the testing of our shallow *Modular*-CNN with the proposed layers. The images are normalised to have zero mean and unity variance before they are employed for the training and testing of the CNNs.

Similar to all previous experimentations, the *Modular*-CNN is optimised by adding *modules* until the performance of the CNN does not improve anymore. The optimal *Modular*-CNN architecture consists of two sequential *modules* and it is depicted in Fig. B.3. The addition of any parallel *module* leads to a decrease in the accuracy of the *Modular*-CNN.

We train our proposed *Modular*-CNN and PlainNet on the MNIST dataset with and without the novel NL and ETL layers and without performing any data augmentation. We then evaluate their performances on the MNIST test set. The results are presented in Table B.2, where they are compared with other state-of-the-art algorithms. Furthermore, the effect of the novel layers on the validation set for various values of $\alpha$ and for both PlainNet and *Modular*-CNN are presented in Fig. B.4.

From Table B.2 and Fig. B.4, it can be deduced that the addition of the novel NL and ETL layers has a beneficial effect on the performance of both our *Modular*-CNN and PlainNet. Furthermore, the addition of the ETL layer leads to a smaller loss on the validation set and consequently, a better generalisation ability of the *Modular*-CNN and PlainNet. The optimal value of the parameter $\alpha$ for the ETL layer is reported next to the corresponding error in Table B.2. Finally, our *Modular*-CNN outperforms PlainNet and achieves comparable performance with the other state-of-the-art methods, although with a much larger number of parameters and thus a better learning capacity.

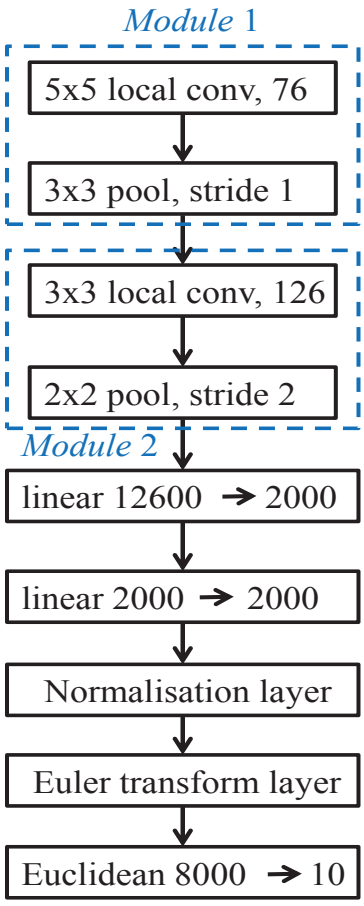


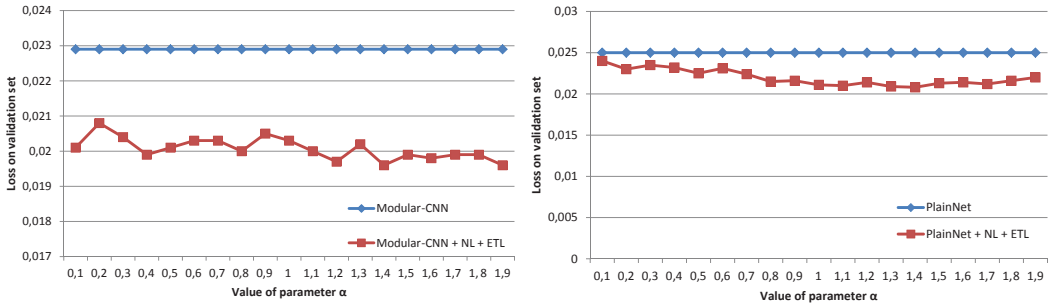

Figure B.3: Optimal *Modular*-CNN architecture for the MNIST dataset. In the convolutional layers, the number of output filters is shown last, while in the fully connected blocks, the transition from $M$ to $N$ features is depicted. The ReLU units after each convolutional and linear layer are omitted.

Table B.2: Classification performance on the MNIST dataset. The results are presented in the format: mean ± std.

| Method | Error (%) | # params |
|---|---|---|
| Maxout [12] | 0.45 | 0.42M |
| NIN [13] | 0.47 | 0.35M |
| DSN [198] | 0.39 | 0.35M |
| RCNN-96 [212] | 0.31 | 0.67M |
| PlainNet [14] | 0.71 ± 0.057 | ≈ 9.99M |
| PlainNet+NL | 0.69 ± 0.059 | ≈ 10.02M |
| PlainNet+NL+ETL | 0.63 ± 0.055 ($\alpha$=1.4) | ≈ 10.08M |
| *Modular*-CNN | 0.54 ± 0.049 | ≈ 29.28M |
| *Modular*-CNN+NL | 0.53 ± 0.051 | ≈ 29.3M |
| *Modular*-CNN+NL+ETL | 0.51 ± 0.05 ($\alpha$=1.4) | ≈ 29.34M |

## B.3 Conclusions

A few conclusions can be drawn from the experimentation on the CIFAR-10 and MNIST datasets. Firstly, although the proposed *Modular*-CNN outperforms PlainNet on both datasets, the param-




(a) (b)

Figure B.4: Performance of the *Modular*-CNN and PlainNet with and without the novel NL and ETL layers with respect to the average loss on the validation set for the MNIST dataset.

eters of the *Modular*-CNN are $2-4\times$ the parameters of the PlainNet. On the one hand, the much larger number of parameters of a CNN gives it a higher learning capacity, thus enabling a CNN to learn better the classification task. On the other hand, it is harder to optimise a CNN with increased number of parameters as the searching space that the parameter values form is of a higher dimensionality and the training is more probable to get stuck to local optima. This means that the *Modular*-CNN has both advantages and drawbacks with respect to the PlainNet.

Secondly, the novel NL and ETL layers have a positive impact on the performances of both our shallow *Modular*-CNN and the deeper PlainNet, which means that these layers have the potential to be employed as general CNN layers. However, more experimentation is required to verify that the NL and ETL layers can be employed as general CNN layers. This is because the improvement that is achieved by the use of the novel layers can be attributed to the increased number of CNN parameters due to the classification of the larger extracted feature vectors.

Finally, the general architecture of the proposed *Modular*-CNN that consists of CNN building blocks (i.e. *modules*) can be optimised based on the problem at hand. The results show that the *Modular*-CNN adapts well to the classification tasks and leads to high performances. For the MNIST dataset, especially, the *Modular*-CNN can reach performances that are close to state-of-the-art methodologies. This conclusion can open the way towards proposing other more powerful CNNs that can be optimised with respect to the problem that needs to be solved without significant changes in their architectures.