Capturing Correlations of Local Features for Image Representation[☆]

Xiaopeng Hong^{a,*}, Guoying Zhao^a, Stefanos Zafeiriou^b, Maja Pantic^b,
Matti Pietikäinen^a

^a*Department of Computer Science and Engineering, University of Oulu, Finland*

^b*Department of Computing, Imperial College London, UK*

Abstract

Local descriptors are popular ways to characterize the *local* properties of images in various computer vision based tasks. To form the *global* descriptors for the image regions, the first-order feature pooling is widely used. However, as the first-order pooling technique treats each dimension of local features separately, the pairwise correlations of local features are usually ignored.

Encouraged by the success of recently developed second-order pooling techniques, in this paper we formulate a general second-order pooling framework and explore several analogues of the second-order average and max operations. We comprehensively investigate a variety of moments which are in the central positions to the second-order pooling technique. As a result, the superiority of the second-order standardized moment average pooling (*2Standmap*) is suggested. We successfully apply 2Standmap to four challenging tasks namely texture classification, medical image analysis, pain expression recognition, and micro-expression recognition. It illustrates the effectiveness of 2Standmap to capture multiple cues and the generalization to both static images and spatial-temporal sequences.

Keywords: Local feature, feature pooling, covariance matrix

[☆]This work is supported by the Academy of Finland and Infotech Oulu

*Corresponding author

Email addresses: xiaopeng.hong@ee.oulu.fi (Xiaopeng Hong), gyzhao@ee.oulu.fi (Guoying Zhao), s.zafeiriou@imperial.ac.uk (Stefanos Zafeiriou), m.pantic@imperial.ac.uk (Maja Pantic), mkp@ee.oulu.fi (Matti Pietikäinen)

1. Introduction

How to extract informative cues from images plays a central role in various computer vision based tasks such as texture classification, facial analysis, and medical image processing. Image representations are expected to have the discriminative information for distinguishing instances from different classes and also can handle the possibly large intra-class divergence in appearance, such as illumination, color, rotation, and scale. As a result, local features, which characterize the properties of small image patches, are of great interest [1, 2]. The local features are proven informative and robust against noise and background clutters [1, 3, 4, 5, 6, 7]. Feature pooling then, summarizes the local features within an image area and form a global description [8].

Most approaches rely on the first-order pooling, typically implemented by an average or a max operation [8]. The former one usually works together with the popular bag-of-words framework [1, 3, 4, 5, 6, 7] to approximate the distribution of visual words, while the latter is normally in conjunction with the sparse coding framework [9, 10, 11]. These approaches achieve promising results in a number of applications such as texture classification, object classification, and scene classification. However, as the first-order pooling treats each dimension of local features separately, the correlations between local features, which are important for a non-periodic and higher-level appearance analysis, are inevitably ignored.

More recently, Carreira et al. propose the second-order pooling technique to take into account the feature correlations [12]. The investigation in the second-order average and max operations indicates that the second-order pooling technique with appropriate non linear mappings¹ substantially outperforms the commonly used first-order one.

Encouraged by the success of second-order pooling techniques, in this pa-

¹Usually, the second-order average pooling is incorporated with the Log-Euclidean mapping, and the second-order max pooling is followed by the power normalisation.

per we formulate a more general second-order pooling framework which enables to explore several analogues of the second-order average and max operations. Through comprehensive investigation of a variety of moments which are in the central positions to the second-order pooling technique, the superiority of the second-order standardized moment average pooling (*2Standmap*) is suggested. We then successfully apply the 2Standmap to four challenging applications namely texture classification, medical image analysis, pain expression recognition, and micro-expression recognition. Promising results on these four applications illustrates the effectiveness of 2Standmap to capture multiple cues and the generalization to both static images and spatial-temporal sequences.

The remainder of this paper is organized as follows: Section 2 introduces the related work. The second order pooling framework is summarized and analysed in detail in Section 3. Section 4 provides experimental results of comprehensive studies and the applications in four tasks. Finally, the conclusion is given in Section 5.

2. Related Work

The most relevant works to ours are the coder-free second-order pooling approaches. In [12], the second-order average and max operations (hereafter termed *2AveP* and *2MaxP* respectively) are investigated to characterise super pixels for semantic segmentation. The second-order feature pooling takes into account the pairwise correlation of local features. Considering its superiority in performance compared with the first-order pooling method, the second-order pooling is applied to a few hot topics such as object recognition, human pose estimation [13], touch saliency prediction [14], and face representation [15], etc. Moreover, the covariance matrix (CovM) [16, 17, 18, 19, 20] also belongs to the same umbrella. Distinct from the *2AveP* which focuses on the raw moment (moments about zero) in [12], the CovM uses the second order central-moment (namely the moments about the mean) to characterize the pairwise correlation of low level local features.

To the best knowledge of the authors, there is no explicit comparison at the variety of second order moments. In this paper, we aim to fill this gap by carrying out a comprehensive study on this issue in the context of texture classification and medical image analysis. For fair comparisons, besides those simple raw features, such as the intensity (or the intensity of the three color channels), the first and the second-order partial derivatives with respect to x and y as recommended in [20], we also evaluate three groups of typical local features, which include the histogram based descriptors (as used in [12]), the orthogonal transform based descriptors, and the statistics based descriptors. According to the experimental results, the second-order standardized central moment average pooling (2Standmap) is suggested. We further apply the 2Standmap to four challenging applications, and show that this simple coder-free pooling technique performs well even with the commonly used classifiers such as the nearest neighbour classier and the linear support vector machine.

3. The Second Order Pooling Framework

Fig. 1 illustrates the overview of the investigated framework combining low-level raw features and mid-level local descriptors. Given an image region $\mathcal{I}$ with a size of $W \times H$, for each pixel $\mathbf{x} = (x, y)$, a vector $\mathbf{f}(\mathbf{x})$ concatenating all local features is extracted. We then perform the second order pooling to obtain the global descriptor $\mathbf{g}(\mathcal{I})$.

3.1. The second order pooling framework

2nd order average and max operation: Given the d -dimensional $\mathbf{f}(\mathbf{x})$ at pixel $\mathbf{x}$, we apply some pre-defined mapping θ and obtain the pre-processed feature vectors $\mathbf{p}(\mathbf{x}) = \theta(\mathbf{f}(\mathbf{x}))$. We then employ the *second order average pooling* [12] to summarize the pair wise connection of features as a matrix:

$$\mathbf{G}_{2AvgP}(\mathcal{I}) = c \sum_{\mathbf{x} \in \mathcal{I}} \mathbf{p}(\mathbf{x})\mathbf{p}(\mathbf{x})^T, \quad (1)$$

and *second order max pooling* [12], as the following matrix

$$\mathbf{G}_{2MaxP}(\mathcal{I}) = \max_{\mathbf{x} \in \mathcal{I}} \mathbf{p}(\mathbf{x})\mathbf{p}(\mathbf{x})^T, \quad (2)$$

Figure 1: Overview of the second pooling framework.

where c is a normalization factor and the max operation is performed to the outer products of $\mathbf{p}$ in a per-entry manner.

A Variety of Moments

According to different points the moments are about, there are two types of second moments [21] commonly used. The central moments are the moments about the mean. And the raw moments refer to the moments about zero. More specifically, for central moments, the pre-processing function θ is expressed as

$$\theta_{cen}(\mathbf{f}(\mathbf{x})) = \mathbf{f}(\mathbf{x}) - \mu, \quad (3)$$

where μ is the mean of $\{\mathbf{f}(\mathbf{x})\}_{\mathbf{x} \in \mathcal{I}}$. In the case of raw moments, it becomes:

$$\theta_{raw}(\mathbf{f}(\mathbf{x})) = \mathbf{f}(\mathbf{x}) - \mathbf{0} = \mathbf{f}(\mathbf{x}). \quad (4)$$

Moreover, according to whether the features are normalized, the moments can be categorized to two types: the normalised and the unnormalised moments [21]. The former ones are the moments divided by the corresponding variances, while the latter preserved the scale of the original feature vectors.

As a result, there are four combinations of moments, namely the *normalised central moment* (a.k.a. *standardized moment*), the *normalised raw moment*, the *unnormalised central moment*, and the *unnormalised raw moment*.

It is notable that the second order average pooling used in [12] is the *unnormalised raw moment* here and the correlation matrix descriptor [20] is the

standardized moment. This paper thus provides opportunities of comparing all these four moments with both average and max pooling respectively for the first time.

Non-linear Mapping:

Both the $d \times d$ matrices $\mathbf{G}_{2AvgP}$ and $\mathbf{G}_{2MaxP}$ are usually incorporated with appropriate non linear mappings to achieve satisfying performance.

The second-order average pooling results in symmetric and positive semi-definite (SPD) matrices, which form a Riemannian manifold [12]. The logarithmic mapping *logm* [22] maps a point $\mathbf{G}$ on manifolds to its corresponding image $\mathbf{g}$ in vector spaces. More specifically,

$$\mathbf{g}_{2AvgP}(\mathcal{I}) = \text{logm}(\mathbf{G}_{2AvgP}(\mathcal{I})). \quad (5)$$

In case that some matrices obtained via second-order average pooling are not positive definite, we add a small positive number in the diagonal entries of the matrix obtained by Eq. (3) to avoid the non-positive definite problem, though this case rarely happen in practice.

The second-order max pooling matrix is not SPD and hence the matrix logarithmic mapping in Eq. (5) cannot be applied to the second-order max pooling matrix $\mathbf{G}_{2MaxP}$. Instead, the power normalization mapping [23] is used. For each entry $G(i, j)$ of $\mathbf{G}_{2MaxP}$, we have

$$g_{2MaxP}(i, j) = \text{sign}(G(i, j)) |G(i, j)|^\alpha, \quad (6)$$

where α is parameter controlling the magnitude, which is usually within the range of $(0, 1]$.

As either $\mathbf{G}_{2AvgP}$ or $\mathbf{G}_{2MaxP}$ is symmetric, the upper triangle elements are unpacked and concatenated into a vector to form the global representations.

3.2. Elementary Features Extraction

The design and selection of local features are important since it is sufficient correlated local features that ensure the discriminative power of the global descriptors. Existing features can be briefly categorized into two sets.

The first set contains the low-level regular features. Basically, according to different sources of features, most of the existing elementary features can be categorized into three classes: 1). physical quantities related features, such as the intensity [16]; 2). feature coming from digital filter convolutions such as the gradients [16, 17, 19, 18, 20] or the responses of Gabor filters [19]; and 3). the elementary functions of the former two, for example the magnitudes and the orientations [18, 20]. Some popular choices are the intensity (or the intensity of the three color channels), the elementary functions of the first and the second-order partial derivative with respect to x and y , respectively.

The second one contains one or several (e.g., T) mid-level local descriptors of a patch centered at $\mathbf{x}$. The regular feature vector $\mathbf{f}_R$ and the vectorization of the local descriptors $\{\mathbf{f}_P^t\}$, $t = 1, \dots, T$ are concatenated into a feature vector $\mathbf{f}$. The remaining part of this subsection mainly focuses on how to obtain the feature vectors from the second set. For the regular features, readers may refer to [17, 18, 19, 16, 20] for more details.

Given the patch of interest centered at $\mathbf{x}$, i.e., $\mathcal{N}_{\sigma_x, \sigma_y}(\mathbf{x}) = \left\{ (x', y') \mid |x - x'| \leq \sigma_x, |y - y'| \leq \sigma_y \right\}$, where σ_x, σ_y are the patch support in 2D directions respectively, several typical local descriptors are described as follows:

Histogram based descriptors: Histograms are usually used as an approximation of the distribution of discrete variables. 1D or 2D dimensional histograms are computed covering the neighbouring pixels of $\mathbf{x}$. Take the 1D histogram $\mathbf{h}^{(1)}$ for an example. Suppose $\mathbf{h}^{(1)}$ are defined on one particular B -level quantized feature responses λ (e.g., the gradient orientation) $h(b)$, i.e., the b -th bin of $\mathbf{h}^{(1)}$, $b = 1, \dots, B$, is defined as follows:

$$h(b) = \sum_{\mathbf{x}' \in \mathcal{N}(\mathbf{x})} w(\mathbf{x}') \cdot \delta_b(\lambda(\mathbf{x}')), \quad (7)$$

where $\delta_b(t)$ is the Dirac function for which $\delta_b(t) = 1$ if $t = b$ and $\delta_b(t) = 0$ for otherwise; and $w(\cdot)$ is the weighting function. One out of the B bins of $\mathbf{h}^{(1)}$ stands for one specific pattern which is defined manually or obtained by vector quantization. The calculation of pixel-wise histograms can be accelerated by the

integral histogram technique [24]. The computational complexity is therefore $O(BWH)$.

Orthogonal transform based descriptors: 1D or 2D orthogonal transforms (OT) decorrelate the pixels in an image region via mapping the correlated spatial data into correlated coefficients. Let $\mathbf{I}$ denotes the identity matrix. Basically, given an image region $\mathbf{W} \in \mathbb{R}^{W \times H}$, the 1D OT can be expressed in a way of $\mathbf{T} = \mathbf{A}\mathbf{W}$, where $\mathbf{A}\mathbf{A}^T = \mathbf{I}$. For 2D OT , it usually has a form of $\mathbf{T} = \mathbf{L}\mathbf{W}\mathbf{R}$, where $\mathbf{L}$ and $\mathbf{R}$ are two orthogonal matrices. Orthogonal transform removes of redundancy between neighbouring pixels and thus has good decorrelation power. Moreover, it exhibits good energy compaction for highly correlated images, considering its ability to pack input data into only a few coefficients. It allows to discard coefficients with relatively small amplitudes without introducing visual distortion in the reconstructed image.

Note that in case of the 2D orthogonal transform, it leads to a tensor-based 4D image, where each pixel is a 2D matrix. We thereby select the first s (e.g., $s = 3 \sim 5$) components as elementary features according to some order, for example the Zigzag order to reach the vectorization of the descriptor. In this paper, the transforms which have basis independent of the patch such as the Discrete Cosine Transform (DCT) [25, 26] are preferred since they have efficient implementation according to the convolution theorem through the fast Fourier transform (FFT) with a complexity of $O(WH \log_2(WH))$. The computational complexity via FFT is therefore $O(sWH \log_2(WH))$, regardless of the patch support (σ_x, σ_y) .

Statistics based descriptors: the local covariance matrix approach, a.k.a., L^2ECM [27] is one typical statistics based descriptor. L^2ECM computes $\mathbf{M}(\mathcal{N}(\mathbf{x}))$, i.e., a covariance matrix covering several (saying $d_P = 2 \sim 5$) elementary features within $\mathcal{N}_{\sigma_x, \sigma_y}(\mathbf{x})$ according to Eq. 8 expressed in the following manner:

$$\mathbf{M}(\mathcal{I}) = c \sum_{\mathbf{x} \in \mathcal{I}} (\mathbf{f}(\mathbf{x}) - \mu)(\mathbf{f}(\mathbf{x}) - \mu)^T. \quad (8)$$

We then get its projection $\hat{\mathbf{M}}$ on the tangent space via logarithm operators.

Lastly, the L^2 ECM is expressed by the half-vectorization of $\hat{\mathbf{M}}$, which packing the upper triangular entries into a vector.

Comprehensive review of the Riemannian manifold related to the L^2 ECM descriptor is beyond the scope of this paper. Readers may refer to [20, 28, 17, 22, 18, 19] for recent advances and development on the Riemannian manifold.

Integral images can also be applied to achieve fast implementation [20]. The computational complexity of constructing the integral images is $O(n^2 WH)$. The most computational burden is associated with the eigen-decomposition of the local covariance matrix, results in the computational complexity of $O(n^3 WH)$.

Given the feature vector concatenating regular features and the vectorizations of local descriptors, we compute the second order pooling matrix as described in the previous sub-section to form the global descriptor $\mathbf{g}$. Note that in practice, the local descriptors are not limited to the ones described above. Indeed, any descriptor which can be expressed by a vector can be employed by the second order pooling framework. Different kinds of descriptors capture different cues and hence their combination provides enhanced discriminative information.

4. Experiments

In this section, we firstly evaluate the issue of moment types for the second order pooling framework. We then apply the best one to four applications from static images to dynamic video sequences. The programs are implemented in Matlab.

4.1. Investigation in Moments and Local Features

In this sub-section, we use the task of texture classification as an example to investigate in a variety of moments for both average and max pooling. The challenging KTH-TIPS2-a (abbreviated as TIPS2) dataset, is selected. TIPS2 [29] contains four samples of eleven different materials, totalling 4608 images. We follow the setup by Caputo *et al.* [29], where three of the four samples for each material are available for training, and the images from the remaining sample

are for testing. Random experiments are repeated for one hundred times and the average accuracy is reported. The parameter-free Nearest Neighbour classifier (NN) [29] is used unless otherwise noted, since our focus is on the representations. The Euclidean distance is used in NN for distance measurement.

To evaluate the effect of correlation of local features and regular raw features, we carry out experiments on the TIPS2 dataset using five commonly used raw feature sets, i.e. $\mathbf{f}_R = [R, G, B, |I_x|, |I_y|]^T$. We employ 8-bin Histogram of Gradient Orientation (HIGO) and the first 5 coefficients from 2D-DCT (excluding the DC component) for the local histogram and the local orthogonal transform descriptor respectively. The L²ECM is used as the statistics based descriptors, using I , $|I_x|$, $|I_y|$, $|I_{xx}|$, $|I_{yy}|$ as raw features. The local patch for this experiments are fixed to $\sigma_x = \sigma_y = 8$. We also report the results of combining the local HIGO and DCT features.

The results are listed in Table 1. We have the following remarks:

Firstly, the second order average pooling substantially outperforms the max pooling regardless of the combinations of local features. It is consistent with the observation in [12].

Secondly, for the second order average pooling, the normalised central moments and the normalised raw moments achieves better results than the unnormalised ones in general. Similar observations can be made about the max pooling, where the normalised central moments and the normalised raw moments perform similarly while the other two are much inferior. The reason of the superiority of the normalised moments lies in that the normalisation disregards the standard deviations of elementary features and thus greatly improves robustness.

Thirdly, when using the three local descriptors namely the HIGO, DCT, and the L²ECM solely, the differences in accuracy among these three are marginal.

Fourthly, the combination of HIGO and DCT (HIGO-DCT) further improves the accuracy of classification. However, the L²ECM fails to provide complementary information beyond either HIGO or DCT. As a result, either HIGO-L²ECM or DCT-L²ECM cannot outperform HIGO or DCT. Moreover, the combination

Table 1: Evaluation of different moments and pooling techniques. The number in the parentheses indicates the dimension of the local features used. The abbreviations ‘nc’, ‘nr’, ‘uc’, and ‘ur’ refer to the normalised central moment, normalised raw moment, unnormalised central moment, and unnormalised raw moment respectively.

Pooling	$2AveP$				$2MaxP$ ($\alpha = 0.25$)			
Moments	nc	nr	uc	ur	nc	nr	uc	ur
HIGO(13)	67.31	66.48	66.60	66.76	54.77	54.97	51.83	56.44
DCT(10)	68.07	63.75	69.59	69.59	59.21	60.50	46.61	45.26
L ² ECM(20)	68.30	66.21	64.54	64.15	51.02	49.08	50.23	47.90
HIGO-DCT(20)	73.38	71.61	70.10	71.53	63.40	62.36	56.59	58.56
HIGO-L ² ECM(28)	67.03	68.69	64.89	66.19	52.66	55.03	50.85	51.99
DCT-L ² ECM(25)	67.40	69.22	66.34	66.09	56.70	57.04	49.19	47.74
ALL(33)	68.01	69.54	65.72	67.15	57.58	59.19	51.53	51.61
Pooling	$2MaxP$ ($\alpha = 0.50$)				$2MaxP$ ($\alpha = 0.75$)			
Moments	nc	nr	uc	ur	nc	nr	uc	ur
HIGO(13)	54.92	54.47	49.55	54.16	54.65	54.18	46.92	51.71
DCT(10)	59.78	60.67	45.32	44.88	59.71	60.15	43.32	44.57
L ² ECM(20)	51.35	45.94	51.02	45.12	52.03	44.47	51.39	41.39
HIGO-DCT(20)	63.57	62.37	54.93	56.30	62.25	62.11	51.25	54.17
HIGO-L ² ECM(28)	52.75	55.01	47.17	47.83	52.90	55.01	45.32	43.24
DCT-L ² ECM(25)	57.96	57.57	46.93	45.55	58.39	58.23	44.99	42.42
ALL(33)	58.79	59.86	47.98	47.89	59.08	60.17	45.53	44.05

of the three does not show any superiority in accuracy either.

Finally, when comparing the two normalised moments namely ‘nc’ and ‘nr’, the normalised raw moment are the better choice for HIGO-L²ECM, DCT-L²ECM, and the combination of the three local features (‘ALL’ in Table 1) in general, while the normalised central moment is better for other combinations of local features. Among them, the best result is obtained when HIGO is incorporated with DCT followed by using the normalised central moment as the second pooling technique. One reason is that the moment about the mean provides clearer information about the distribution’s shape.

We further evaluate the moments and pooling techniques on another task namely medical image analysis and reach consistent observations. More details can be found in Section 4.3 and Table 4.

To compare the efficiency of feature extraction for local features, in Table 2 we list the total running time of calculating local features for the 4608 texture images on TIPS2. More specifically, we implement the integral histogram technique for HIGO, adopt the fast Fourier Transform based implementations for DCT, and use the integral histogram as well as the cell function based implementation for L²ECM in the platform of MATLAB R2013b. The experiments are carried out on a desktop with an Intel Core ‘i5-4570’ CPU @3.2GHz and 8G physical memory. It can be easily observed from Table 2 that the differences in the total running time of R7, HIGO, and DCT are marginal. However, the one for L²ECM is much higher, say over 100 times more than the others. Obviously the L²ECM is in much lower speed than HIGO and 2D-DCT. The main reason lies in the time-consuming pixel-wise matrix logarithm operation. The superiority in efficiency of HIGO and 2D-DCT is definitely shown by the experimental results.

As encouraged by the comparative experimental results, we use the second-order standardized average pooling (*2Standmap*) as an example, and apply it to four typical computer vision tasks.

4.2. Texture Classification

To compare with other widely used approaches using the nearest neighbor classifier, we fix a 20- d feature vectors for 2Standmap as follows: $R, G, B, |I_x|, |I_y|, |I_{xx}|, |I_{yy}|$, the 8-bin HIGO, and the first 5 non-DC coefficients from 2D-DCT, the same with the one illustrated in the rightmost column in Table 1.

Besides the TIPS dataset, we also use KTH-TIPS (abbreviated as TIPS) [30], which contains images of 10 materials, giving a total of 81 images per material. We follow the experimental set-up by Zhang *et al.* [31], in which 40 images of each material are randomly selected for training while the 41 remaining ones are for testing.

On the TIPS dataset, different texture classification approaches are used for comparative study. We compare with the widely used approaches evaluated in the comparative study [31, 7]. We also compare with the (improved) Fisher Vector (IFV) [23, 32, 33], which is a state-of-the-art approach aggregating local descriptors into a global image representation, and the Deep Convolutional Neural Network (DCNN) [34, 35, 36], which has produced some stellar results on a wide range of computer vision tasks in recent years. We use the VLFeat toolbox by [37] for IFV and set the whitened local descriptor for IFV to 60 and 80 respectively and the number of Gaussian in the Mixture 10. These two implementation are denoted for short by ‘IFV (60)’ and ‘IFV (80)’. For DCNN, we use the MatConvNet toolbox [38]. We calculate the deep representation based on the pre-trained ‘Alex Net’ [35] model by removing the softmax layer and also the final full connected layer. Other results of these approaches are either from the original publications or from the comparative evaluation [31, 7]. The accuracy and the dimension of the descriptor are listed in Table 3.

Table 2: Total running time of local feature extractions for the 4608 texture images on TIPS2. ‘R5’ refers to the five regular features used in Section 4.1.

Local features	R7	HIGO	DCT	L ² ECM
Total running time (second)	144	190	126	19,768

According to our best knowledge, the current best result on the TIPS dataset is 99.29% achieved by the approach of Sorted Random Projections (SRP) [39]. However this result can only be achieved by using support vector machine. When using the NN classifier as in this paper, the accuracy becomes 97.71%. It is observed the 2Standmap achieves an accuracy of 98.50%, which is with a marginal distance to the DCNN, equivalent to the Basic Image Features-columns [40], and outperforms the other approaches under consideration including the IFV.

We also evaluate our approach on the TIPS2 database. The 2Standmap achieves the third best accuracy of 73.38% among those methods compared in Table 3, comparable to the IFV. Not surprisingly, the DCNN representation obtains the best result with an accuracy of 78.39%. However, the dimension of DCNN is 4096, nearly 20 times higher than ours.

In [27], the L^2 ECM obtains an accuracy close to 98% on TIPS, using the k -NN ($k = 5$) which is perform in the level of one quarter of the image rather than the holistic texture image. We implemented the L^2 ECM with the same patch support and the maximum number of raw features. In the framework of NN as in our experiments, their performance drops to 96% and 68% respectively. Moreover, as pointed out in the previous experiments, the feature extraction time of the L^2 ECM are rather inferior.

In addition, compared with the CovM₅ [16], the 2Standmap descriptor obtains a substantial increase in accuracy by about 17%. This comparison suggests the necessary of efficiently capturing the correlations of one local descriptor and other raw features, which is reached by the proposed approach.

Since these two datasets are challenging, especially the TIPS2 dataset, the promising performance proves the effectiveness of our 2Standmap representation.

It is worth noting that no learning stage is required to form the proposed 2Standmap descriptor. In contrast, most of the state-of-the-art approaches such as DCNN, IFV, LEP, and SRP require an extra learning stage to form the texture description from the training images.

4.3. Medical Image analysis

To show that the 2Standmap descriptor is general for appearance based image representation, we apply it to the task of medical image analysis, which is highly relevant to the task of texture classification. The 2D-HeLa dataset [41], which is a benchmark for descriptors and classifiers in biomedical diagnosis, is used for evaluation.

The aim of 2D-HeLa is to automatically identify sub-cellular organelles from fluorescence microscope images. It contains ten classes of HeLa cells which are stained with various organelle-specific fluorescent dyes. Each class contains various (from 70 to 98) images, resulting in 862 images in total. All images on 2D-Hela are with a resolution of 382×382 . Large variations in appearance can easily be observed because of the non-rigid action of HeLa cells.

For fair comparison, we utilize the contrast-limited adaptive histogram equalization and adopt the Radial basis function kernel SVM as used in [42, 43]. The parameters C and γ are determined using the grid search strategy respectively.

Table 3: Comparison results on TIPS and TIPS2.

	TIPS	TIPS2	Dim.
VZ-MR8	93.50	62.35	440
VZ-joint	92.4	61.93	440
uniform LBP	93.17	61.00	857
Local Ternary Pattern	93.17	65.27	1714
completed LBP	95.13	66.58	2200
Weber local descriptor	91.08	63.97	2880
Local Energy Pattern	97.56	71.53	2916
Basic Image Features-columns	98.50	71.56	1296
CovM ₅	92.82	57.95	15
IFV (60)	98.25	73.71	1200
IFV (80)	98.42	74.14	1600
DCNN	98.93	78.39	4096
<i>2Standmap</i>	<i>98.50</i>	<i>73.38</i>	<i>210</i>

The feature set for 2Standmap is the same with the 20- d one in Section 4.2. All experimental results are obtained using 5-fold cross validation. Images of each class are randomly split into five parts, with four of fifth for training and the remaining one fifth for testing.

In Table 4, we evaluate the moments and pooling techniques and report the average accuracy over 5-fold cross validation. We reach consistent observations as we made in Section 4.1. Firstly, the second order average pooling outperforms the max pooling. Secondly, the normalised moments achieve the best. Thirdly, the L²ECM does not bring any improvement in accuracy compared with using HIGO-DCT only such that the combination of the three does not outperforms the combination of HIGO-DCT.

We compare our Standmap with the state of the art and summarize the results in Table 5. We conduct the experiments for CovM₅, Fisher Vector and Deep CNN and the results of other methods are from corresponding papers. It can be observed that the proposed approach achieves the highest classification accuracy of 95.93% among all approaches under comparison. What is surprising is that the accuracy of the DCNN representation is inferior. One reason is that the fluorescence microscope images on 2D HeLa are significantly different from those on ImageNet challenge. In spite that fine-tuning is now a popular way of adjusting the learnt deep representation from a source dataset to better suit the target dataset, the small scale of 2D HeLa data constraint the fine-tuning. In contrast, our Standmap is learning free and thus provides an economic way of finding sufficient informative features to handle the large inter-class variations on this small-scale dataset.

4.4. Pain Intensity Detection

The UNBC-MacMaster Shoulder Pain Expression Archive database [46] was used. It contains 48,000 labelled frames of spontaneous facial expressions caused by genuine pain from 200 videos across 25 subjects. Each frame was coded according to FACS. Eleven action units (AUs), namely AU4, AU6, AU7, AU9, AU10, AU12, AU20, AU25, AU26, AU27, and AU43, which are potentially

Table 4: Evaluation of different moments and pooling techniques on 2D HeLa database.

Pooling	$2AveP$				$2MaxP$ ($\alpha = 0.25$)			
Moments	nc	nr	uc	ur	nc	nr	uc	ur
HIGO-DCT(20)	95.9	95.9	95.0	94.9	82.2	80.1	82.9	77.2
ALL(33)	93.4	94.0	93.1	92.9	80.9	80.4	82.9	77.2
Pooling	$2MaxP$ ($\alpha = 0.50$)				$2MaxP$ ($\alpha = 0.75$)			
Moments	nc	nr	uc	ur	nc	nr	uc	ur
HIGO-DCT(20)	80.5	79.0	83.0	77.3	78.7	77.2	78.7	76.0
ALL(33)	80.9	79.2	80.3	77.3	78.6	77.7	80.5	77.1

Table 5: Performance comparison among approaches on 2D HeLa database.

Approaches	Accuracy (%)
LBP ^{ri} [42]	82.3
Local Ternary Pattern ^{ri} [42]	91.4
Local Phase Quantization [42]	79.7
EQP [42]	87.0
ENS [42]	90.0
Dominant LBP [43]	90.0
Dominant Local Ternary Pattern [43]	92.2
multiResolution NN [44]	95.3
disCLBP [45]	95.4
CovM ₅	87.2
Fisher Vector (60)	90.9
Fisher Vector (80)	94.0
Deep CNN	74.5
<i>2Standmap</i>	95.9

related to pain, were taken into account. AU43 was coded on a two level intensity dimensions, while other AUs were on a five level (A-E). To obtain the intensity of the whole frame, the Prkachin and Solomon pain intensity (PSPI) scale was computed as the sum of intensity of AU4, the maximum intensity of AU6 and AU7, the maximum intensity of AU9 and AU10, and the one of AU10. The PSPI is in a range of $[0, 15]$. Note that the PSPI FACS pain scale is currently the only intensity metric which measures on a frame-by-frame scheme.

For every face image, 66 facial landmarks point are detected by the database publisher [46]. As the preprocessing step, the landmarks are aligned by applying the Procrustes analysis [47]. An aligned image is then divided into patches.

According to the best knowledge of the authors, [48] is the only one in which continuous pain intensity estimation can be carried out in the level of frames. More specifically, Ref. [48] compares three baseline approaches namely the locations of 66 facial landmark points (PTS), DCT, and LBP. The combinations among them are also evaluated. Two methods are used for late fusion, namely the mean fusion which calculates the weighted mean of the responses of the regression function based on one single descriptor directly, and the function using Relevance Vector Regression (RVR).

Ref. [48] suggests to divide the facial images into blocks and employ the region descriptor to represent each block separately. In this paper, each facial image is equally divided into 8 by 8 non-overlapping blocks with a resolution of 16×15 . The logarithms of the three 2Standmap descriptors are computed, vectorized, and summarized into a unique vector. The concatenated vector is L2 normalised. To learn the regression function, we adopt the L2-regularized L2-loss support vector regression by using the efficient Liblinear package [49].

We use the following 11- d feature vector for 2Standmap $[I, x, y, |I_x|, |I_y|, |I_{xx}|, |I_{yy}|]^T$ and 4-bin HIGO. The dimension of the block 2Standmap descriptor after half-vectorization reduces to 66, since there are only 240 pixels in each block.

The comparison between the proposed 2Standmap and the approaches in [48] can be found in Table 6. The Mean Squared Error (MSE) and the Pearson Product-moment Correlation Coefficient (PCC) are used as the quantitative

measure. It can be observed that the 2Standmap beats all approaches that only rely on a single descriptor. 2Standmap also outperforms the combination of three descriptors using RVR. There is only a marginal distance between 2Standmap and the mean fusion of the three descriptors [48]. Unlike the approach of using mean fusion, there is no additional fusion step required in our approach.

Table 6: Pain intensity estimation on the UNBC-MacMaster Shoulder Pain Expression Archive dataset

	MSE	PCC
PTS	2.592	.363
DCT	1.712	.528
LBP	1.812	.483
PTS/DCT/LBP _{mean}	1.373	.547
PTS/DCT/LBP _{RVR}	1.804	.502
<i>2Standmap</i>	<i>1.42</i>	<i>0.55</i>

4.5. Micro-expression Recognition

We apply our 2Standmap descriptor to the task of Micro-Expressions (ME) recognition. Micro-expressions are short, involuntary facial expressions, which usually indicate the true but hidden emotion. We carried out experiments on the recently released Spontaneous MICro-expression database (SMIC) [50], which includes three categories, namely positive, negative, and surprise. According to different capture devices, there are three types of data collected, i.e., high speed (HS), normal visual (VIS) camera data, and near-infrared (NIR) data. 20 subjects are asked to watch 16 short movie clips and their facial responses are recorded. Not all subjects show intense enough micro-expressions when recording. It results in 164, 71, and 71 micro-expression clips for HS, VIS, and NIR data respectively. We follow the experimental setup in [50], which employs the temporal interpolation (TIM) approach [51] to normalize each clip to a fixed frame length through temporal interpolation and reversible projection.

Moreover, leave-one-subject-out cross validation is performed and the average accuracy is reported as the final performance measurement. As in [50], linear Support vector machine was adopted to learn the classifiers.

Because the three categories are defined and annotated at the level of the video sequence, the information extracted is required to cover all frames of one video clip. Consequently micro-expressions recognition on SMIC is considered as a 3D spatial-temporal task.

Micro-expressions recognition is a 3D spatial-temporal task. To achieve satisfying performance, the extracted features are required to characterize the intrinsic dynamics over frames. We extend the successful block based scheme to the volume based one in the 3D case. Accordingly, the volume 2Standmap descriptor is computed across all pixels in one volume. In addition, to make our 2Standmap feasible to 3D applications, inspired by [52], we divide the 3D volume into three orthogonal planes namely the xoy, xot, and yot planes. We use the following feature set for 2Standmap as follows $[|I_{dim1}|, |I_{dim2}|]^T$ and 4-bin HIGO for each plane separately, where $|I_{dim1}|$ and $|I_{dim2}|$ refer to the first-order partial derivatives with respect to two coordinates for each plane. Afterwards, the obtained volume 2Standmaps are half vectorized. We evaluate the 2Standmap in each of the three planes and all the combinations of them. Finally, the one on xot plane achieves the highest accuracy of 57.9%, 61.8%, and 45.0% for the HS, VIS, and NIR data respectively. It leads to a substantial improvement by nearly 10% when compared with the best accuracy of 48.8%, 52.1%, and 38.0% correspondingly in [50] obtained by Local Binary Pattern histograms from Three Orthogonal Planes (LBP-TOP) approach. The superiority of our 2Standmap to the conventional LBP-TOP descriptor is obvious.

5. Conclusion

This paper presents a general second-order pooling framework and explores several analogues of the second-order average and max operations by comprehensively investigating a variety of moments. We show the superiority of the

second-order standardized moment average pooling (Standmap). The Standmap provides a unified approach to capsule both low-level information from raw features and the mid-level visual cues from the local descriptors. It is of low dimension, discriminative, efficient, and learning free. We demonstrate applications of 2Standmap in texture classification, medical image analysis, pain expression recognition, and micro-expression recognition and obtain highly competitive results. In future work, we plan to combine our Standmap with the deep convolutional neural network representation and apply it to other challenging tasks such as object recognition and action recognition.

References

- [1] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (10) (2005) 1615–1630.
- [2] L. Seidenari, G. Serra, A. Bagdanov, A. Del Bimbo, Local pyramidal descriptors for image recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* 31 (5).
- [3] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Trans. Pattern Anal. Mach. Intell.* 24 (7) (2002) 971–987. doi:10.1109/TPAMI.2002.1017623.
- [4] D. Lowe, Distinctive image features from scale-invariant keypoints, *Int. J. Comput. Vision* 60 (2) (2004) 91–110.
- [5] M. Varma, A. Zisserman, A statistical approach to material classification using image patch exemplars, *IEEE Trans. Pattern Anal. Mach. Intell.* 31 (11) (2009) 2032–2047. doi:10.1109/TPAMI.2008.182.
- [6] X. Tan, B. Triggs, Enhanced local texture feature sets for face recognition under difficult lighting conditions, *IEEE Trans. Image Process.* 19 (6) (2010) 1635–1650.

- [7] J. Zhang, J. Liang, H. Zhao, Local energy pattern for texture classification using self-adaptive quantization thresholds, *IEEE Trans. Image Process.* 22 (1) (2013) 31–42.
- [8] Y.-L. Boureau, J. Ponce, Y. LeCun, A theoretical analysis of feature pooling in visual recognition, in: *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 111–118.
- [9] Y.-l. Boureau, Y. L. Cun, et al., Sparse feature learning for deep belief networks, in: *Advances in neural information processing systems*, 2008, pp. 1185–1192.
- [10] J. Yang, K. Yu, Y. Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, IEEE, 2009, pp. 1794–1801.
- [11] J. Yang, K. Yu, T. Huang, Supervised translation-invariant sparse coding, in: *Computer Vision and Pattern Recognition (CVPR)*, 2010 IEEE Conference on, IEEE, 2010, pp. 3517–3524.
- [12] J. Carreira, R. Caseiro, J. Batista, C. Sminchisescu, Free-form region description with second-order pooling, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, preprint (1) (2014) 1–1.
- [13] C. Ionescu, J. Carreira, C. Sminchisescu, Iterated second-order label sensitive pooling for 3d human pose estimation.
- [14] B. Ni, M. Xu, T. Nguyen, M. Wang, C. Lang, Z. Huang, S. Yan, Touch saliency: Characteristics and prediction.
- [15] R. Huang, M. Ye, P. Xu, T. Li, Y. Dou, Learning to pool high-level features for face representation, *The Visual Computer* (2014) 1–13.
- [16] O. Tuzel, F. Porikli, P. Meer, Region covariance: A fast descriptor for detection and classification., in: *Proc. European Conference on Computer Vision*, 2006, pp. 589–600.

- [17] V. Arsigny, P. Fillard, X. Pennec, N. Ayache, Geometric means in a novel vector space structure on symmetric positive-definite matrices, *SIAM J. Matrix Anal. Appl.* 29 (1) (2007) 328–347.
- [18] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, M. Harandi, Kernel methods on the riemannian manifold of symmetric positive definite matrices.
- [19] R. Sivalingam, D. Boley, V. Morellas, N. Papanikolopoulos, Positive definite dictionary learning for region covariances, in: *Proc. ICCV*, 2011.
- [20] O. Tuzel, F. Porikli, P. Meer, Pedestrian detection via classification on riemannian manifolds, *IEEE Trans. Pattern Anal. Mach. Intell.* 30 (10) (2008) 1713–1727. doi:10.1109/TPAMI.2008.75.
- [21] W. Feller, *An introduction to probability theory and its applications*, Vol. 2, John Wiley & Sons, 2008.
- [22] X. Pennec, P. Fillard, N. Ayache, A riemannian framework for tensor computing, *Int. J. Comput. Vision* 66 (2006) 41–66.
- [23] F. Perronnin, J. Sánchez, T. Mensink, Improving the fisher kernel for large-scale image classification, in: *Computer Vision–ECCV 2010*, Springer, 2010, pp. 143–156.
- [24] F. Porikli, Integral histogram: A fast way to extract histograms in cartesian spaces, in: *Proc. CVPR*, Vol. 1, IEEE, 2005, pp. 829–836.
- [25] N. Ahmed, T. Natarajan, K. Rao, Discrete cosine transform, *IEEE Trans. Comput.* 100 (1) (1974) 90–93.
- [26] R. Bracewell, R. Bracewell, *The Fourier transform and its applications*, Vol. 31999, McGraw-Hill New York, 1986.
- [27] P. Li, Q. Wang, Local log-euclidean covariance matrix (L2ECM) for image representation and its applications, in: *Proc. ECCV*, 2012.

- [28] Y. Lui, Advances in matrix manifolds for computer vision., *Image Vision Comput.* 30 (6-7) (2012) 380–388.
- [29] B. Caputo, E. Hayman, P. Mallikarjuna, Class-specific material categorisation, in: *Proc. ICCV*, 2005, pp. 1597–1604.
- [30] B. Caputo, E. Hayman, M. Fritz, J.-O. Eklundh, Classifying materials in the real world, *Image Vision Comput.* 28 (1) (2010) 150–163. doi: 10.1016/j.imavis.2009.05.005.
- [31] J. Zhang, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, *Int. J. Comput. Vision* 73 (2) (2007) 213–238.
- [32] K. Simonyan, O. M. Parkhi, A. Vedaldi, A. Zisserman, Fisher vector faces in the wild, in: *Proc. BMVC*, Vol. 1, 2013, p. 7.
- [33] J. Sánchez, F. Perronnin, T. Mensink, J. Verbeek, Image classification with the fisher vector: Theory and practice, *International Journal of Computer Vision* 105 (3) (2013) 222–245.
- [34] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proceedings of the IEEE* 86 (11) (1998) 2278–2324.
- [35] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [36] K. Chatfield, K. Simonyan, A. Vedaldi, A. Zisserman, Return of the devil in the details: Delving deep into convolutional nets, in: *Proceedings of the British Machine Vision Conference*, 2014.
- [37] A. Vedaldi, B. Fulkerson, Vlfeat: An open and portable library of computer vision algorithms, in: *Proc. of the International Conference on Multimedia*, ACM, 2010, pp. 1469–1472.

- [38] A. Vedaldi, K. Lenc, Matconvnet – convolutional neural networks for matlab, CoRR abs/1412.4564.
- [39] L. Liu, P. Fieguth, D. Clausi, G. Kuang, Sorted random projections for robust rotation-invariant texture classification, *Pattern Recogn.* 45 (6) (2012) 2405–2418. doi:10.1016/j.patcog.2011.10.027.
- [40] M. Crosier, L. D. Griffin, Using basic image features for texture classification, *Int. J. Comput. Vision* 88 (3) (2010) 447–460.
- [41] M. Boland, R. Murphy, A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of hela cells, *BMC Bioinformatics* 17 (12) (2001) 1213–1223.
- [42] L. Nanni, A. Lumini, S. Brahnman, Local binary patterns variants as texture descriptors for medical image analysis, *Artificial intelligence in medicine* 49 (2) (2010) 117–125.
- [43] L. Nanni, S. Brahnman, A. Lumini, Selecting the best performing rotation invariant patterns in local binary/ternary patterns, in: *Proc. Int’l Conf. IP, Comp. Vision, and Pattern Recognition*, 2010.
- [44] A. Chebira, Y. Barbotin, C. Jackson, T. Merryman, G. Srinivasa, R. Murphy, J. Kovačević, A multiresolution approach to automated classification of protein subcellular location images, *BMC Bioinformatics* 8 (1) (2007) 210.
- [45] Y. Guo, G. Zhao, M. Pietikäinen, Discriminative features for texture description, *Pattern Recognition* 45 (10) (2012) 3834–3843.
- [46] P. Lucey, J. F. Cohn, K. M. Prkachin, P. E. Solomon, S. Chew, I. Matthews, Painful monitoring: Automatic pain monitoring using the unbc-mcmaster shoulder pain expression archive database, *Image and Vision Computing* 30 (3) (2012) 197–205.

- [47] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, M. Pantic, A semi-automatic methodology for facial landmark annotation, in: Proc. CVPRW, 2013.
- [48] S. Kaltwang, O. Rudovic, M. Pantic, Continuous pain intensity estimation from facial expressions, in: Advances in Visual Computing, Springer, 2012, pp. 368–377.
- [49] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, C.-J. Lin, Liblinear: A library for large linear classification, The Journal of Machine Learning Research 9 (2008) 1871–1874.
- [50] X. Li, T. Pfister, X. Huang, G. Zhao, M. Pietikäinen, A spontaneous micro-expression database: inducement, collection and baseline, in: Proc. FG, 2013.
- [51] Z. Zhou, G. Zhao, M. Pietikäinen, Towards a practical lipreading system., in: Proc. of CVPR, 2011.
- [52] G. Zhao, M. Pietikäinen, Dynamic texture recognition using local binary patterns with an application to facial expressions, IEEE Trans. Pattern Anal. Mach. Intell. 29 (6) (2007) 915 –928. doi:10.1109/TPAMI.2007.1110.