

IMPERIAL COLLEGE LONDON

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

Multimodal image registration and mosaicking of artworks: an approach based on mutual information

Author:

Maria Eugenia Villafane

Supervisors:

Prof. Pier Luigi Dragotti

Dr. Catherine Higgitt

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Electrical and Electronic Engineering,
Communications and Signal Processing Group

October 9, 2024

Statement of Originality

I declare that this thesis is the result of my own work. Information and ideas derived from the work of others has been acknowledged in the text and a list of references is given in the bibliography. The material of this thesis has not been and will not be submitted for another degree at any other university or institution.

Copyright Declaration

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial-No Derivatives 4.0 International Licence (CC BY-NC-ND 4.0). Under this licence, you may copy and redistribute the material in any medium or format on the condition that you credit the author, do not use it for commercial purposes and do not distribute modified versions of the work.

When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law.

Acknowledgements

I would like to thank:

- The Arts and Humanities Research Council's Collaborative Doctoral Partnership consortium (AHRC - CDP) for kindly sponsoring my PhD studies and therefore funding this research.
- My supervisor from Imperial College London, Professor Pier Luigi Dragotti.
- My supervisor from The National Gallery London, Doctor Catherine Higgitt.
- The National Gallery London and the staff at the Scientific Department for making available the photographic and technical images of painted artworks that gave place to the research presented in this thesis.
- Nathan Daly, Marta Melchiorre Di Crescenzo and Joseph Padfield for sharing with me invaluable insights on image modalities, and on more general image and signal processing techniques involved in art investigation.
- Christine Kimbriel from The Hamilton Kerr Institute, University of Cambridge, for sharing her insights on image processing techniques used in cultural heritage.
- The Thomson Collection at the Art Gallery of Ontario for making available the images of the *Portrait of Jane Boughton, née Coningsby* (AGOID.29371, The Thomson Collection at the Art Gallery of Ontario), *Unknown Man, formerly called Edward, Baron Herbert of Cherbury* (AGOID.29377, The Thomson Collection at the Art Gallery of Ontario), and other artworks of this collection.
- John K Delaney, Senior Imaging Scientist, Scientific Research, National Gallery of Art in Washington DC, for making copies of the *NGA-GWU-HSI-registration algorithm* software available to the National Gallery in London, since this made it possible to run experiments with it for comparison of registration results obtained with our method as presented in Chapter 3.
- Rob Erdmann from Rijksmuseum and University of Amsterdam, for the enlightening conversations at the beginning of my research and sharing with me his insights on his research in image processing.

- My colleagues Herman Verinaz Jadan, Su Yan, Junjie Huang, Marek Hilton, Roxana Alexandru, Vincent Leung, Renke Wang and Siying Liu, Hongjian Xiao and Michalis Lazarou for being so generously supportive by sharing their time, knowledge and insights about the field of signal processing.
- My family in Buenos Aires and my family in London: Gaetano, Nero y Nova.

Abstract

We present a novel approach for registration of multimodal technical images associated with painted artworks. We developed both an automated method for area-based registration and mosaicking based on mutual information, and a graphical user interface (GUI) that allows heritage end-users to implement our method from the browser of any computer, without the need of interfacing with the code directly.

Our method was initially developed for registering element distribution maps resulting from macro X-ray fluorescence (MA-XRF) scanning of painted artworks, which take the form of a layered image stack. This stack is treated as the moving image for registration to the target fixed image - which is usually, but not limited to, the visible image of the same artwork. Our method can register multiple moving images simultaneously (each one arranged as a layered image stack, and each covering part of the fixed image), as well as it can be applied for registering various other image modalities.

Operating in consecutive stages, the method registers one moving image at a time, progressively evaluating a range of transformations that are applied to each image in the stack. Finally, a unique, optimised transformation that provides the highest average mutual information across all images in the stack is identified with consensus. This transformation is then applied to the entire moving image to obtain the best alignment between the moving and fixed images when overlapped.

Our browser-based GUI facilitates setting up the images to align, as well as storing the initial settings assigned by the user to each set of moving images. The user can then execute, automatedly, the registration of multiple moving images in the same interface - immediately after setting up initial parameters or at a later session. The results are accessible from the GUI and can be saved in the computer of the user.

Contents

1	Introduction	14
1.1	Background and motivation	16
1.1.1	Image modalities	19
1.1.2	Image mosaicking	20
1.2	Conferences and publications	21
1.3	Outline of this thesis	22
2	Background and State of the Art	23
2.1	Introduction	23
2.2	Images retrieved from the MA-XRF scanning of artworks	25
2.2.1	Collection of MA-XRF data	26
2.2.2	Challenges and current methods for the processing of MA-XRF data	27
2.2.3	Experimental details	30
2.2.4	Challenges to overcome for the registration of images obtained from MA-XRF scanning of artworks	31
2.3	Feature-based image registration	33
2.3.1	Feature detection methods	34
2.3.2	Feature matching algorithms	36
2.3.3	Computing transformations	39
2.3.4	Examples of feature-based image registration methods	47
2.4	Area-based image registration	57
2.4.1	Computing transformations: Non-rigid transformations	58
2.4.2	Metrics of alignment - An overview	60
2.4.3	Optimisers	64
2.5	Learning Based Methods	65
2.5.1	Voxelmorph	65
2.5.2	Synthmorph	68
2.5.3	Experiments of image registration using Synthmorph	70
2.6	Discussion	73

2.7	Summary of this chapter	74
3	An automated, area-based image registration method	75
3.1	Introduction	75
3.2	Overview of the method	76
3.3	Implementation of the method	78
3.4	Similarity and affine transformations	82
3.5	Non-rigid transformations	84
3.5.1	B-splines transformations: control-grid point	85
3.5.2	B-splines transformations: number of parameters	86
3.5.3	B-splines transformations: L-BFGS optimiser	86
3.6	Discussion	88
3.7	Summary of this chapter	89
4	Experiments	91
4.1	Registration of images that capture an entire 2D artwork	91
4.1.1	Experimental details	92
4.1.2	Image stack composed of element maps obtained from MA-XRF scanning only	92
4.1.3	Image stack composed of images obtained from MA-XRF scanning plus additional synthetic images	95
4.1.4	Comparison of results: registration with feature-based methods	98
4.2	Registration and mosaicking of images that capture partial areas of a 2D artwork .	100
4.2.1	Registration of image stacks to the visible image of artwork	103
4.2.2	Registration of image stacks to the visible image of artwork with additional information of neighbouring image stacks	103
4.3	Runtime performance	105
4.4	Discussion	107
4.5	Summary of this chapter	107
5	Graphical User Interface (GUI)	109
5.1	Introduction	109
5.1.1	Software and accessibility constraints: client-server architecture	111
5.2	Browser-based <i>client</i> application: interface for selecting images in the browser and setup of initial locations	112
5.2.1	Format of images for registration and limitations on image size	114
5.3	Execution of the area-based registration method	115
5.4	Visualisation of results	117
5.5	Discussion	119
5.6	Summary of this chapter	119

6 Conclusion	120
6.1 Summary of thesis achievements	121
6.2 Future work	122
A User guide: image registration using the first version of the browser-based image registration GUI and running the registration algorithm from a local installation	125
A.1 Assigning initial parameters for image registration	126
A.2 Implementing image registration with initial parameters (i.e. stored in <i>settings</i> file)	130
Bibliography	148

List of Figures

1.1	Introduction: image registration	14
1.2	Element distribution maps: selection of 15 images derived from the MA-XRF scanning of the <i>Portrait of Jane Boughton, née Coningsby</i>	19
1.3	Image mosaicking: Mosaic image of MA-XRF scanning of “ <i>The Death of Actaeon</i> ” (NG6420, The National Gallery, London), elemental map of lead.	21
2.1	Background, introduction: mapping function	24
2.2	Images retrieved from the MA-XRF scanning of artworks: diagram of the acquisition of an MA-XRF dataset from an easel painting	26
2.3	Images retrieved from the MA-XRF scanning of artworks: observed XRF spectrum of a single pixel of the MA-XRF datacube	28
2.4	Images retrieved from the MA-XRF scanning of artworks: representative MA-XRF datacube of an easel painting	29
2.5	MA-XRF scanning of artworks: device set up	31
2.6	MA-XRF scanning of “ <i>The Virgin of the Rocks</i> ” (NG1093, The National Gallery, London): image stack for registration	32
2.7	Feature detection: FAST (Features from Accelerated and Segments Test) - corner detector	35
2.8	Feature detection: Difference-of-Gaussians (DoG)	35
2.9	Feature matching: BRIEF descriptors	37
2.10	Computing transformations: basic motion models	40
2.11	Computing transformations: affine and homography	43
2.12	Computing transformations: resampling of the moving image	46
2.13	Examples of feature-based methods: SIFT	48
2.14	Examples of feature-based methods: ORB - comparison to SIFT	50
2.15	Examples of feature-based methods: Modified ORB - experiments (1)	52
2.16	Examples of feature-based methods: Modified ORB - experiments (2)	53
2.17	Examples of feature-based methods: Modified ORB - experiments (3)	53
2.18	Examples of feature-based methods: Conover et al.	56

2.19	Non-rigid transformations: area of influence of control points $\mathbf{x}_k$	60
2.20	Non-rigid transformations: parameterised control-point grid, area of influence of control points	61
2.21	Mutual Information (MI)	63
2.22	Examples of learning-based methods: Voxelmorph	67
2.23	Examples of learning-based methods: Synthmorph	69
2.24	Learning-based methods: registration exercises with Synthmorph - overview of the model	71
2.25	Learning-based methods: registration exercises with Synthmorph - results	72
3.1	Element distribution maps: selection of 15 images derived from the MA-XRF scanning of the <i>Portrait of Jane Boughton, née Coningsby</i>	77
3.2	Method - Sequence of 4 searches at progressively increased resolution	78
3.3	Method - Selection of best transformation at each search	79
3.4	Method - Sequence of overlays of registered images obtained at each search	80
3.5	Construction of the Search-space: moving image evaluated across across the entire visible image, vs. its evaluation across a "region of interest" within the visible image.	81
3.6	Non-rigid transformations: implementation of a parameterised control-point grid	85
3.7	Registration of MA-XRF scanning of " <i>The Death of Actaeon</i> ": overlays of target visible image with lead map (Pb-L lines)	87
4.1	Registration of MA-XRF scanning. Image stack of 15 derived images: initial setup	93
4.2	Registration of MA-XRF scanning. Image stack of 15 derived images, to visible image: overlays of target visible image with Au-L α map	93
4.3	Registration of MA-XRF scanning of the <i>Portrait of Jane Boughton, née Coningsby</i> : sequence of overlays of registered images obtained at each search	94
4.4	Registration of MA-XRF scanning. Image stack of 15 derived images, vs. same image stack plus synthetic image: initial setup	95
4.5	Registration of MA-XRF scanning. Image stack of 15 derived images, vs. same image stack plus synthetic image: comparison of final alignments. Overlays of target visible image with Au-L α maps.	96
4.6	Registration of MA-XRF scanning. Image stack of 15 derived images, vs. same image stack plus synthetic image (enhanced resolution version): initial setup	97
4.7	Registration of MA-XRF scanning. Image stack of 15 derived images, vs. same image stack plus synthetic image (enhanced resolution version): comparison of final alignments. Overlays of target visible image with Au-L α maps (enhanced resolution version).	98

4.8	Registration of MA-XRF scanning, comparison of results obtained with feature-based method.	99
4.9	Element distribution maps: selection of 12 images derived from the MA-XRF scanning of the “ <i>The Death of Actaeon</i> ”	101
4.10	Registration of MA-XRF scanning of “ <i>The Death of Actaeon</i> ” (NG6420, The National Gallery, London): target visible image and image stacks	102
4.11	Iterative registration of MA-XRF scanning of “ <i>The Death of Actaeon</i> ”: initial registration of image stacks to the visible image	103
4.12	Iterative registration of MA-XRF scanning of “ <i>The Death of Actaeon</i> ”: sequence of registration of neighbouring image stacks to the visible image	104
4.13	Iterative registration of MA-XRF scanning of “ <i>The Death of Actaeon</i> ”: mosaic image of neighbouring image stacks (elemental map of lead) registered to the visible image	105
4.14	Mosaic after registration of MA-XRF scanning of “ <i>The Death of Actaeon</i> ”: comparison sequence of registration techniques	106
4.15	Mosaic after registration of MA-XRF scanning of “ <i>The Death of Actaeon</i> ”: comparison sequence of registration techniques - Detail	106
5.1	GUI: Overview.	110
5.2	GUI: client-server architecture. Server in remote computer	112
5.3	Registration of MA-XRF scanning with GUI. Image stack of 15 derived images, vs. same image stack plus synthetic image: comparison of final alignments. Overlays of target visible image with Pb-L maps.	113
5.4	GUI: <i>settings</i> file	115
5.5	GUI: registration executed by user in local computer.	116
5.6	Registration of MA-XRF scanning with GUI (second version). Image stack of 15 derived images, vs. same image stack plus synthetic image: comparison of final alignments. Overlays of target visible image with Pb-L maps.	117
5.7	GUI: registration results. <i>Transformed</i> moving images as individual padded PNG images with transparent channel to match size of visible fixed image.	118
A.1	User Guide to GUI: landing page of the application.	127
A.2	User Guide to GUI: opening the first image.	127
A.3	User Guide to GUI: opening the moving image.	128
A.4	User Guide to GUI: parameters of the moving image.	128
A.5	User Guide to GUI: modifying the parameters of the images that compose the moving image.	129
A.6	User Guide to GUI: uploading a previously saved “ <i>settings.json</i> ” file.	129
A.7	User Guide to GUI: setting up the virtual environment.	130

A.8	User Guide to GUI: indicating the location of the results folder (1).	131
A.9	User Guide to GUI: indicating the location of the results folder (2).	131
A.10	User Guide to GUI: folder for storing results.	132
A.11	User Guide to GUI: folder for storing results with “ <i>settings.json</i> ” file.	132
A.12	User Guide to GUI: indicating the location of the python file (1).	133
A.13	User Guide to GUI: indicating the location of the python file (2).	133
A.14	User Guide to GUI: commencement of registration using our area-based method.	134
A.15	User Guide to GUI: subfolders for storing intermediate results.	134
A.16	User Guide to GUI: folder with final results.	135
A.17	User Guide to GUI: opening the PNG images resulting from our area-based method.	136
A.18	User Guide to GUI: selecting the images to open as layers for the final mosaicked image.	136
A.19	User Guide to GUI: Final mosaicked image from PNG layers.	137

Chapter 1

Introduction

In the fields of heritage science and art investigation, the study of materials and techniques of artworks is commonly supported by the acquisition of technical images and their comparison to the visible image of the same artwork. In order to ensure a like-for-like comparison, it is common practice to align these technical images (here considered to be the *moving* image ¹) to the visible image of the artwork (here serving as the target *fixed* image ²). This is done by applying a geometrical transformation to the moving image so as to align it to the target fixed image, which remains unaltered throughout the process of alignment. This process is known as *image registration* [1, 3, 4, 5, 6].

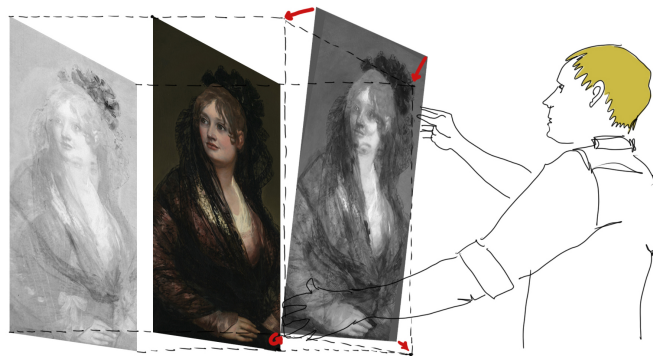


Figure 1.1: Image registration is the process of finding a spatial transformation to determine pixel-to-pixel correspondence from one image to another [1]. This image shows a user "deforming" a greyscale image (representative of the technical *moving* image) so as to align it to the visible image of the artwork (here representing the target *fixed* image). Images after "Doña Isabel de Porcel" (NG1473, The National Gallery, London) after Spring et al. [7].

For the purpose of studying different aspects of an artwork, a range of so-called "technical images" may be obtained by scanning the artwork with a variety of wavelengths of electromagnetic radiation including visible light, infrared radiation and X-rays [8]. These images are categorised by the method and type of sensor utilised for their generation, and the resulting categories are referred

¹ Also referred to as *sensed* image [1, 2].

² Also referred to as *source* image or *reference* [1, 2].

to as *image modalities* [9] (see Section 1.1.1). Thus, we refer to the registration of technical images to the visible image of the same artwork as a *multimodal* image registration, since it entails the registration of images produced by different image modalities. Multimodal image registration is broadly implemented across multiple disciplines in various fields, such as remote sensing, computer vision and medical imaging [1, 2, 10].

As we will discuss in Chapter 2, there is a wide variety of *automated* approaches for conducting image registration. In particular, automated approaches for multimodal image registration are typically conducted with *area-based* methods (see Section 2.4). However, while there is a variety of automated registration techniques being used in art investigation, these commonly rely on methods based on feature detection [11, 12, 13] and are therefore are not ideal for multimodal image registration, since a feature point in one modality may not be present in another modality (*feature-based* methods, see Section 2.3).

In the case that the user decides to conduct registration in a *manual* or semi-automated manner, it is customary that corresponding feature points are identified manually across the images to align, in software packages like Photoshop [14], GIMP [15] or VIPS-NIP2 [16]. Based on the findings, a geometrical transformation is then applied to the moving image for its registration to the visible image of the artwork. This approach to registration may yield good results when the images to register contain common features - in example, matching cracks shown in high-resolution images. However, selecting features by hand can be time-consuming (especially in high-resolution images, as in this case the user may need to inspect the images to align at pixel-level in a visual manner) and the results may depend on the skill of each user. Additionally, some image modalities (such as macro X-ray fluorescence (MA-XRF) scanning) tend to show features that may be different to those apparent in the visible image, as they may relate to different properties of the artwork (see Section 2.2.4). Finally, images of moderate or low resolution (as it is usually the case with images obtained from MA-XRF scanning) may only show fine details such as craquelure with a "blocky" or coarse appearance, resulting in the fact that details as fine as craquelure are usually beyond the working resolution of this type of images (see Section 2.1 and Section 4.1.1 for typical resolution of MA-XRF images). Thus, corresponding feature points may be challenging to identify, posing a problem to conducting multimodal image registration in a manual manner. All these aspects contribute to the importance of having access to automated methods for registration of multimodal technical images within the field of art investigation.

We also note that digital spectroscopic imaging methods have become more accessible within the field of art investigation in recent years. Spectroscopic methods may produce large volumes of data with spatial and spectral domains. This type of data is often referred to as *datacube*³, which can be considered as a stack of images (of size set by the spatial dimensions of the datacube) at different wavelengths (as is the case for hyperspectral imaging) or energies (as is the case for

³ Also referred to as *image-cube* [17].

MA-XRF scanning).

The imaging equipment incorporates a spectrometer that separates up the broad band signal into smaller wavelength ranges [18]. For example, infrared reflectography works over a similar range to that of hyperspectral imaging in the Short-Wave Infrared Range (SWIR), but one captures the full range of radiation in one image, and the other separates it up into multiple individual images [19].

As we discuss in Section 2.2, it is customary to consider the images of each stack to be aligned by construction ⁴, since they are all generated from the same scanning of the optical device.

The conventional approach to address registration of a stack of images is to take one image that may present significant information of the scanned area of the object (such as salient features or well-defined shapes), and register that image to the visible image of the artwork. Once this *representative* image is successfully registered, it is customary to apply the same transformation to all the images belonging to the same stack. In this case, it is important to note that the choice of the *representative* image to register in the first place might have a strong impact on the quality of the registration of the full image stack, as different images of the same stack may allow different best transformations to be identified (this is due to the fact that each individual image may present exclusive features relative to different motifs or pictorial patterns of the visible image - see Section 2.1). For this reason, when addressing the registration of image stacks, it is desirable to consider all the images in the stack so as to identify one unique best transformation to apply to the entire stack with *consensus* among all these images.

1.1 Background and motivation

In this thesis, we present a novel approach to address automated registration of multimodal technical images of artworks. In particular, we present an area-based method developed for addressing, automatedly, the registration of technical images obtained from macro X-ray fluorescence (MA-XRF) scanning of artworks to the visible image of the same artwork.

With the images obtained from the processing of this type of scanning data arranged into an image stack, our method assigns a unique best geometrical transformation that is applied to the entire stack (here the *moving* image) for its alignment to the visible image of the artwork (here the *fixed* image). Using the mutual information of these images as a metric of alignment, this unique best transformation is selected with consensus among all the images of the stack. Mutual information (MI) is frequently used as a metric for registration of images obtained with different modalities, particularly in the medical imaging sector [23, 24, 25], since it overcomes challenges that are common in multimodal registration such as the presence of different features across the images to align (see Section 2.4.2).

⁴ Extensive work on filter-based multi-spectral imaging was conducted by David Saunders, John Cupitt and Haida Liang during their time at the National Gallery, since the original images produced were not absolutely aligned by construction [20, 21, 22]. Note that this work lead to the development of the software package NIP2 [16].

In order to identify a unique best transformation with consensus among all images in the stack, all transformations within a range of parameter values are assessed for the registration of each individual image against the target fixed image. The geometrical transformation that yields the highest average MI across all images in the stack is selected as the best transformation to apply to the entire stack. This process is iterated four times. At the beginning, coarse transformations (with big step sizes and a moderate number of parameters to optimise) are evaluated, and they are progressively refined (reducing the step size and increasing the number of parameters to optimise) as the method proceeds (see Section 3.2 for illustrative diagrams and a more detailed overview of this method). Note that this approach is designed to work with artworks that are nearly flat objects, such as panel and easel paintings. For this reason, similarity and affine transformations allowing only moderate distortions are evaluated in the first three stages (see Section 3.4), and is only in the last stage that a non-rigid registration is evaluated (see Section 3.5). In this way, the moving image (i.e. the image stack) is progressively better aligned to the target fixed image. This hierarchical approach allows the computational time to be reduced whilst gradually improving the registration of the images to align.

It is important to note that the image stack can be composed of one single image, or of multiple images. Our method returns a unique best transformation that can be applied to the entire image stack for its registration to the target fixed image (which is usually -but not limited to- the visible image of the same artwork). This is due to the fact that all images in the stack are assumed to be aligned by construction, since they are all generated from the same capture of the scanning device.

Since our method *identifies* the best transformation to apply to the moving image from a range of candidate transformations, it is important to have images in the stack that show information that can be related to the target fixed image. For this reason, when the image stack is composed of multiple images, it might occur that individual images contain different information from one another. In this case, the moving image can be *enhanced* by adding to it a new *synthetic* image resulting from the combination of images of the original stack, where each of the images to combine show dissimilar information (i.e. different motifs) of the artwork (see Section 4.1.3). Conversely, the target fixed image can be *enhanced* by combining its information with that of a technical image which has already been registered (see Section 4.2.2). This can be particularly helpful to register neighbouring scanned areas and obtain a *mosaic image* from various partial technical images of the artwork to study (see Section 1.1.2).

The area-based method presented in this thesis was developed with a particular application in mind, that is the automated registration of technical images obtained from MA-XRF scanning of painted artworks to the visible image of the same artwork. The research leading to its development was conducted under the Arts and Humanities Research Council’s Collaborative Doctoral Partnership consortium scheme (AHRC - CDP, see <https://www.ahrc-cdp.org/>) as a joint collaboration between the Scientific Department of The National Gallery in London and the Communications

and Signal Processing Group of the Electrical and Electronic Engineering department (CSP - EEE) of Imperial College London. This aspect made it possible for the research to be targeted to clear end users ⁵ who supplied the data and the associated challenges. The team at the Scientific Department of The National Gallery provided a variety of technical images acquired with different modalities, with a particular interest in addressing the registration of images obtained from MA-XRF scanning. For this purpose, they provided raw data resulting from MA-XRF scanning of several painted artworks. As described in Section 2.2.2, this data was processed with the deconvolution methods developed by Yan et al. [26, 27] to obtain the corresponding technical images for their registration to a photographic, high-resolution visible image of the same artwork. The team at the Scientific Department of The National Gallery also provided feedback regarding the development of the algorithms supporting the registration method and the output images, in such ways that the research was conceived as an end-user application from an early stage.

It is also worth noting that the algorithms supporting this area-based method and the registration experiments conducted throughout this research were extensively discussed with the team of the *ARTICT / Art Through the ICT Lens: Big Data Processing Tools to Support the Technical Study, Preservation and Conservation of Old Master Paintings* project - a collaboration between The National Gallery, University College London and Imperial College London, funded by the Engineering and Physical Sciences Research Council (EPSRC - EP/R032785/1, see <https://art-ict.github.io/artict/home.html>). Thus, this thesis is targeted to a dual readership from the field of signal processing within Electrical and Electronic Engineering, and the field of cultural heritage within the Arts and Humanities, aiming to address both sets of audiences in a clear and effective manner.

Lastly, it is important to note that, similarly to other automated methods for image registration, our method needs to be initialised with an initial location of the moving image so that is located as close as possible to its target position within the visible fixed image. This step may require the user to use a different software platform to that of the software producing the registration to prepare the images. With the motivation of creating a streamed-line workflow that allows to unify the assignment of the initial location and the registration of the images to align, we have developed a *graphical user interface* (GUI) that brings together these two tasks (see Sections 4.4 and 5.1).

Throughout the experiments presented in this work, we show the implementation of our method for registration of images derived from MA-XRF scanning of painted artworks to the visible image of the same object. Nonetheless, it is important to note that our method is broadly applicable to the registration of images obtained with other modalities, either serving as moving and/or fixed images.

⁵ We target end users within the cultural heritage sector to benefit from our easily accessible graphical user interface (GUI) for automation of image alignment and inspection of results. At the time of submitting this thesis, it is customary that this type of users may not have access to advanced image processing PCs, or have coding expertise to interface with image processing platforms to automate tasks of this kind. In this way, users from the National Gallery London and other heritage users who may have data but may not have the technical skills or infrastructure to examine this data, can benefit from the GUI to aid their work.

1.1.1 Image modalities

In the fields of heritage science and art investigation, technical images of artworks may be obtained by exposing the object to study to a variety of wavelengths of electromagnetic radiation including visible light, infrared radiation and X-rays [9, 11]. In some cases, the wavelength of electromagnetic radiation of choice may lay outside the visible spectrum and thus permit to collect information of the sub-surface layers of the artwork [8], such as underdrawings or pentimenti - which may not appear in the visible image of the artwork. Each type of exposure defines the *modality* of the images obtained, which document features created with materials to which each modality is sensitive [28]. For example, X-ray images will enhance features with high radio-opacity such as lead white, and infrared reflectography will highlight contrast between features with different infrared-absorbing properties [11]. In the case of scanning an artwork with MA-XRF, the signal at each pixel location can be processed [26] to obtain the distribution maps of the chemical elements detected at that area ⁶ (see Section 2.2.1). The output is a stack of images, where each layer is a 2D matrix representing the quantity map of the corresponding chemical element. An example of the maps is shown in Fig. 1.2.



Figure 1.2: Element distribution maps: selection of 15 images derived from the MA-XRF scanning of the *Portrait of Jane Boughton, née Coningsby* (AGOID.29371, The Thomson Collection at the Art Gallery of Ontario). Note that some of these images are subject to pixel averaging (a, b, d, i, j, m, n, o), or are processed by having another element subtracted (g, j, m).

⁶ This is only one very specific way to process MA-XRF data, please refer to Yan et Al. [26] for a review of methods.

The method described in this thesis has been applied for registration of element map images obtained with MA-XRF scanning, as well as for registration of images obtained by illuminating an artwork with ultraviolet light, visible light at a raking angle, infrared reflectograms and images obtained with X-radiography.

1.1.2 Image mosaicking

Image mosaicking (also referred to as *mosaicing*) is the process of forming a composite image of an object by means of registering a set of smaller images that capture partial, contiguous details of the same object [29]. This may be the situation for large objects that cannot be portrayed in a single capture of an optical device, and in turn, the object may be portrayed by multiple consecutive partial images that need to be combined to recreate the full extent of the original entire object.

Note that it is common practice to allow for overlapping between these consecutive individual images, so as to provide some guidance for inferring the exact location of one partial image relative to the next one ⁷. There are essentially two approaches to address image mosaicking:

- The user can align any two consecutive partial images by selecting feature points within the corresponding overlapping regions. By progressively aligning one partial image to the consecutive one, an overall composite image of the entire object can be created. The resulting composite image is then registered to the visible image of the object. The challenge with this approach is the potential propagation of errors - since distortions may be introduced into the partial images that do not relate to the visible image of the object, making registration to the visible image (which is normally the aim of creating a composite technical image of the object to study) challenging.
- The user can register, individually, each partial image to its relevant location within the target image. Once all partial images have been registered, they may be merged together to create a composite image of the full extent of the object.

Note that the first approach entails the user to select the feature points within the overlapping areas of the partial images to align. This is one of the main uses of the semi-automated method NIP2 [16], especially for joining consecutive images obtained with infrared reflectograms together. Note that with early infrared reflectography, the infrared reflectogram of a painting would be created by repeated image capture (photographic) of the image from a vidicon system displayed on a video screen [30, 17]. In an analogous manner, X-ray images were assembled by printing out the individual X-ray film plates and combining them together on a board in an overlapping manner - or by mosaicking the individual digital scans of these X-ray film plates.

⁷ For the MA-XRF scanning of large areas of artworks, it is a common practice at the National Gallery to allow for an overlap of circa 50 mm between any two consecutive areas to scan. Note this is a fixed overlap rather than a percentage of the total area of the artwork.

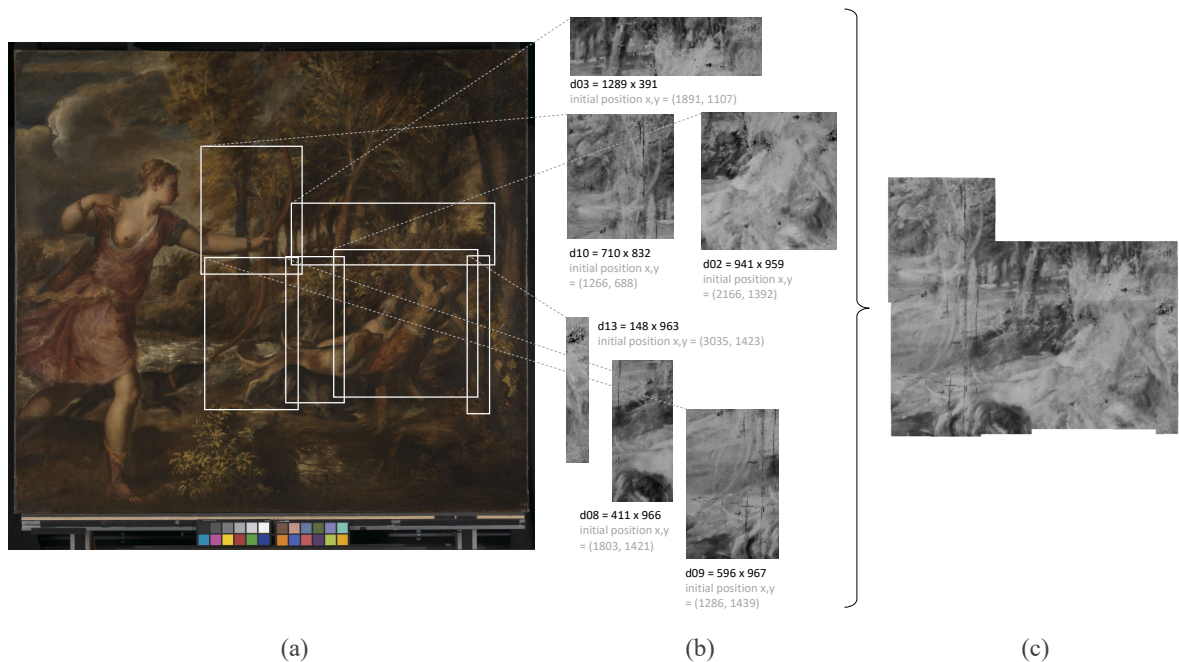


Figure 1.3: Mosaicking of element maps obtained from datacube derived from MA-XRF scanning of “*The Death of Actaeon*” (NG6420, The National Gallery, London). Part (a) shows the visible image of the artwork (here serving as the target-fixed image). Part (b) shows the images of lead maps (Pb-L lines) of scanned areas d02, d03, d08, d09, d10 and d13 (here serving as moving images for registration to the target image, and mosaic), with indication of their approximate location within the visible image. Part (c) shows the resulting *mosaic* image. This case study is described in detail in Section 4.2.

Our method enables registration and mosaicking of multiple image stacks, where each stack is treated as an individual moving image for its registration to its corresponding region within the same target fixed image. This is achieved through a sequence of progressively refined registrations assessed by the corresponding increase of the mutual information of the images to align. This approach is described in detail in Section 4.2. Note that, although our method was originally developed for the registration of technical images obtained from MA-XRF scanning to the visible image of the same artwork, it can be broadly implemented for the registration of other image modalities, such as infrared reflectography, ultraviolet-induced visible fluorescence imaging and raking light photography, and thus also for image mosaicking and creation of image composites.

1.2 Conferences and publications

The work in this thesis has led to the following publications:

- Computational approaches for technical imaging in cultural heritage - 7th Image Processing for Art Investigation (IP4AI) meeting (Online, 27-04-2022). Peer-reviewed conference (online), poster: *Multimodal Image Registration of Old Masters Paintings*.

- TECHNART 2023 - International conference on analytical techniques in art and cultural heritage (Lisbon, 08-05-2023). Peer-reviewed conference (in person), presentation no. 1082 : *Registration of multimodal images of artworks: an approach based on mutual information.*
- SPIE Optical Metrology 103: Optics for Arts, Architecture, and Archaeology (O3A) IX (Munich, 26-06-2023). Peer-reviewed conference (in person), presentation and publication in proceedings: Paper 12620-8. *Multimodal image registration and mosaicking of artworks: an approach based on mutual information.*
- X-Ray Fluorescence imaging (MA-XRF) and Reflectance Imaging Spectroscopy (RIS) 2024 meeting (Washington DC, 04-06-2024). Peer-reviewed conference (in person), presentation: *GUI - Graphical user interface for area-based registration of MA-XRF images.*

1.3 Outline of this thesis

The contents of this thesis are organised as follows:

- Chapter 2. This chapter starts by presenting an overview of the current methods utilised for processing the data that is obtained from Macro X-ray fluorescence scanning of artworks, which is the modality we focus on throughout the work presented in this thesis: from the acquisition of the raw data, to the estimation of the images that are representative of the corresponding elemental distribution maps. This is followed by a general review of the field of image registration and a closer look into the state-of-the-art techniques of the registration methods that are at the basis of our work. We conclude this chapter by presenting experiments conducted with these methods and the insights that informed our technical decisions and helped shaping our method, which is presented in the following three chapters.
- Chapter 3. In this chapter we describe in detail our automated, area-based image registration method.
- Chapter 4. In this chapter we present experimental results using the area-based method presented in Chapter 3 on registration and mosaicking of images obtained from MA-XRF scanning of painted artworks, as well as a comparison of results obtained with the implementation of other image registration methods developed for the cultural heritage sector.
- Chapter 5. In this chapter we present the graphical user interface (GUI) that allows the user to implement our image registration method by means of an intuitive interface that is accessible from the browser of any computer connected to the internet, without the need of interfacing with the code directly. A user guide to our GUI is presented in the Appendix A.
- Chapter 6. In this chapter we present a summary of the contributions of the work presented throughout this thesis, and propose potential avenues of further work.

Chapter 2

Background and State of the Art

This chapter covers the core topics that are at the basis of the automated area-based method for image registration of multimodal technical images of artworks - the main subject of this thesis. In particular, this chapter addresses the process of retrieving images from the datacube derived from the MA-XRF scanning of an artwork, which is the type of images we have focused on along the development of our image registration algorithm. Here, we present the challenges associated with the registration of such images to the visible image of the same artwork. This is followed by a review of feature-based and area-based image registration methods, and their use in the field of heritage science and art investigation, as well as in the medical imaging sector. Within the latter group of methods, this chapter covers the basic principles of our metric of choice, that is the mutual information of the images to align. Finally, a brief review of state-of-the-arts learning-based methods is presented, together with a few experiments conducted with this type of approach for the registration of technical images of artworks. The chapter concludes with the insights that, gained from this implementation, helped shaping the area-based image registration method that is presented in Chapter 3.

2.1 Introduction

As mentioned in Chapter 1, image registration is widely used in various fields, and often constitutes the initial step prior to the study of images obtained with different sensing techniques.

Two images are involved in the registration process. One image, the *moving* image $I_M(\mathbf{x})$, is deformed to fit the other image, the *fixed* image $I_F(\mathbf{x})$. Both fixed and moving image are of dimension d and are each defined on their own spatial domain: $\Omega_F \subset \mathbb{R}_d$ and $\Omega_M \subset \mathbb{R}_d$, respectively. Registration is the problem of finding a displacement $\mathbf{u}(\mathbf{x})$ that makes $I_M(\mathbf{x} + \mathbf{u}(\mathbf{x}))$ spatially aligned to $I_F(\mathbf{x})$ [31]. This is equivalent to say that registration is the problem of finding a transformation $\mathbf{T}(\mathbf{x}) = \mathbf{x} + \mathbf{u}(\mathbf{x})$ that makes $I_M \mathbf{T}(\mathbf{x})$ spatially aligned to $I_F(\mathbf{x})$ (Fig 2.1).

To ensure that all the pixels of the moving image will be mapped to the fixed image domain in

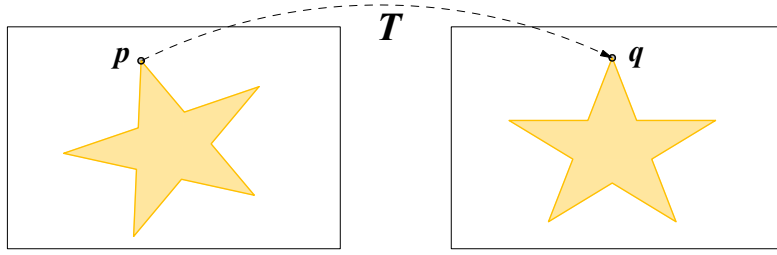


Figure 2.1: Image registration is the process of finding a spatial transformation to determine point-to-point correspondence from one image to another [1]. To the left is the fixed image, and to the right the moving image, where the transformation T is defined as a mapping from the fixed image to the moving image. This function maps pixel p in the fixed image domain to pixel q in the moving image domain. Image modified from Klein and Staring [31].

a satisfactory manner, the transformation T is defined as a mapping from the fixed image to the moving image, i.e $T: \Omega_F \in \mathbb{R}^d \rightarrow \Omega_M \in \mathbb{R}^d$. We refer to this as *resampling* (see 10). Once the geometric transformation has been applied and in order to evaluate the quality of the alignment, a similarity metric S can be implemented. Examples of similarity metrics are normalised cross-correlation and the mutual information of the images to align (see 2.4.2).

The methods to perform image registration are mainly divided into two groups: methods that rely on identifying and matching corresponding features across the images to register (*feature-based* methods, see Section 2.3), and methods that rely on matching regions of equivalent intensities within the images to register [2] (*area-based* methods, see Section 2.4). More recently, methods based on deep learning have started to emerge [32, 33]. Following the above-mentioned "strategy" of either being feature-based or area-based, learning-based methods focus on learning either corresponding matching features across the images to align, including a deep learning component as part of a larger algorithm [32, 34] or directly learning, from the intensities of both fixed and moving images, the deformation vector field to be applied to the latter for its registration to the fixed target image [35, 36, 37, 38] (see Section 2.5).

Feature-based methods offer a variety of techniques to undertake the tasks of detecting salient details, extracting them as feature points and creating corresponding matches across the images to align, so as to identify a transformation to apply to the moving images for registration to the target fixed image. Feature-based methods are used in images that contain distinctive objects, such as is the case of applications in remote sensing and computer vision. Most generally, these techniques yield good registration results when seeking registration of images of similar level of intensity values that are obtained with the same type of detector but may have been captured at different times or from different viewpoints, resulting in these images to be obtained at different resolutions or lighting conditions. However, in the medical imaging sector, technical images tend to lack such defined details [2]. Moreover, it is common to require registration of images obtained from different modalities (i.e. images obtained with different detectors or optical techniques), posing

the challenge of dissimilar pixel intensities across the images to align. Thus in this field, image registration is usually conducted with area-based methods.

This is also the case in the field of art investigation: multimodal image registration is fundamental for the study of technical images (that may serve as the *moving* image) against the visible image of the same artwork (usually a single-capture, high resolution photographic image of the full-visible artwork that serves as the *fixed* image). However in the heritage science field, while it is possible in some cases to find feature points to guide the registration of multimodal images (either manually, as it is customary in this field, by using software like photoshop or semi-automated methods like NIP2 [16], or by means of automated approaches [39, 11, 12]) it is also common that technical images tend to lack defined details that may be consistently visible across the various modalities - such is the case of images derived from modalities such as Macro X-ray fluorescence (MA-XRF).

In some circumstances, the lack of distinctive features in medical images can be overcome by making use of minute markers to act as features that may be apparent throughout the modalities of the images to align. An example of this may be introducing fiducial markers in the region to register, or the interactive selection done by an expert, so as to conduct image registration by matching these "bespoke" features. Some image registration methods in the art investigation field implement a process for selecting intrinsic features of the artwork like craqueleure, which may appear across the different image modalities to align when images are captured at a resolution that is high enough to portray this type of fine details [11, 40, 41]. However, this is not always the case for images obtained with MA-XRF, where the images obtained may be of a lower resolution¹ than that of the photographic, visible image of the artwork, and thus may not be able to resolve fine cracks that may be apparent in the visible image. Moreover, images from different modalities may show different sub-surface features than those apparent in the target visible, creating another challenge of implementation of feature-based methods for image registration of multimodal images.

2.2 Images retrieved from the MA-XRF scanning of artworks

As mentioned in Chapter 1, we focus on the development of an image registration method that could operate with a stack of technical images so as to align the entire stack to a target fixed image of the same art work, simultaneously and with consensus among the images of the stack - where it is assumed that the constituent images of the stack are aligned by construction. In particular, we focus on the registration of the element distribution maps obtained from MA-XRF scanning of artworks to a visible image of the same art work. This scanning technique can be used for non-invasive elemental analysis of paintings. Since different pigments used in the painting

¹ images obtained with MA-XRF tend to be of a much lower resolution than images obtained with other modalities, due to the long time of capture needed to retrieve the signal at each pixel of the image. Typical resolution of MA-XRF scanning of standard size paintings is about 500 - 1000 μm per pixel. Note that the technical images obtained at this resolution would not show craquelure, however with high resolution and prominent cracks it is possible to see these type of fine details.

may contain different elements, producing the distribution maps of the elements can give further information about an artist’s palette, improve the understanding of their painting techniques, for example by revealing how paint layers have been built up, and even reveal sub-surface features of the painting [42, 43, 44]. These elemental distribution maps are the technical images that we endeavor to align to the visible image of the artwork.

In order to understand the characteristics of working with this type of technical images, this section is structured as follows: a description of MA-XRF data and its collection is presented, followed by a description of the challenges around the processing of this data and current processing methods. Finally, we present the details of the data used in the experiments of image registration presented in this research (see. Section 4).

2.2.1 Collection of MA-XRF data

A diagram of the acquisition of a MA-XRF dataset from an easel painting is shown in Fig. 2.2(a). Here, each spot of the area of the artwork to scan is illuminated by an X-ray source with a primary X-ray beam. This may lead to electronic transitions and emission of X-ray photons from the atoms of the chemical elements that may form part of the pigments used in the painting [26]. Each chemical element has a unique atomic structure and atomic energy levels. For each element, transitions between different electronic shells (or sub-shells) release photons with a unique set of energies, called characteristic X-rays [45]. The characteristic X-rays of an element are generated in series (e.g. K-series, L-series, M-series, etc.) and 11 dominant characteristic X-rays are considered, which are then combined into groups for more insightful visualisation results when showing their distribution maps. These groups are called *emission line groups*.

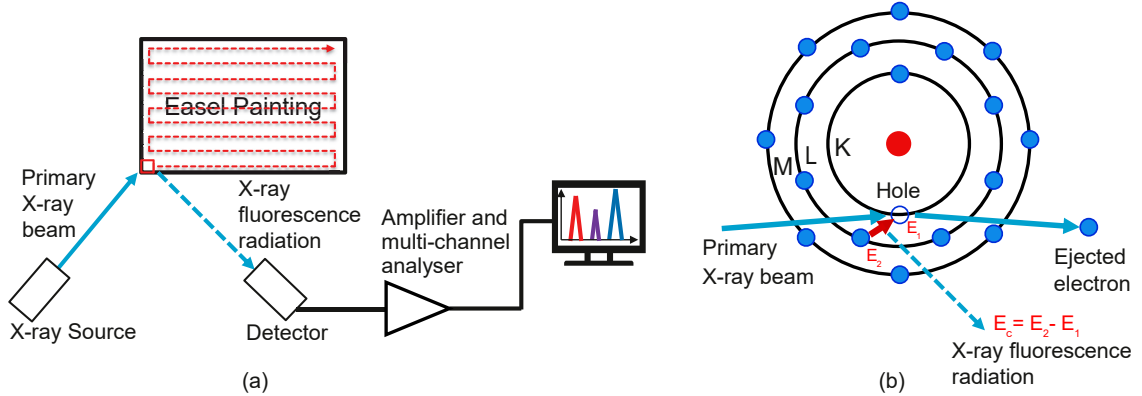


Figure 2.2: Part (a): Diagram of the acquisition of a MA-XRF dataset from an easel painting. Part (b): X-ray fluorescence radiation caused by atom ionisation. Image (a) after *When de Prony Met Leonardo: An Automatic Algorithm for Chemical Element Extraction From Macro X-Ray Fluorescence Data* [26], and image (b) modified from Su Yan’s PhD Thesis, Imperial College, 2022.

Yan et al. [26] efficiently summarise the working principles of the sensor and the structure of the collected data: "The energy of the released photons (E_c) is equal to the energy difference

between the two shells where the electronic transition occurs (Fig. 2.2(b)), which is characteristic of the atom of a chemical element. The detector, with its amplifier and multi-channel analyser, then collects and counts the released X-ray photons with respect to their energy levels, resulting in a pixel of data in the form of a spectrum [46, 47]. Thus, the data collected describes the presence of chemical elements distributed along their corresponding emission line groups. For each point of the area of the artwork that is scanned, a pixel of spectral data is collected in this manner. The resulting MA-XRF dataset is a three-dimensional datacube, with two spatial and one energy dimensions, as shown in Fig. 2.4(a). The range of the energy dimension is from 0 to 40 kilo-electron-volts (keV) and is divided into 4096 energy channels.

2.2.2 Challenges and current methods for the processing of MA-XRF data

With this modality, the data collected at each pixel is the spectral response for each characteristic element X-ray (also called an *element line*). Due to the effect of the detector, this spectral response is in the form of a narrow Gaussian-shaped peak, covering a range of energies. Yan et al. [26] state that the data collected at each scan (the observed XRF spectrum) "can be regarded as a combination of several Gaussian peaks", as shown in Fig. 2.3. However, a problem occurs when multiple peaks of characteristic X-rays are within close proximity, since they may overlap with one another. The process of identifying the individual peaks and the elements that may correspond to the presence of these signals within the XRF spectrum is referred to as XRF spectrum *deconvolution*. Since pigments in artworks can be used in complex mixtures and applied in layers, multiple elements may be excited and thus detected at a single spot (pixel) of the scanning. For these reasons, the data collected (i.e. the spectrum) may be challenging to deconvolve [26].

Yan et al. [26] succinctly present the methods developed in recent years for the processing of MA-XRF data: "Many approaches have been proposed for XRF spectrum deconvolution, based on the use of linear or nonlinear regressions or Bayesian methods to fit the XRF spectrum with Gaussian functions [46], of which those most relevant to MA-XRF scanning are described here. Van Espen et al. [48, 49, 50] created 'AXIL' (Analysis of X-ray spectra by Iterative least Squares), which fits the element peaks with Gaussian functions and the background with a polynomial using iterative least-squares. Developing from this, Solé et al. [51] created an open source software named 'PyMca', which similarly fits the XRF spectrum with Gaussian functions using non-linear least-squares but also takes specific characteristic lines as well as layered samples into consideration. Alfeld and Janssens [52] developed 'Datamuncher' in Interactive Data Language (IDL) for fast processing of the huge datasets typically produced with MA-XRF scanning and to produce elemental distribution maps (from one or more datasets), by making use of the fitting with AXIL or PyMca and Dynamic Analysis (DA) [53, 54]. These tools were used by the original developers of the now commercially available M6 JETSTREAM MA-XRF scanning instrument produced

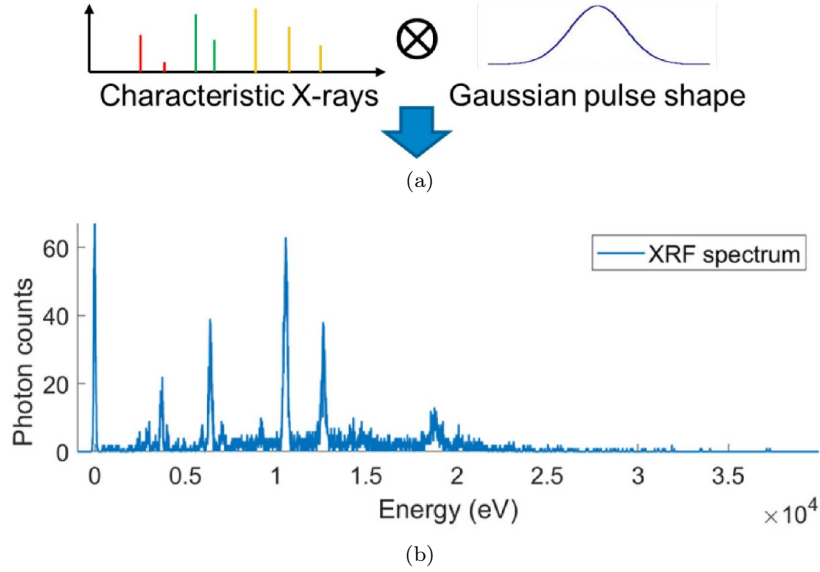


Figure 2.3: Construction of the signals to analyse: (a) Characteristic X-Rays denoting the position of the energy of each element and Gaussian peak. (b) The observed XRF spectrum of a single pixel of the MA-XRF datacube (here collected using a Bruker M6 JETSTREAM instrument) consists of a series of overlapping and broadened characteristic element X-ray lines. Images after *When de Prony Met Leonardo: An Automatic Algorithm for Chemical Element Extraction From Macro X-Ray Fluorescence Data* [26].

by Bruker Corporation, the instrument most widely adopted for use by the heritage sector [55]. Conover proposed a method that uses non-linear regression to fit the XRF spectrum with Gaussian functions centred at the element characteristic energies [56]. However, this method relies on XRF spectra with high signal-to-noise ratios and achieving this requires long acquisition time. Most instrument producers also provide proprietary deconvolution and quantification tools in the software for their instruments, including Bruker Corporation for their M6 JETSTREAM, although these packages have not necessarily been developed specifically for MA-XRF scanning."

Yan et al. [26] also make mention of the main challenges of processing this type of data: "Fitting XRF spectra with Gaussian functions requires knowledge of the number, widths and locations of the Gaussian functions. The widths can be obtained by measurement [56] or existing models [52, 48]. However, the number and locations of the Gaussian functions are unknown parameters related to the chemical elements present in the painting, where the number should be the total number of characteristic X-rays for these elements and the locations should be the energies of the characteristic X-rays. As a result, all existing approaches require user input to determine which elements are likely to be present in the dataset. These approaches then fit the spectra based on the specified elements and indicate their distributions in the painting. However, the element distribution maps that may be obtained from a given dataset can vary greatly, depending on the deconvolution approach that is used. For example, if the selection of elements that may be present in the dataset is incorrect

(e.g. elements missed or added in error), it might lead to obtaining inconsistent results. Thus, the selection of elements by the user has a significant effect on the accuracy of the final results." [26].

In an attempt to overcome some of these challenges, Su Yan et al. [26] proposed a novel approach to *automatically* produce the element distribution maps from MA-XRF datasets, without the need for the users to build a complex model or to input any predictions of elements present in the painting. Their approach allows for the detection of very weak element signals and can be divided into the following parts: firstly, pre-processing steps are implemented on the raw data so as to estimate the shapes of the characteristic X-ray peaks in the XRF spectra, which Yan refers to as *pulses* (Fig. 2.4(b), *Pre-processing*). Then, the elemental pulses in the XRF spectra are localised and detected using variations of Finite Rate of Innovation (FRI) theory [57, 58], and Prony’s method [59] (Fig. 2.4(b), *FRI-based pulse detection*). The detected pulses are then assigned to the appropriate characteristic element X-rays to identify which chemical elements are present in each pixel of the painting (Fig. 2.4(b), *chemical element estimation*) [26]. This approach results in two sets of distribution maps for each emission line group: a confidence map and a quantity map are produced for each line group of each detected element [26].

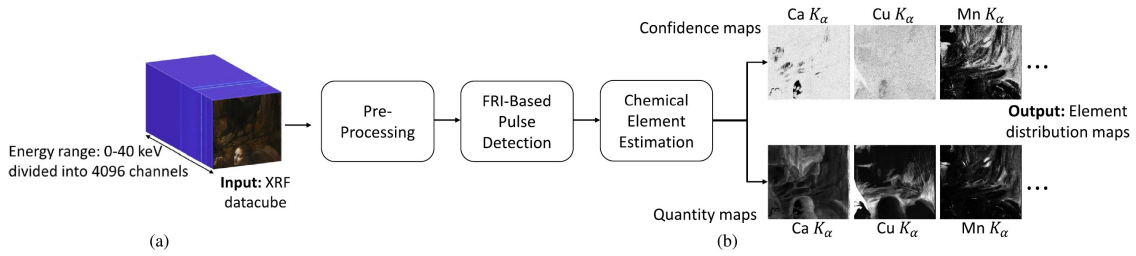


Figure 2.4: This sequence illustrates the processing of the dataset obtained by MA-XRF scanning of “*The Virgin of the Rocks*” (NG1093, The National Gallery, London) [60], area d10. Part (a) shows the MA-XRF dataset, which we call datacube, of width, height and energy dimensions, obtained with a Bruker M6 JETSTREAM instrument. Part (b) shows the steps involved in the method developed by Yan et al. [26] for deconvolving the MA-XRF datacube, and obtaining the images of the confidence and quantity maps - each of these groups of images is arranged into an image stack. Note that for image registration, we work with the image stack of quantity maps. Image after *When de Prony Met Leonardo: An Automatic Algorithm for Chemical Element Extraction From Macro X-Ray Fluorescence Data* [26].

These elemental maps are the technical images obtained from the spectral deconvolution of the raw data obtained by MA-XRF scanning. Note that subjecting the raw data to a process of spectral deconvolution is a necessary step, since the element signals in their original format (i.e. if accessing them from the raw data, straight after being obtained by MA-XRF scanning) are normally too weak to detect. Thus, since the elemental maps are the produce of the spectral deconvolution of a series of X-ray lines for the elements detected in the scanned artwork, these elemental maps are arranged as an image stack, where each image is representative of one element of those detected in the artwork. The elemental maps are presented as grey-scale images, with white representing high quantity, and black representing low quantity of the chemical element detected at each pixel

of the scanned area. Due to the fact that they are generated simultaneously from the same raw data, all elemental maps in the same stack are considered to be aligned by construction.

Note that the elemental distribution maps are indicative of the presence of chemical elements that are associated with the pigments used by the artist at or below the surface of the artwork. Element distribution maps may relate, pictorially, to the features that are apparent in the visible image of the artwork. However, they may also show distributions related to features hidden below the surface of the artwork. Thus, implementing feature detection methods for the registration of this type of images may not always lead to satisfactory results.

The elemental distribution maps resulting from spectral deconvolution of the MA-XRF scanning are the images that we work with, composing a stack of images that is treated as the moving image, for its registration to the visible image of the artwork (Fig. 2.4(b), *quantity maps*).

2.2.3 Experimental details

The MA-XRF datasets included in this research were acquired using the Bruker M6 JETSTREAM equipment: at The Fitzwilliam Museum / Hamilton Kerr Institute for the acquisition of the dataset of *Portrait of Jane Boughton, née Coningsby* (AGOID.29371, The Thomson Collection at the Art Gallery of Ontario, see Fig. 1.2), and at The National Gallery London for datasets of “*The Virgin of the Rocks*” (NG1093, The National Gallery, London, see Fig. 2.5), and “*The Death of Actaeon*” (NG6420, The National Gallery, London, see Fig. 4.9). The equipment operates in a continuous scanning mode and has a 30 W rhodium (Rh) anode X-ray tube with polycapillary optics operated at 50 kV and 600 μ A. The National Gallery equipment has a single 60 mm² silicon (Si) drift detector with a threshold of 275 kcps while the Fitzwilliam Museum/Hamilton Kerr Institute equipment has a dual detector system.

The process of deconvolution of *Portrait of Jane Boughton, née Coningsby* (AGOID.29371, The Thomson Collection at the Art Gallery of Ontario) was conducted with proprietary software of Bruker M6. In the case of the datasets from a series of regions of “*The Virgin of the Rocks*” (NG1093, The National Gallery, London), the spectral deconvolution was conducted with the method proposed by Yan et al. [26] described above. The images resulting from this deconvolution, namely the confidence and quantity elemental maps, are the images produced by this method. The method developed by Yan et al. [26] is designed to detect a total of 34 elements considered likely to be present in easel paintings and detectable using the instrumentation described above [55]. A total of 7 emission line groups are considered based on combinations of the dominant characteristic X-ray lines. These groups are K_α , K_β , L_l , L_α , L_β , L_γ , and M_α . For the data included in this research, only 31 elements were considered, resulting in a total of 93 element maps (Fig. 2.6). Note that only the quantity maps (elemental distribution maps) are considered for subsequent image registration, since they relate pictorially to the artwork under investigation.

Lastly, the dataset of “*The Death of Actaeon*” (NG6420, The National Gallery, London) was

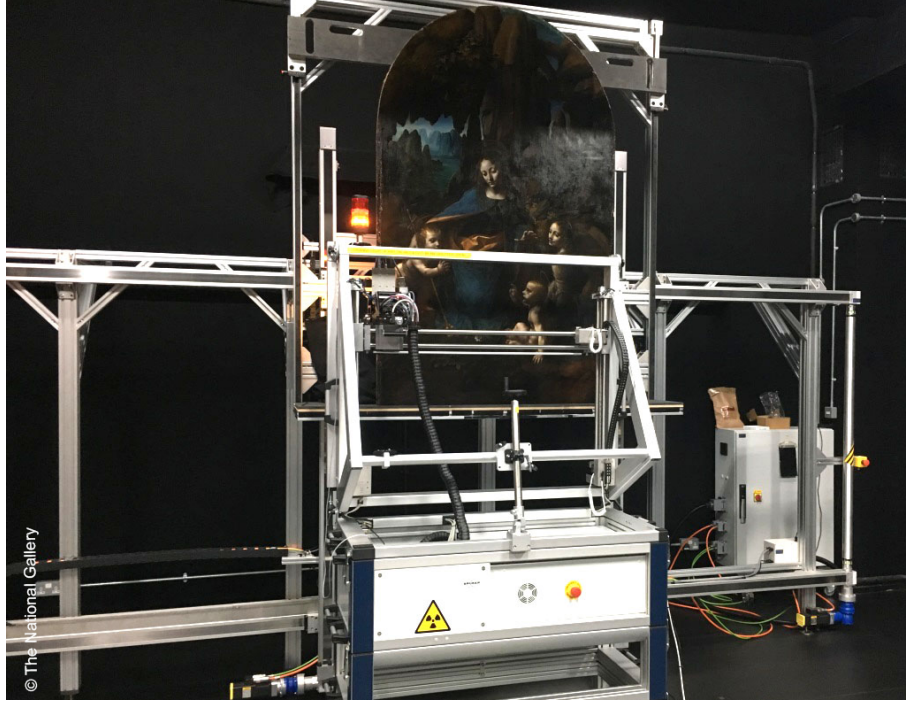


Figure 2.5: Bruker’s M6 JETSTREAM in action at the National Gallery in London. Image after the National Gallery in London <https://www.nationalgallery.org.uk/>.

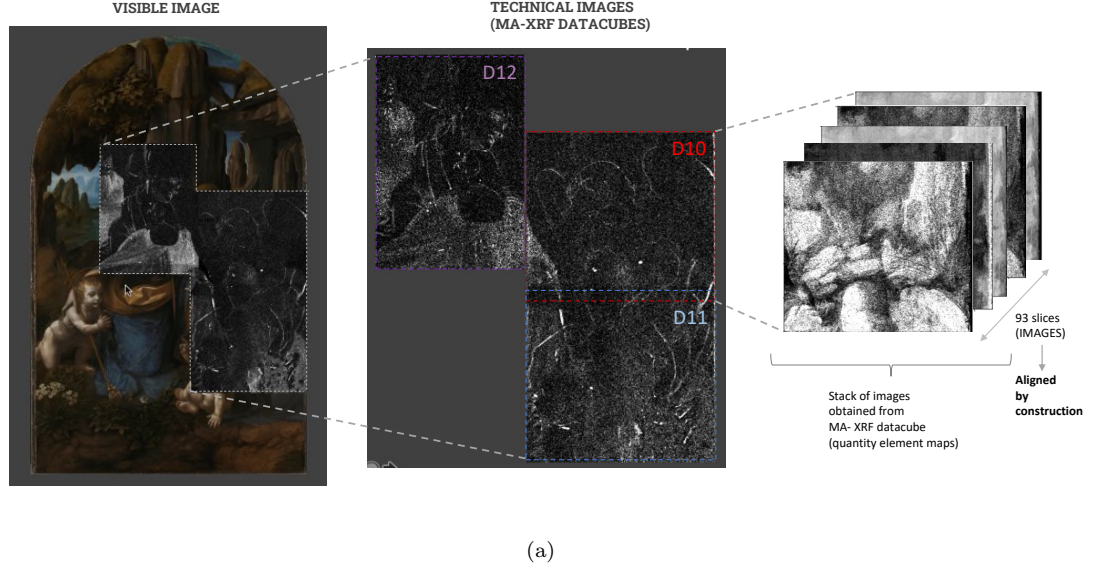
processed with another method proposed by Yan et al. [27], where the spectral deconvolution process generates only chemical elemental quantity maps.

Note that for registration, the automated area-based method for registration of multimodal images of artworks that is presented in Chapter 3 (the main topic of this thesis) allows to use the entire stack of element distribution maps or a just a selection of those, where the elements are present in significant amount. It is important to note that, conceptually, this method forms part of a pipeline of methods optimised for MA-XRF data that started with the methods proposed by Yan et al. [26, 27] under the direction of Professor Dragotti and Doctor Catherine Higgitt, driven by the necessity to register and mosaic the resulting elemental maps.

2.2.4 Challenges to overcome for the registration of images obtained from MA-XRF scanning of artworks

The main characteristics of images obtained with MA-XRF scanning have been presented throughout this section. These characteristics can pose challenges for the registration of technical images of artworks that are obtained with this modality, especially when pursuing their registration to the visible image of the same artwork. These are as follows:

- images obtained with MA-XRF scanning may often show different sub-surface features than those apparent in the target visible image (see Section 2.1).
- while it is possible in some cases to find feature points to guide the registration of multimodal



DATACUBES – VISUALISATION OF SLICES

Image-stack of Datacubes NG1093

(total of 93 images for registration)

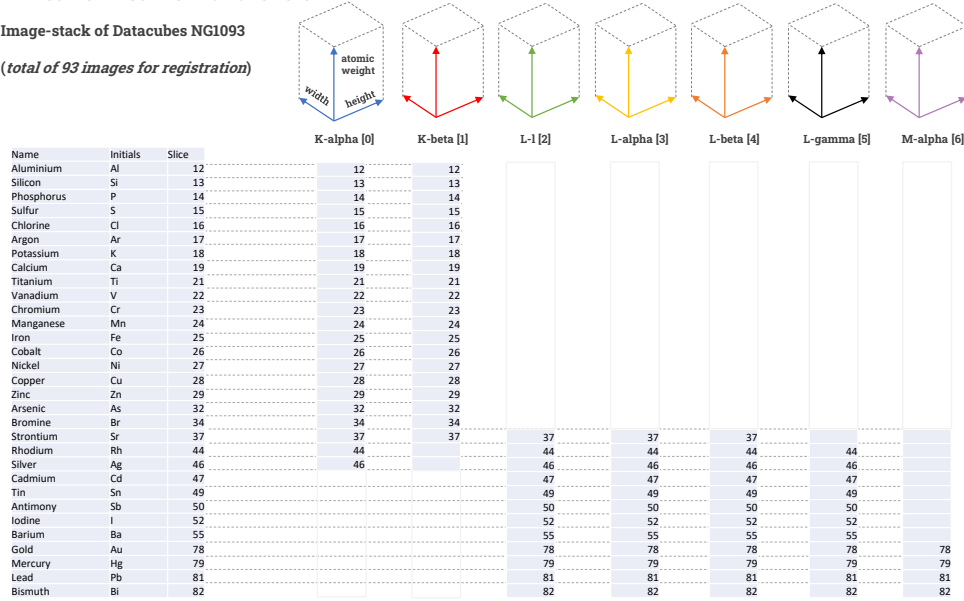


Figure 2.6: MA-XRF scanning of "The Virgin of the Rocks" (NG1093, The National Gallery, London). Part (a) shows a diagram of scanned areas d10, d11 and d12, and a representation of the stack of images corresponding to the 93 element quantity maps obtained by the deconvolution of data acquired with MA-XRF scanning of area d10, using the method by Yan et al. [26]. Part(b) shows a diagram representative of the 93 images retrieved from 31 elements selected taking account of the 7 emission line groups K_α , K_β , L_l , L_α , L_β , L_γ , and M_α . With this technique, 2 datacubes are obtained at each scanning: one datacube of 93 for the quantity maps, and one datacube of 93 for the confidence maps. Note that for registration, we use the datacube of quantity maps. These 93 slices (or a selection of them) may constitute the image stack that is treated as the moving image, for its registration to the visible image of the artwork.

images, it is also common that technical images tend to lack defined details that may be consistently visible across the various modalities - such is the case of images derived from modalities such as Macro X-ray fluorescence (MA-XRF) (see Section 2.1).

- pixel intensities can vary greatly between element distribution maps obtained from MA-XRF scanning of the same artwork, and they might not necessarily maintain a linear relationship with the intensities at the target fixed image. However, the full set of maps need to be registered together to the same visible area of the painting, as all maps that are obtained from the same scanning are aligned by construction (see Section 2.2.2).

These challenges are addressed along the topics that follow in this chapter, making clear the choices of tools that are at the basis of our image registration algorithm that is presented in depth in Chapter 3.

2.3 Feature-based image registration

As mentioned at the beginning of this chapter, feature-based methods rely on identifying and matching corresponding features across the images to register, and thus are structured in the following manner [1]:

- *Feature detection*: To register two images, a number of features are selected from the images and correspondence is established between them. These features are pixels that may be representative of a corner of an object, or another salient detail of the object depicted in the image.
- *Feature matching*: In this step, the correspondence between the features detected in the moving image and those detected in the fixed image is established. For this purpose, various feature descriptors and similarity measures along with spatial relationships among the features are used [2].
- *Computation of the transformation function*: Once features are identified within the images to align, and their correspondence across the images is established, a transformation function can be found to resample the moving image to the geometry of the fixed image. This can be done once the coordinates of the set of corresponding feature points in the images are known. Different types of transformations require different amounts of pairs of corresponding points within the images to align [1]. The higher degree of freedom (DoF) required to register the images, the more corresponding pairs of points will be needed to compute the desired transformation. For instance, affine transformations, with a $\text{DoF} = 6$, require 3 pairs of points across the images to align, whereas homographic transformations, with a $\text{DoF} = 8$, require 4 pairs of points across the images to align.

Each of these three steps -feature detection, feature matching and computation of the transformation function- are addressed separately within the image registration methods that will be presented in this section. Note that whilst some methods allow for the feature detection step to be addressed manually (i.e. NIP2 [16]), the next sections make focus on methods that resolve this step in an automated manner. What follows is a brief introduction to each of these steps, where different yet relevant techniques are presented. These constitute the basis of a selection of feature-based image registrations, to be presented towards the end of this section.

2.3.1 Feature detection methods

Here we present a selection of automated methods to identify features within the images to align.

FAST (Features from Accelerated and Segments Test)

Features from Accelerated and Segments Test (FAST [61, 62, 63]) is a feature detection method, specialising in features that are representative of corners within an image. This algorithm works in the following way: it builds a circle of S pixels to assess whether a candidate point $\mathbf{p}$ is actually a corner. Each pixel in the circle is labeled with an integer number 1 to S in a clockwise manner. It is common to use $S = 16$ (thus building a Bresenham circle of radius 3 [64], see Fig. 2.7). The conditions for $\mathbf{p}$ to be classified as a corner are as follows:

For a set of N contiguous pixels $Q \in S$,

$$\forall \mathbf{x} \in Q, I_{\mathbf{x}} < I_{\mathbf{p}} + t \quad (2.1)$$

OR

$$\forall \mathbf{x} \in Q, I_{\mathbf{x}} > I_{\mathbf{p}} - t \quad (2.2)$$

where $I_{\mathbf{x}}$ is the intensity of the pixel $\mathbf{x}$, $I_{\mathbf{p}}$ is the intensity of the pixel $\mathbf{p}$, and t is a threshold value. These conditions mean that, if a set of N contiguous pixels in the circle are all brighter than the intensity of candidate pixel $\mathbf{p}$ plus a threshold value t , or are all darker than the intensity of candidate pixel $\mathbf{p}$ minus threshold value t , then $\mathbf{p}$ is classified as corner.

There is a tradeoff of choosing N , the number of contiguous pixels and the threshold value t . On one hand the number of detected corner points should not be too many, on the other hand, the high performance should not be achieved by sacrificing computational efficiency. For $S = 16$, N is usually chosen as 12 [64].

Difference-of-Gaussians (DoG) / SIFT

In imaging science, difference of Gaussians (DoG) is a feature enhancement algorithm that involves the subtraction of one Gaussian blurred version of an original image from another, less blurred version of the original (Fig. 2.8). This method is commonly used to support the performance of

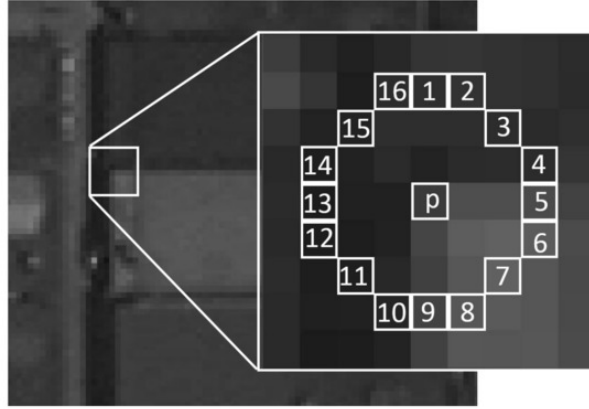


Figure 2.7: FAST corner detector is shown here, using a circle of 16 pixels (a Bresenham circle of radius 3) to classify whether a candidate point p is actually a corner. In this circle, 13 contiguous pixels are darker than pixel p (namely, pixels index 9 to 16, followed by pixels index 1 to 4, with amount of pixels = 13, this being larger than $N = 12$), thus satisfying the condition set in eq 2.1. Then, p is classified as corner. Image from https://commons.wikimedia.org/wiki/File:FAST_Corner_Detector.jpg.

feature detection methods, such as SIFT. In the simple case of grayscale images, the blurred images are obtained by convolving the original grayscale images with Gaussian kernels having differing width (standard deviations). Blurring an image using a Gaussian kernel suppresses only high-frequency spatial information. Subtracting one image from the other preserves spatial information that lies between the range of frequencies that are preserved in the two blurred images. Thus, the DoG is a spatial band-pass filter that attenuates frequencies in the original grayscale image that are far from the band center. [65] Conducting this process at various blurred levels, and then at various image scales constitutes the basis of the feature detection algorithm called SIFT.

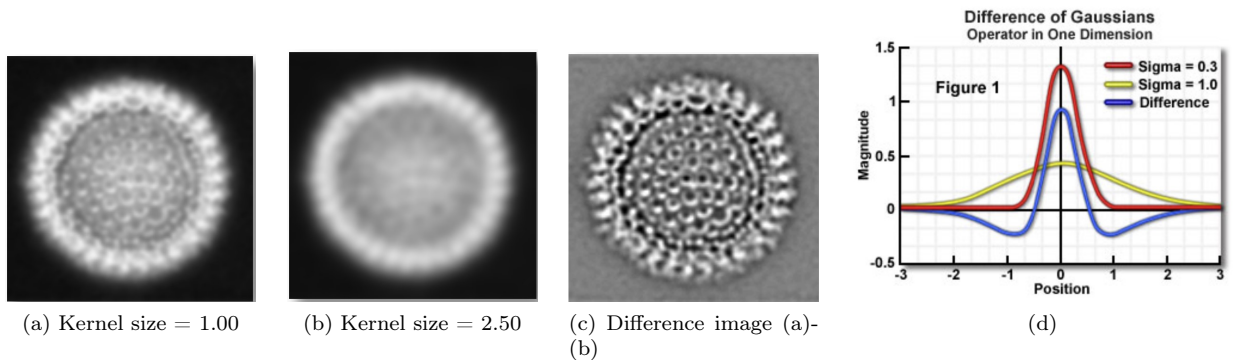


Figure 2.8: Difference-of-Gaussian (DoG). Fig(a) shows the original image, (b) shows the blurred version of the original image with a Gaussian kernel of 2.50, and (c) shows the difference of (a) minus (b), being noticeable the enhancement of the details of the original image. Fig (d) shows the difference of the operators, in one-dimension. Images from Davidson et al. [66].

2.3.2 Feature matching algorithms

Once features have been identified, it is necessary to establish their correspondence across the images to align to define the transformation that will be applied to the moving image. While the matching of features is clear when these are manually selected, automated methods rely on assessing these matches in an algorithmic manner. For this purpose, *descriptors* are applied to each feature. Descriptors are vectors containing information about the pixels surrounding each feature, becoming a compact representation that can be used to distinguish one feature from another [3]. This information can be the intensity of the neighbouring pixels, the changes in intensity values or the gradients in the local area. By assessing the similarity of any two descriptors across the images of interest, corresponding pairs of features can be determined. Thus, the goal of feature matching algorithms is to ensure that each pair of keypoints is indicative of the same feature within these images.

Binary Robust Independent Elementary Features (BRIEF)

Binary Robust Independent Elementary Features (BRIEF [67]) is a feature point descriptor. It implements a binary feature vector to represent the similarity or discrepancy of the neighbouring pixels around a keypoint (pixel coordinates corresponding to a feature within the images). By creating one vector per keypoint, the similarity of these vectors is assessed by computing their Hamming distance, which is more efficient to compute than the L^2 norm² [67].

This method works in the following way: it starts by smoothing the images using a Gaussian kernel in order to prevent the descriptor from being sensitive to high-frequency noise. Then it selects, at random, pairs of pixels in a defined neighborhood around the keypoint. The defined neighborhood of pixel-pairs around the keypoint is known as a *patch*, which is normally set to be a square of 32 x 32 pixels [69]. For each patch p , a vector of length n_d is created by evaluating n_d pairs of pixels $(\mathbf{x}, \mathbf{y})$ within that patch. The following test τ is conducted on p :

$$\tau(p; \mathbf{x}, \mathbf{y}) := \begin{cases} 1 & \text{if } \mathbf{p}(\mathbf{x}) \leq \mathbf{p}(\mathbf{y}), \\ 0 & \text{otherwise} \end{cases} \quad (2.3)$$

where $\mathbf{x}$ and $\mathbf{y}$ are two points (i.e pixel-pairs) within the observed patch p , $\mathbf{p}(\mathbf{x})$ is the pixel intensity at $\mathbf{x} = (u, v)^T$. Patterns to conduct the binary tests (Fig. 2.9) are uniquely defined by choosing a set of n_d $(\mathbf{x}, \mathbf{y})$ -location pairs [67]. Thus, the BRIEF descriptor is taken to be:

$$\tau(p; \mathbf{x}, \mathbf{y}) := \sum_{1 \leq i \leq n_d} 2^{i-1} \tau(p; \mathbf{x}_i, \mathbf{y}_i) \quad (2.4)$$

Note that this descriptor is an n_d -dimensional bitstring, where it is customary that $n_d = 128$, 256 or 512. For each pair, the following assessment is made: If the first pixel is brighter than the

² The L^2 norm computes the Euclidean length of a vector, which is distance from the origin to the point $\mathbf{x}$ [68].

second, it assigns a value of 1 to the corresponding entry in the binary feature vector, otherwise it assigns a value of 0. Once both vectors for the corresponding patches in the fixed and moving images are created, the Hamming distance can be computed. The shorter this distance, the more similar the binary feature vectors are deemed to be. Features are matched with one another based on how similar their respective binary feature vectors are. One of the main advantages of BRIEF is the implementation of the Hamming distance for assessing the similarity between features, which is very efficient to compute, instead of the L^2 norm [67].

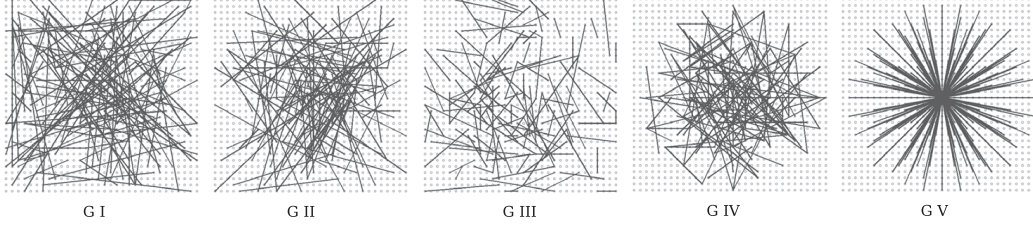


Figure 2.9: Different patterns of BRIEF descriptor patches (32 x 32 pixels in size, with 128 tests in every patch). Equal descriptor patches are placed at the candidate keypoints across the fixed and moving images. Here, each line at each pattern is indicative of the locations of the pixels to test (following Eq. 2.3) and obtain the binary descriptor vector for each keypoint. Note that patches *GI* to *GIV* are all generated by random sampling, except patch *GV* which results from a equivalently-spaced polar distribution. In the paper of Calonder et al. [67], it is noted that patch *GIII* enjoys a better performance in terms of successful keypoint matching. Images after Calonder et al. [67].

RANdom Sample Consensus (RANSAC)

RANdom SAMple Consensus (RANSAC [70]) is a *robust parameter estimator*. When making use of descriptors to assess the correspondence between feature points in two images, the presence of repeated patterns, noise, and geometric and intensity differences between the images can result in the assignment of a few incorrect correspondences. In such cases, where it may not be known which correspondences are correct and which ones are not, a robust estimator can be used to identify some or all of the correct correspondences, and use the coordinates of the corresponding feature points to determine the parameters of the transformation [3].

Thus, once an initial amount of corresponding pairs of feature points has been computed, it is necessary to identify the final subset that will yield the most accurate alignment. One way of doing this is by computing a least squares estimate, or to use a *robustified* (iteratively re-weighted) version of least squares [4]. However, in many cases, the best way to initialise this approach is to identify a set of *inlier* correspondences, i.e., pairs of feature points that are all consistent with some particular estimated motion [4].

At the beginning of this method, a subset of k correspondences is selected at random, which is then used to compute an estimated motion $\mathbf{p}$. The *residuals* of the full set of correspondences $\mathbf{r}_i$

are calculated as follows:

$$\mathbf{r}_i = \bar{\mathbf{x}}'_i(\mathbf{x}_i; \mathbf{p}) - \hat{\mathbf{x}}'_i \quad (2.5)$$

where $\bar{\mathbf{x}}'_i$ are the estimated (mapped) locations, and $\hat{\mathbf{x}}'_i$ are the detected feature points at the moving image [4].

RANSAC then counts the number of inliers that are within ϵ of their detected location, i.e., whose residual $\|\mathbf{r}_i\| \leq \epsilon$. Note that the value ϵ depends on the application, but is commonly around 1-3 pixels. Another robust parameter estimator, Least median of squares (LMS, [71]), finds the median value of the $\|\mathbf{r}_i\|$ values.

This process, of random selection and calculation of residual values $\|\mathbf{r}_i\|$, is repeated S times. The sample set with largest number of inliers (in this case, the selected pairs of feature points) is kept as the final solution. Note that the value of S increases exponentially with the number of correspondences k (for example, in the case of looking for the correspondence to validate pairs of feature points for an affine transformation between two images to align, $k = 3$, as 3 pairs of matching points is necessary to determine this transformation).

More generally, the initial parameter guess $\mathbf{p}$ and the full set of computed inliers are then passed on to the next data fitting stage [4].

To ensure a good chance of finding a true set of inliers from a random sampling of k correspondences, it is necessary to try a sufficient number of trials S . Let P be the total probability of success after S trials, and p the probability that any given correspondence is valid. The probability in one trial that all k random samples are inliers is p^k . In this way, the probability that all S trials will fail is

$$1 - P = (1 - p^k)^S \quad (2.6)$$

and thus the required minimum amount of S trials is

$$S = \frac{\log(1 - P)}{\log(1 - p^k)}. \quad (2.7)$$

In example, in order to obtain a 99% probability of success ($P = 0.99$), Stewart [72] provides the following examples of the minimum amount of S trials that are required:

k	p	S
3	0.5	35
6	0.6	97
6	0.5	293

Table 2.1: RANSAC - number of trials S in relation to correspondences k , so as to attain a 99% probability of success.

Thus, the necessary amount of trials S grows quickly with the amount of correspondences required. This provides a motivation for using the minimum amount of sample points k possible for any given trial. This is, in practice, how RANSAC is normally used [4].

In this case, RANSAC looks for a best-fit model, by exploring the possible transformations that can result from the combination of a restricted number of candidate pairs (k correspondences) of feature points across the images to align. All transformations resulting from the matching of these pairs are evaluated, and the ones that keep the highest number of inliers is selected as the set of pairs to calculate - in the next step - the transformation to apply.

2.3.3 Computing transformations

Algorithms for creating seamless mosaic images by aligning consecutive individual images to one another are among the oldest and most popularly used in computer vision [4]. When working towards image alignment with feature-based methods, once the correspondence between pairs of feature points have been established across the images to align, a transformation function can be determined so as to apply it to the moving image for its registration to the target fixed image. The transformation is a geometric operation, based on the coordinates of corresponding pairs of feature points at both fixed and moving images, and it is used to spatially align the images by transforming the geometry of one image to that of the other [3].

Thus for image alignment, the coordinates of a pixel in one image need to be related to the coordinates of the corresponding pixel in another image. For this purpose, an appropriate mathematical model may be determined to establish this relation. Thus, in this section, *basic motion models* [4] are reviewed. Building up from translation, rotation and scaling; similarity, affine and homography (also known as projective) transformations are described (see Fig. 2.10). They are reviewed because they are fundamental to the development of the image registration method presented in Chapter 3. These are *well-known* transformation functions that have a fixed number of parameters, and describe the global geometric relations between the images to align [3]. These transformations require progressively more corresponding pairs of feature points to compute the desired transformation, in turn allowing for more degrees of freedom to register the images.

In principle, a geometric operation modifies the *coordinates* of image pixels, thus transforming an image I to a new image I' [5],

$$I'(x', y') \leftarrow I(x, y), \quad (2.8)$$

by this operation, the value of image I at pixel with original location (x, y) , moves to the new location (x', y') in the newly transformed image I' . In this way, the *values* of the pixels that are subject to this operation do not change, but only their *locations* change [5]. This process can be modelled in the following way:

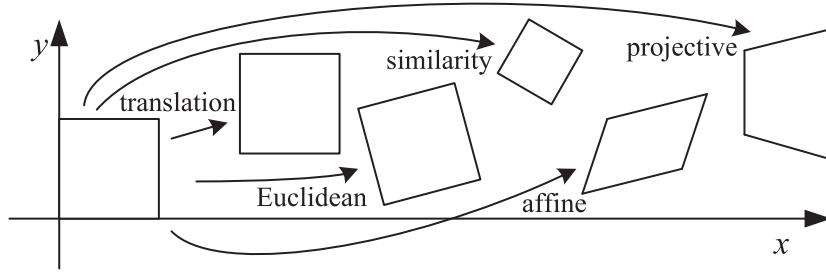


Figure 2.10: Basic motion transformations. Building up from translation, rotation and scaling; similarity, affine and homography (also known as projective) transformations are described in this section. Note that Euclidean transformations result from the combination of translation and rotation only. Image from Paragios et al. [4].

$$T : \mathbb{R}^2 \rightarrow \mathbb{R}^2, \quad (2.9)$$

where T is a 2D transformation function, or geometric mapping, that assigns to each 2D coordinate point $\mathbf{x} = (x, y)$ a new location at a corresponding target point $\mathbf{x}' = (x', y')$ in the newly transformed image I' ,

$$(x', y') = T(x, y) \quad (2.10)$$

It is important to note that the coordinates (x, y) and (x', y') specify point locations in the continuous image plane $\mathbb{R} \times \mathbb{R}$. However, when transforming digital images, the main problem is that the pixels $I(u, v)$ are defined on a discrete grid $\mathbb{Z} \times \mathbb{Z}$, and not on a continuous plane [5]. Thus, it is highly possible that a transformed coordinate $(u', v') = T(u, v)$ produced by the transformation function $T()$ does not correspond to a discrete point location on the grid $\mathbb{Z} \times \mathbb{Z}$. The solution to this problem is to compute the transformation (that is applied to the moving image) in the fixed image domain (process known as *resampling*), and to compute the intermediate pixel values for the transformed image by interpolation.

Basic motion models

The mapping function $T()$ in Eq. 2.10 is an arbitrary continuous function for 2D coordinate transformations. It is commonly presented as two separate functions, for the x and y components [5], respectively:

$$x' = T_x(x, y) \quad \text{and} \quad y' = T_y(x, y) \quad (2.11)$$

Translation (shift) is defined by a vector (t_x, t_y) [5]:

$$\begin{aligned} T_x : x' &= x + t_x \\ T_y : y' &= y + t_y \end{aligned} \quad \text{or} \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \quad (2.12)$$

Scaling (stretching or contracting) along the x or y axis by the factor s_x or s_y respectively [5].

To apply scaling in only one direction, the other factor may be set to one. This is defined as:

$$\begin{aligned} T_x : x' &= s_x \cdot x \\ T_y : y' &= s_y \cdot y \end{aligned} \quad \text{or} \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} \quad (2.13)$$

Shearing along the x and y axis by the factor m_x or m_y respectively [5]. To apply shear in only one direction, the other factor may be set to zero. This is defined as follows:

$$\begin{aligned} T_x : x' &= x + m_x \cdot y \\ T_y : y' &= y + m_y \cdot x \end{aligned} \quad \text{or} \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & m_x \\ m_y & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} \quad (2.14)$$

Rotation by an angle θ , with centre of rotation at the coordinate origin [5]:

$$\begin{aligned} T_x : x' &= x \cdot \cos \theta - y \cdot \sin \theta \\ T_y : y' &= x \cdot \sin \theta + y \cdot \cos \theta \end{aligned} \quad \text{or} \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} \quad (2.15)$$

In order to rotate the image around an arbitrary centre point $\mathbf{x}_c = (x_c, y_c)$ (i.e placing the centre of rotation at the top left corner of the image), the image needs to be translated, by $(-x_c, -y_c)$, to that point $\mathbf{x}_c$. Then, the image needs to be rotated (as per Eq. 2.15), and then translated back again to its original position, by (x_c, y_c) [5]. Therefore:

$$\begin{aligned} T_x : x' &= x_c + (x - x_c) \cdot \cos \theta - (y - y_c) \cdot \sin \theta \\ T_y : y' &= y_c + (x - x_c) \cdot \sin \theta + (y - y_c) \cdot \cos \theta \end{aligned} \quad \text{or} \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x_c \\ y_c \end{pmatrix} + \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \cdot \begin{pmatrix} x - x_c \\ y - y_c \end{pmatrix} \quad (2.16)$$

Similarity transformation

Similarity transformations are the result of combining translation, rotation and isotropic scaling. This type of transformation is also known as the *transformation of the Cartesian coordinate system*[3]. This is defined by:

$$\begin{aligned} T_x : x' &= x \cdot s \cdot \cos \theta - y \cdot s \cdot \sin \theta + t_x \\ T_y : y' &= x \cdot s \cdot \sin \theta + y \cdot s \cdot \cos \theta + t_y \end{aligned} \quad (2.17)$$

where s is the isotropic scaling factor (with $s = s_x = s_y$), θ is the rotation angle, and (t_x, t_y) is the translation (i.e. new location of the coordinate system origin of the moving image with respect to that of the fixed image). Note that similarity transformations allows for four degrees of freedom,

where the amount of degrees of freedom corresponds to the amount of independent variables (here the above-mentioned parameters) that can be modified in the transformation that is applied to the moving image. In matrix form, this can be written as the multiplication of the matrices of each one of the four transformations described above, as follows:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (2.18)$$

Note that parameters s , θ , t_x , and t_y can be determined by establishing a minimum of two pairs of corresponding points in the images. In this type of transformation, the angle between corresponding lines in the original image will remain unchanged in the newly transformed image. By calculating the length ratio of the line segment obtained from the two points in the moving image over the same line segment in the fixed image, the scaling factor that is applied to the moving image with respect to the fixed image can be determined [3].

Affine transformation

Affine transformations are the result of the combination of translation, scaling, shearing and rotation - namely the transformations presented in Equations 2.12, 2.13, 2.14, 2.15. This type of transformation allows to map an image into another with six degrees of freedom and using three pairs of corresponding points (Fig. 2.11(a)). By means of an affine transformation, parallel lines in the original image will remain parallel in the newly transformed image. In matrix form, this can be written as:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} 1 & m_x \\ m_y & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \quad (2.19)$$

where s_x , s_y are the scaling parameters along x and y axis respectively, m_x , m_y are the shear parameters along x and y axis respectively, θ is the angle of rotation and t_x , t_y are the translations along x and y axis respectively. This can be represented in homogeneous coordinates as a matrix multiplication [73], as follows:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & m_x & 0 \\ m_y & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (2.20)$$

which, expanding brackets [73], gives:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s_x \cos \theta + s_x m_x \sin \theta & s_x m_x \cos \theta - s_x \sin \theta & t_x \\ s_y m_y \cos \theta + s_y \sin \theta & s_y \cos \theta - s_y m_y \sin \theta & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \quad (2.21)$$

This is the longer, more explicit version of the most common way of writing the matrix for affine transformations, where the six degrees of freedom (in particular, parameters $a_{11}, a_{12}, t_x, a_{21}, a_{22}, t_y$) become more evident:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \quad (2.22)$$

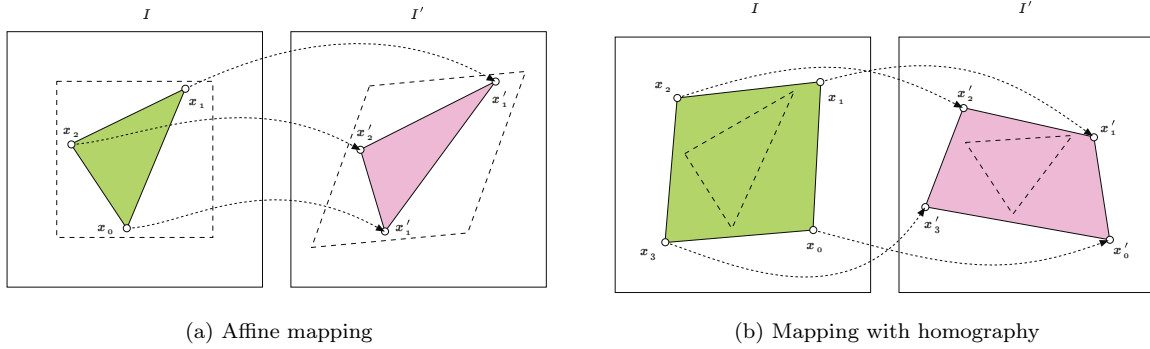


Figure 2.11: Mapping with affine transformation and homography. Part (a) shows mapping with three pairs of corresponding points $(x_0, x'_0), (x_1, x'_1), (x_2, x'_2)$. This is sufficient to specify a unique affine 2D transformation. Note that when a shape undergoes an affine transformation, parallel lines remain parallel. Part (b) shows mapping with four pairs of corresponding 2D points, $(x_0, x'_0), (x_1, x'_1), (x_2, x'_2), (x_3, x'_3)$. This is sufficient to specify a unique projective transformation. In this type of mapping, straight lines may remain straight. However in general, neither parallelism between lines nor length ratios may be preserved, causing a rectangle in the moving image to be mapped to a quadrilateral of different proportions in the transformed moving image. Images from Burger and Burge [5].

Homography

Whilst affine transformations allow for mapping between images with three pairs of corresponding feature points, homography (also referred to as *projective* or *homographic* transformations) allows for mapping between images with four pairs of corresponding feature points. Thus, mapping a set of points (x_0, x_1, x_2, x_3) to their corresponding locations (x'_0, x'_1, x'_2, x'_3) requires eight degrees of freedom - two more than the six degrees of freedom needed for affine transformations. Note that, whilst lines that are parallel remain parallel after an affine transformation, in the case of a homography, straight lines remain straight after being mapped with this type of transformation.

Adding two more degrees of freedom to the affine transformation (Eq. 2.22), a homography (Fig. 2.11(b)) can be expressed in homogeneous coordinates [5] as follows:

$$\begin{pmatrix} x' \\ y' \\ s \end{pmatrix} = \begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (2.23)$$

with a_{20} and a_{21} as the two additional parameters, and a scalar $s = a_{20}.x + a_{21}.y + 1$. Expressing this in Cartesian coordinates [5] this is:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \frac{1}{a_{20}.x + a_{21}.y + 1} \cdot \begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (2.24)$$

Thus, homographic transformations allow for preserving straight lines in the newly transformed image. Homography is the most general transformation that allows for mapping straight lines (from the original moving image) to straight lines (in the transformed moving image) [5], as well as it allows for mapping any N th-order algebraic curve onto another N th-order algebraic curve. For example, a circle may be mapped onto another second-order curve, such as an ellipse or another conic section. Note that distances between points on a line may not be proportionally preserved, and parallel lines do not necessarily remain parallel after a homography mapping. Thus, homographic transformations are also known as "perspective" or "pseudo-perspective" [5].

Homographic transformation parameters from four pairs of points

Let us have four pairs of corresponding 2D points $(\mathbf{x}_0, \mathbf{x}'_0), (\mathbf{x}_1, \mathbf{x}'_1), (\mathbf{x}_2, \mathbf{x}'_2), (\mathbf{x}_3, \mathbf{x}'_3)$, with one point $\mathbf{x}_i = (x_i, y_i)^T$ in the moving image and one point $\mathbf{x}'_i = (x'_i, y'_i)^T$ in the fixed image. In this way, we have eight unknown transformation parameters $a_{00}, a_{01}, a_{02}, a_{10}, a_{11}, a_{12}, a_{20}, a_{21}$, which can be found by solving a system of linear equations. Multiplying Eq. 2.24 by the common denominator on the right hand side [5] gives:

$$\begin{aligned} x' \cdot (a_{20}.x + a_{21}.y + 1) &= a_{00}.x + a_{01}.y + a_{02} \\ y' \cdot (a_{20}.x + a_{21}.y + 1) &= a_{10}.x + a_{11}.y + a_{12} \end{aligned} \quad (2.25)$$

and thus

$$\begin{aligned} a_{20}.x.x' + a_{21}.y.x' + x' &= a_{00}.x + a_{01}.y + a_{02} \\ a_{20}.x.y' + a_{21}.y.y' + y' &= a_{10}.x + a_{11}.y + a_{12} \end{aligned} \quad (2.26)$$

for any pair of corresponding points $\mathbf{x} = (x, y)^T$ and $\mathbf{x}' = (x', y')^T$.

By rearranging Eq. 2.26 and respectively inserting the known moving and fixed point coordinates (x_i, y_i) and (x'_i, y'_i) , the following pair of linear equations can be obtained:

$$\begin{aligned} x'_i &= a_{00}.x_i + a_{01}.y_i + a_{02} - a_{20}.x_i.x'_i - a_{21}.y_i.x'_i \\ y'_i &= a_{10}.x_i + a_{11}.y_i + a_{12} - a_{20}.x_i.y'_i - a_{21}.y_i.y'_i \end{aligned} \quad (2.27)$$

for each point pair $i = 0, \dots, 3$ and the eight unknowns $a_{00}, \dots, a_{21}$. This results in eight equations. Then, we can proceed to combine these equations in the usual matrix notation form. This yields:

$$\begin{pmatrix} x'_0 \\ y'_0 \\ x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \end{pmatrix} = \begin{pmatrix} x_0 & y_0 & 1 & 0 & 0 & 0 & -x_0x'_0 & -y_0x'_0 \\ 0 & 0 & 0 & x_0 & y_0 & 1 & -x_0y'_0 & -y_0y'_0 \\ x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1y'_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2x'_2 & -y_2x'_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2y'_2 & -y_2y'_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3x'_3 & -y_3x'_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -x_3y'_3 & -y_3y'_3 \end{pmatrix} \cdot \begin{pmatrix} a_{00} \\ a_{01} \\ a_{02} \\ a_{10} \\ a_{11} \\ a_{12} \\ a_{20} \\ a_{21} \end{pmatrix} \quad (2.28)$$

or

$$\mathbf{b} = \mathbf{M} \times \mathbf{a}. \quad (2.29)$$

Note that all elements of the vector $\mathbf{b} = (x'_0, \dots, y'_3)^T$ and the matrix $\mathbf{M}$ are derived from the coordinates of the four pairs of corresponding points, and are thus known. By solving the system of linear equations in Eq. 2.29 with standard numerical methods³, the unknown parameters $\mathbf{a} = (a_{00}, \dots, a_{21})^T$ can be found [5].

If more than four pairs of corresponding points are used for recovering the eight parameters of a homography, the system of linear equations in Eq. 2.28 may have more equations than unknowns. Thus, the system may become overdetermined. Normally, n corresponding pairs of points result in having $2n$ equations. In this way, the vector $\mathbf{b}$ in Eq. 2.28 is of length $2n$ and the matrix $\mathbf{M}$ is of size $2n \times 8$, whilst vector $\mathbf{a}$ remains the same. This type of overdetermined systems can be solved using a least-squares method (minimising $\|\mathbf{M}\mathbf{a} - \mathbf{b}\|$), for example, using the singular-value (SVD) [5].

Implementing transformations: Resampling and Interpolation

Throughout this section we have introduced a selection of geometric transformations that can be applied to a moving image for its registration to a target fixed image. An aspect that needs to be considered is that, although transformations are presented as applicable in a continuous domain, digital images are composed of discrete elements (pixels) at integer-valued coordinates [5]. Thus,

³ for solving linear systems, such as the Gauss algorithm.

when applying a transformation to a digital image, we are effectively transferring information from a discrete domain (that of the moving image) into another discrete domain (that of the fixed image).

One way this can be done is by mapping the moving image directly onto the fixed image (Fig. 2.12 (a)). This approach is commonly known as *source-to-target* [5]:

$$(x', y') = T(u, v) \quad (2.30)$$

Where (x', y') are continuous coordinates in the fixed image I' , and T is the transformation to map each pixel (u, v) in the moving image I onto a location within I' . However, one problem that can arise from this approach is that some elements (x', y') in the fixed image I' may not be assigned a value from the moving image I . This may be the case if T is a transformation where I is enlarged, potentially leading to holes in the resulting image as a consequence of consecutive discrete pixels in the moving image being mapped too far away each other in the fixed image domain [5]. The opposite case may also be problematic, as shrinking the moving image can result in one location in the fixed image domain to be assigned with duplicate information from the moving image and thus lose image content. In order to avoid the above-mentioned complications, it is customary to reverse the process of image generation, by mapping the discrete positions (u', v') in the fixed image, to the continuous positions (x, y) in the moving image with the inverse transformation T^{-1} . This approach is known as *target-to-source* [5].

$$(x, y) = T^{-1}(u', v') \quad (2.31)$$

Again, the position (x, y) may not correspond to a precise discrete pixel (Fig. 2.12 (b)), and so is necessary to assign a value following interpolation of the corresponding neighbouring pixel values. This is the standard approach for mapping images that are subject to geometrical transformations.

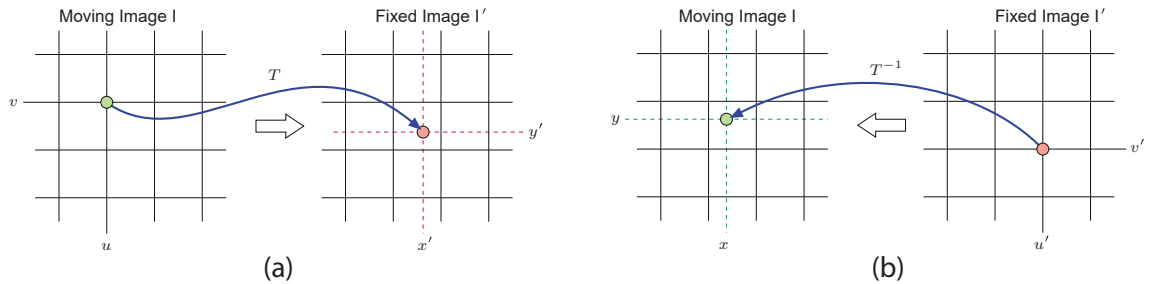


Figure 2.12: Part (a) shows the *source-to-target* approach [5]. This approach is particularly problematic for transformations concerning scaling of the moving image. Part (b) shows the *target-to-source* approach [5], which overcomes this challenge by mapping the discrete positions (u', v') in the fixed image, to the continuous positions (x, y) in the moving image with the inverse transformation T^{-1} . Images after: Images after Burger and Burge [5].

2.3.4 Examples of feature-based image registration methods

Building on the concepts presented so far, we proceed to introduce a selection of feature-based image registration methods. They all follow the structure of extracting features and evaluating their correspondence across the images to align, for then identifying a best transformation to apply to the moving image for its registration to the target fixed image.

Together with the description of the method, we discuss their application for the registration of technical images associated with art investigation.

Scale invariant feature transform (SIFT)

Scale invariant feature transform (SIFT, [74, 75]) is an algorithm for feature detection and description. This method uses difference-of-Gaussian (DoG, see Fig. 2.8) at various image scales and levels of Gaussian blur for detecting feature points, acting as a multi-scale corner detector. For matching the features once identified, it implements a rotation-invariant feature descriptor of, typically, 128 dimensions. The area around the feature point is subdivided into 16 quadrants, and a histogram of eight bins for gradient directions is obtained for each subdivision. The resulting histograms are then concatenated, resulting in the 128-dimension descriptor [5]. The main steps involved in the calculation of SIFT features are as follows:

- Identification of feature points in scale space (across various scales of the same image) by looking for locations that are maxima or minima of a DoG function, so that features can be localized under a wide range of viewing conditions and different scales (Fig. 2.13 (a, b, c)).
- Key point refinement by fitting a continuous model to determine precise location and scale - this process ensures the feature point is a salient feature in the adjacent images along the scale space. This is done by comparison of neighbouring pixel values in relation to the candidate feature point (see Fig. 2.13 (d)).
- Orientation assignment by the dominant orientation of the feature point from the directions of the surrounding local image gradients (Fig. 2.13 (e, f, g)).
- Implementation of a feature descriptor by creating a normalised histogram of local gradients at each feature point (Fig. 2.13 (h, i, j, k, l)).

Thus, the SIFT detector establishes feature point matches by first estimating the scale and orientation of a region of interest in the image (detected by the above-mentioned sequence of operations) and then calculating its centre. From the experimental results obtained by Goshtasby [3], he concludes that overall, the SIFT detector is the most accurate at a comparison study published in 2012 against other over 10 other descriptors. Please note in this comparison study, many well-known descriptors are part of this analysis, but ORB was not available at that stage and so is not included in the comparison charts.

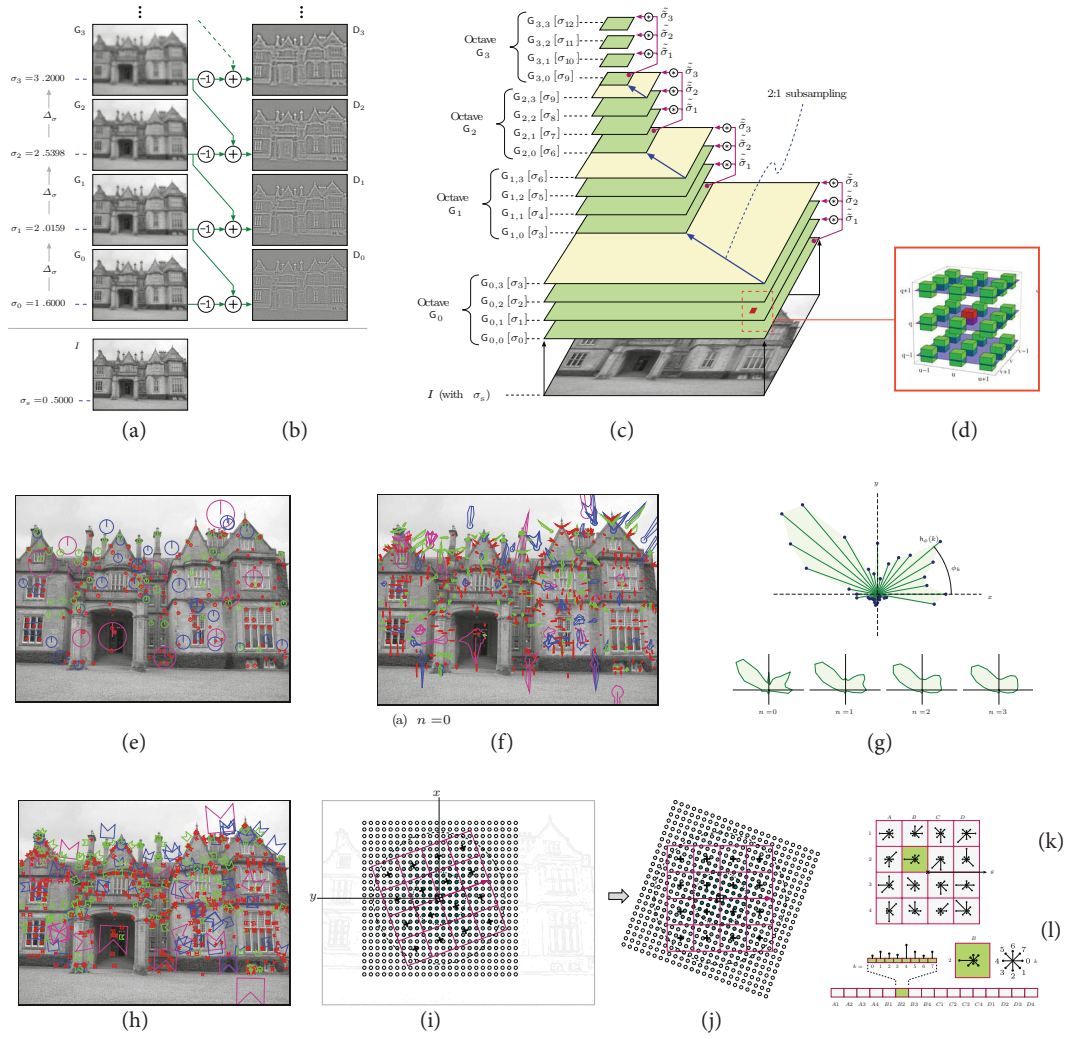


Figure 2.13: This sequence describes the steps of the SIFT algorithm. Part (a) shows the input image (prefiltered with a Gaussian of width $\sigma_s=0.50$) and the first four levels of Gaussian blur (G_0 to G_3 as slices of the Gaussian scale space). From these, the images $D_0, \dots, D_3$ (DoG scale-space) are obtained by calculating the point-wise difference $G_{m+1} - G_m$ (part (b)). Part (c) shows the Hierarchical Gaussian scale space. Each octave extends over 3 steps, where the first image at each octave is obtained by 2:1 sub-sampling of the top level image below. Within each octave, the same set of Gaussian kernels ($\sigma_1, \sigma_2, \sigma_3$) are used to calculate the following levels from the octave's base level $G_{p,0}$. Part (d) shows the diagram of a pixel selected as potential feature (in red), and its 26 neighbouring pixels (in green) for comparison of extrema in the DoG scale space, at lower and upper consecutive image levels. Part (e) shows selected feature point candidates, after rejecting edge-like features, by controlling the maximum curvature ratio. Part (f) shows the selected feature point candidates aligned to their dominant orientation, calculated from the surrounding local image gradients after a smoothing process of the corresponding orientation histogram (part (g)). Part (h) shows final feature points with sizes indicating their absolute scale, and colors indicating the octave at which each feature point was detected, with red = 0, green = 1, blue = 2, magenta = 3. Part (i) and (j) show geometry of the SIFT descriptor in relation to the discrete sample grid of the associated octave. Part (k) shows the descriptor structure for collecting 8 angles (bins) at each quadrant, and the resulting histogram of 8 bins $\times$ 16 quadrants = 128 dimensional feature vector. Images after Burger and Burge [5].

Note that methods currently in use for heritage science and art investigation such as DataHandlerP [76, 77], ImageCompare [12, 78], ImageJ (using macros developed by NU-ACCESS) [79, 80] and Fast adaptive multimodal feature registration (FAMFR) [81] all use SIFT for detecting and matching features across images to align.

Oriented FAST and Rotated BRIEF (ORB)

Oriented FAST and Rotated BRIEF (ORB [82]) is an algorithm for feature detection and description, introduced in 2011 as a computationally efficient replacement to SURF (Speeded-Up Robust Features [83]) and SIFT [74]. ORB builds on the FAST feature detector (see 2.3.1) and on the BRIEF descriptor (see 2.3.2), and enhances these methods by introducing the following variations to the method:

- The addition of a measure of *cornerness* and an orientation component to FAST. The first aspect is addressed by employing a Harris Corner measure [84], to avoid identifying features along edges. The second aspect is addressed by computing the intensity centroid [85] of the area around the identified feature, which makes it possible to construct a vector (from the feature to the centroid) thus providing orientation to each feature detected.
- The efficient computation of oriented BRIEF features. This is done by computing the binary tests of BRIEF together with a rotation matrix R_θ , with θ being discretized at intervals of $2\pi/30$ (12 degrees) [82].
- Analysis of variance and correlation of oriented BRIEF features. This validates the importance of ensuring non-correlation between the pairs of features to compare at every descriptor, so that every test can contribute to the result (Fig. 2.14(a-b)).
- A learning method for de-correlating BRIEF features under rotational invariance, leading to better performance in nearest-neighbor applications (Fig. 2.14(c)).

Following the above-mentioned enhancements, the main components of this method are thus re-named as Oriented FAST and Rotated BRIEF. Tests are conducted by performing homographic transformations applied to moving images for their registration to their corresponding target fixed images, in the following manner:

- Pick a reference view V_o
- For all V_i , find a homographic transformation H_{io} that maps $V_i \rightarrow V_o$.
- Now, use the H_{io} as ground truth for descriptor matches from SIFT, SURF and ORB.

The development of ORB leads to improvements in the computational burden that is characteristic of SIFT (refer to 2.3.4). ORB performs as well as SIFT (and better than SURF) on tasks like panorama stitching and patch tracking, whilst being much faster.

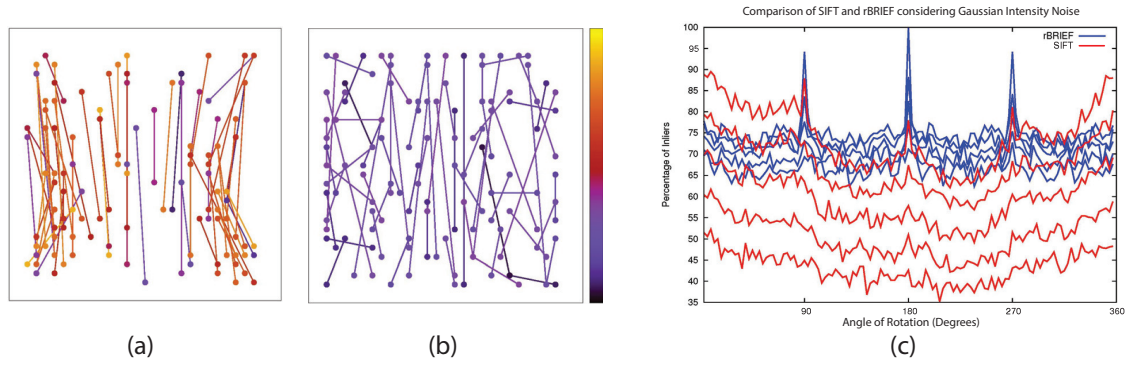


Figure 2.14: Part (a) shows a subset of the binary tests generated by considering high-variance under orientation, whereas Part (b) shows test generated by running the learning algorithm to reduce correlation. Note the distribution of the tests around the axis of the keypoint orientation, which is pointing up. The color coding shows the maximum pairwise correlation of each test, with black and purple being the lowest. The learned tests clearly have a better distribution and lower correlation.

Part (c) shows performance of feature pair matching of SIFT and rBRIEF, when tested for images subject to noise. The noise levels are 0, 5, 10, 15, 20, and 25. Note that SIFT performance degrades rapidly, while rBRIEF is relatively unaffected. Images after Rublee et al. [82].

Thus, ORB is regarded as one of the best performing feature detection and description methods available at the time of this research, in terms of successful feature matching as well as in regards to speed of computation.

In the following section, we show experiments done with a modified version of ORB, to specifically address the changes in intensity values that is characteristic of multimodal images, and is particularly challenging for registration of images obtained with MA-XRF scanning.

Modified ORB - Testing and Results

As mentioned at the beginning of Section 2, feature-based methods may struggle to identify matches between features when seeking registration of images of *different levels* of intensity values. This is due to two main reasons. The descriptor might assign a wrong orientation vector (based on the gradients of the neighbouring pixels around the feature point to evaluate, see 2.3.4, and Fig. 2.13(f)) or might return an invalid score when computing the Hamming distance between two feature vectors of the equivalent feature across the moving and fixed images (see 2.3.2). Both of these situations may lead to the feature descriptor being unable to successfully relate corresponding pairs of feature points within the images to align, thus resulting in unsuccessful registration results.

Images of artworks obtained with MA-XRF scanning often present this challenge: for a particular region of the associated pictorial composition within a given dataset, some element distribution maps may show this region in white if the corresponding chemical elements are present in high abundance. However, the same region may appear dark grey or black in other element distribution maps obtained from the same scan area (reflecting the absence of some other corresponding chem-

ical elements in that same region). Thus, with MA-XRF element maps, pixel intensities can vary greatly between the images obtained from the same scanning of the artwork. Nonetheless, these sets of element maps still need to be registered *together* to the same visible area of the painting. Thus, it is necessary to consider these challenges if addressing registration of this type of images with feature-based methods.

Aiming to address the registration of images obtained from MA-XRF scanning, a first attempt was to implement a modified version of ORB so as to be able to establish correspondences between feature points even if the images to align have different intensity levels. ORB was selected as the feature-based method with highest performance among the ones available at the time of this research.

Here we present experiments done with a modified version of ORB. The main modification consists in allowing more versatility to its feature matching algorithm (Rotated BRIEF, see 2.3.4) to specifically address the discrepancies in intensity values that are common across multimodal images and are particularly challenging for registration of images obtained with MA-XRF scanning.

We also show the reasons why it ultimately failed to provide corresponding matches within images of this type, and how it was necessary to explore area based methods to successfully address their registration to the visible image of the artwork.

Firstly, synthetic images were prepared in order to simulate a strong difference of intensity values across the images to align (Fig. 2.16(a)). Then, building from the feature matching algorithm as defined in Eq. 2.3 where this was conditioned to account for pixels of either *higher* or *lower* intensity values, the algorithm was changed to accommodate pixels of either *similar* or *different* intensity values. Then, the following test τ is conducted on each patch p (area of neighbouring pixels around each feature point) as follows:

$$\tau(p; \mathbf{x}, \mathbf{y}) := \begin{cases} 1 & \text{if } |\mathbf{p}(\mathbf{x}) - \mathbf{p}(\mathbf{y})| \leq t, \\ 0 & \text{otherwise} \end{cases} \quad (2.32)$$

where $\mathbf{x}$ and $\mathbf{y}$ are two points (i.e pixel-pairs) within the observed patch p , $\mathbf{p}(\mathbf{x})$ is the pixel intensity at $\mathbf{x} = (u, v)^T$, and t is a threshold value to consider when comparing the intensities at each pixel-pair. In the same way, for each patch p , a descriptor is created based on a vector of binary tests of length n_d , by evaluating n_d pairs of pixels within that patch (Fig.2.15).

As with the standard ORB, the process for selecting the transformation to apply to the moving image is as follows: features points are selected across both moving and fixed images. Each feature descriptor in the moving image is compared to each feature descriptor in the fixed image, and will be matched to the one returning the lowest Hamming distance. All feature pairs are then sorted into a list in ascending order (pairs with lowest Hamming distance at the top position). Feature pairs from this list are grouped into sets of four pairs (so as to determine a homography), in all possible combinations. The total amount of combinations is $C_{n,k} = \frac{n!}{k!(n-k)!}$, with n = number of

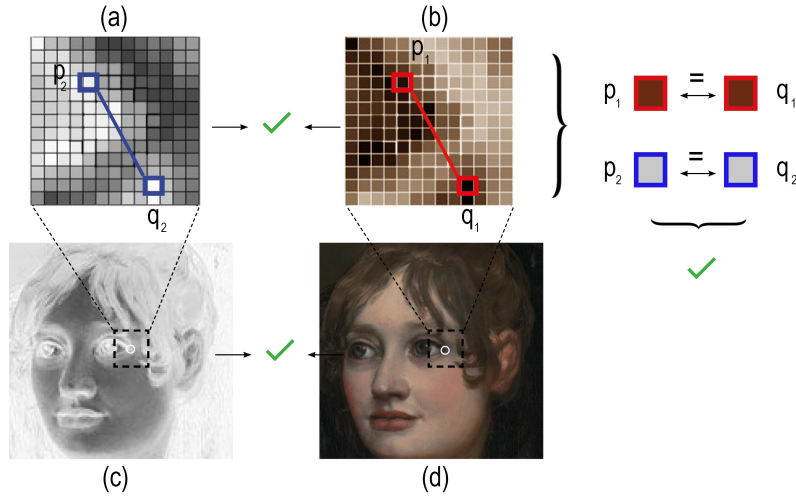


Figure 2.15: This sequence illustrates the construction of the binary vector at each patch around each selected feature point. Parts (a) and (b) are the patches around the feature points selected in the moving image (c) and fixed image (d), correspondingly. Note that in this case, the moving image is a synthetic image for testing purposes, since it has been generated by inverting the intensity levels of the visible image (greyscale version). At patch shown in (a) for the moving image, pixel-pair (p_2, q_2) is evaluated following Eq. 2.32, as well as in (b), where pixel-pair (p_1, q_1) is evaluated at the corresponding patch at the fixed image. Since both pixel-pairs are of similar intensity (and thus within the threshold t), the value to compute at the binary vector will be the same, helping to achieve a low Hamming distance between the resulting vectors. Images of “*Doña Isabel de Porcel*” (NG1473, The National Gallery, London) after Spring et al. [7].

feature pairs, and $k = 4$. Thus, each set of four feature pairs allows a homographic transformation to be constructed to apply to the moving image for its registration to the target fixed image. From these transformations, a best sample is chosen with the RANSAC algorithm (see 9). This proved to find successful transformations for registering images of dissimilar levels of intensity. Pairs of tie points were now successfully matched at the corresponding regions of the painting’s technical and visible images (Fig. 2.16(a)). However, as mentioned in Section 2.1, another characteristic of multimodal images is that images may show different sub-surface features than those apparent in the target visible. In this situation, the modified version of the Rotated BRIEF algorithm failed to return successful matches of tie points (Fig. 2.16(b)), since it failed at the description of the neighbouring pixels due to the differences in the layouts of the corresponding patches (Fig. 2.17).

This analysis highlights that rather than looking isolatedly at particular patches within the images, it is desirable to work with a method that uses area-based metrics involving the full layout of the images to align. This formed the basis of the decision to pursue an area-based method as presented in depth in Section 3.

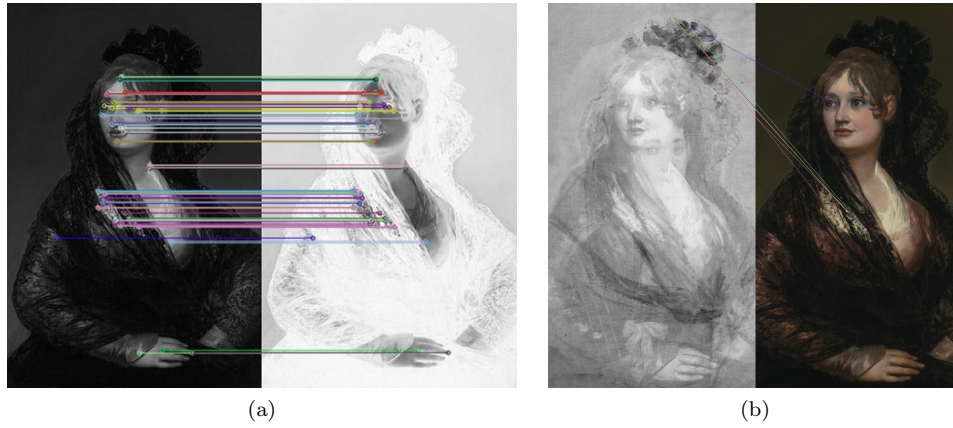


Figure 2.16: This sequence illustrates the modified feature matching algorithm in action, matching feature points across the images with different levels of intensity in a successful manner (visible image in greyscale, and its intensity-inverted image, shown in part (a)). However, it fails to do so when implementing this method for registration of real-life multimodal images (colour version of the visible image, and Pb-L β element map, shown in part(b)). This is due to the fact that such technical images may show different sub-surface features than those apparent in the target visible, making it difficult to ascertain the correspondence between descriptors of the same features across the images to align. Images of “*Doña Isabel de Porcel*” (NG1473, The National Gallery, London) after Spring et al. [7].

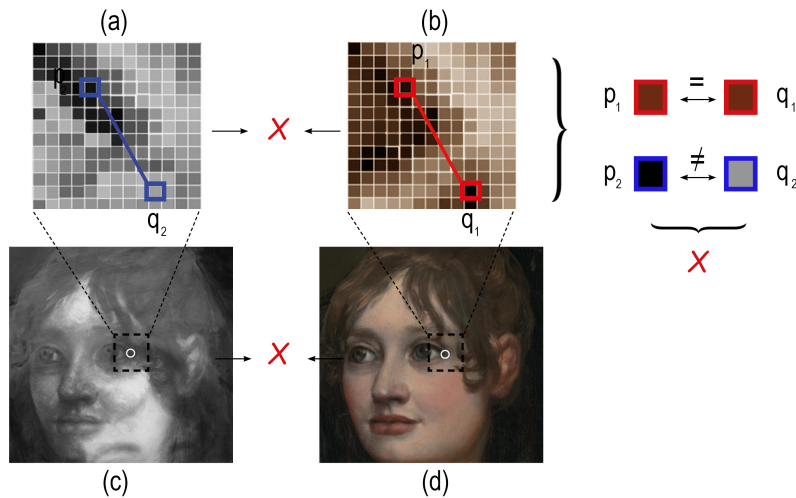


Figure 2.17: As in Fig. 2.15, this sequence illustrates the construction of the binary vector at each patch around each selected feature point. Note that in this case, the moving image (part(c)) is a real technical image of the artwork, showing different sub-surface features than those apparent in the target visible. This poses a challenge when building the feature descriptor at the corresponding locations. Since the patch to evaluate at the moving image (part(a)) may include pixels corresponding to elements that are not apparent in the corresponding patch (part(b)) in the fixed image (part(d)), the resulting Hamming distance between pixel-pairs (p_2, q_2) and (p_1, q_1) will be high and thus will not be considered by the algorithm as a valid feature match. Images of “*Doña Isabel de Porcel*” (NG1473, The National Gallery, London) after Spring et al. [7].

NGA-GWU-HSI-registration algorithm (Conover’s wavelet-based method)

Proposed by Conover et al. [11], this is an automated feature-based method that is designed to address multimodal registration of technical images of artworks. The method has been developed for use at the National Gallery of Art in Washington in the form of three tools:

1. The first tool is designed for the registration of a technical image to the visible image of the same artwork.
2. The second tool can be made to work with a stack of images as the moving image but it bases the transformation on just one image in the stack, and it then applies the resulting transformation to all of the images (e.g. for registration of hyperspectral images).
3. The third tool can be made to work with multiple, adjacent individual images so as to register them simultaneously and thus create a mosaic image with reference to a given visible image (e.g. for the registration of a group of adjacent infrared reflectograms or X-ray images).

Conover acknowledges the challenges presented by discrepancies in pictorial composition that are characteristic of multimodal technical images of artworks. He and his co-authors mention pentimenti and underdrawings as elements that may affect the successful matching of corresponding feature points across images to align [11, 39, 56]. Thus, their work focuses on features that may be available across all modalities, such as craquelure, bubbles and blisters (which are attainable when the technical-moving image has a spatial resolution similarly high as that of the visible-fixed image, and unfortunately this is often not the case with images obtained from MA-XRF scanning). The main steps involved in Conover’s method are as follows:

- *Detection of feature pairs provided by the texture of the artwork:* This method starts by identifying, within the moving image, small regions of the size of the texture features. For this, it uses the modulus maxima of the wavelet transform. The technical image is filtered with low-pass and high-pass filters of various scales n ($L_x H_y(n)$), providing the size of the features to identify. It is also filtered with filters in the opposite order ($H_x L_y(n)$). In this way, two versions of the technical image are obtained: one with emphasised horizontal features, and one with emphasised vertical features. Then, the pixel-by-pixel magnitude of these two new images is computed, to produce the modulus M_n of the original technical image (Fig. 2.18(a)). The local maxima of the modulus is used for determining the initial set of features in the moving image (Fig. 2.18(b)), and *neighbourhoods* around these maxima are defined to avoid spatial clumping and incorrect matching of the features (Fig. 2.18(c)).

After this process has taken place in the moving image, regions of feature points are identified in the fixed image. This is done by using a variant of phase correlation: for both moving and fixed images, the *phase image* is created by implementing a 2D-Fourier transform of the image and then computing the inverse Fourier transform of the result using the phase

information (particularly, the unit magnitude) [86, 11]. These phase images are then used to obtain the maximum of the normalised cross-correlation between them: initially at a down-sampled scale so as to obtain an estimate alignment of the fixed and moving (phase) images. Small search regions in the fixed (phase) image are identified, corresponding to each one of the feature point search regions (*neighbourhoods*) in the moving (phase) image. These small regions, now at full resolution, are then matched by calculating (again) the maximum of the normalised cross-correlation between them. The candidate maximum correlation peak at each search region is selected for identification of the matching feature pairs within both images. What follows is a second step to filter this initial selection of feature point pairs.

- *Filtering the initial set of feature point pairs:* The pairs of feature points identified across the fixed and moving images are filtered based on their distance to a best-fit function plane (Fig. 2.18(d)). Feature points within a distance smaller than a threshold are retained to compute the transformation to apply to the moving image for registration to the fixed image. This threshold is commonly between 0.3 to 1.3 pixels. In Fig. 2.18(e) the best-fit function plane is represented by black points, and matching feature points are represented in blue and red. Blue points are removed from the selection (as they are located at a distance that is higher than the set threshold), despite having good correlation when evaluating the search regions across the moving (phase) image and fixed (phase) image. Red points (located close to the best-fit plane represented by black points) are kept for further computation of the transformation.
- *Generating the registered image:* After filtering, the selection of the feature pairs is kept for computing a geometric transformation to apply to the moving image for registration to the fixed image. For this purpose, a bilinear transformation is used [11, 39]. In the case of the tool that was developed for the registration of multiple adjacent technical images, each moving image is subject to the above mentioned process and registered, one at a time, to the fixed target image. Thus, a mosaic image is formed in an automated piece-wise manner (this is done in one operation, making it possible to upload all of the images to register and then let the software run). In the case that there is overlap between areas of any two consecutive registered images, the pixels in the overlapping region are kept based on their distance to the centre of the corresponding image - the closer they are, the lesser distortion they may present and thus are kept within the final image. This is due to the fact that distortion is greater at the edges, thus these are discarded if redundant information is available due to the overlap between consecutive registered images.

It is important to note that this method, as with our method that is presented in Chapter 3, supports the registration of images differing by a moderate distortion only. To initialise the method,

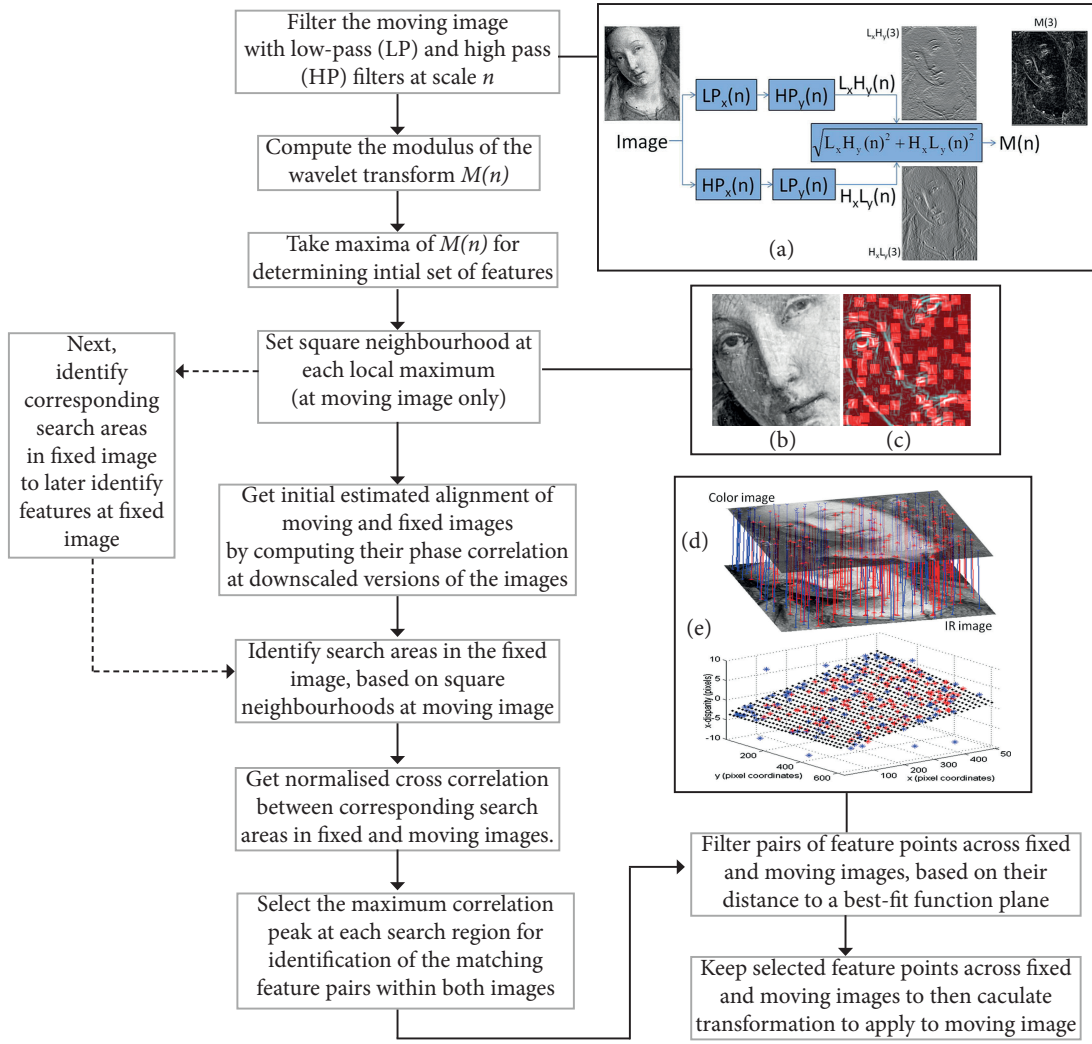


Figure 2.18: This sequence illustrates the stages of the image registration method proposed by Conover et al [11]. Part(a) illustrates the filtering of the moving image with low-pass (LP) and high pass (HP) filters at scale n for the computation of the modulus of the wavelet transform M_n , leading to the detection of texture features. Note that the size of the features identified is in accordance to the scale n , which is $n=3$ for this case. Part(b) shows a detail of the moving image (Infrared image) and Part(c) shows the modulus image with feature points regions identified in red (scale $n=3$). Part (d) illustrates the matching feature point pairs between the color image (top) and infrared image (bottom). Note that this illustration is for visualisation purposes only, as the matching of feature points is done in the phase images of the moving and fixed images. Part (e) illustrates the plane representative of the best-fit function. The points in blue represent feature pair points that have been removed as they are far apart from the best-fit plane, although they have good correlation peaks. The points in red represent feature points that lie at a distance within the threshold t to the best-fit plane, thus they are kept for determining the transformation to apply to the moving image for registration to the fixed image [11]. Images after Conover et al. [11].

Conover indicates that images must be placed at similar orientations [11]. Specifically, before starting the step of feature point identification, images must be rescaled and rotated manually, to

within 5–10 % [11]. The versions of this software that have been tried at the National Gallery also create an output new fixed image that has been scaled to match the moving image. In most cases, this corresponds to creating a downsampled version of the visible image of the artwork to match the resolution of the technical images that are being registered to it. It is also of interest to note that a user interface of the compiled software has also been provided.

2.4 Area-based image registration

Area-based [2] image registration, also referred to as *optimisation-based* [3] or *intensity-based* [87] registration, is commonly used in the field of medical imaging, as such images may visualise soft-tissue organs that often lack distinctive features [2] and thus it may not be possible to implement their registration with feature-based methods. For this reason, suitable registration methods may depend on the relationship between the corresponding intensity values of the images to align, rather than on the extraction of well-defined feature points across such images. Methods of this type are structured in the following manner:

- *Computation of the transformation function:* instead of having dedicated steps for detecting salient features and for then matching these into corresponding feature pairs across the images to align, area-based methods look directly into computing a transformation that is applied to the moving image, and evaluating the resulting overlap of the images. This transformation is iteratively assessed by means of measuring the quality of the overlap, and is then refined so as to progressively improve the resulting alignment. Insights into the group of transformations that are commonly used for this type of approach are covered in 2.4.1.
- *Assessment of the transformation with metrics of alignment:* Depending on the type of images to align, a variety of metrics can be used at each iteration to evaluate the progression of their alignment - where, among others, normalised cross-correlation and mutual information are commonly used. The alignment is then considered to be optimal if it maximises the metric of choice. In the case of registration of multimodal images, the intensity values of the images to align may be *statistically* dependent [2], and thus it is desirable to implement a metric that can operate on this basis. In particular, mutual information is commonly used for registration of multimodal images, since it measures the statistical dependency of intensity values that is typical of this type of images [87], and it is thus our metric of choice (See 2.4.2 for more insights on this topic).
- *Optimisation of the transformation:* At each iteration, an assessment of the overlapping regions is conducted to progressively modify the transformation that is applied to the moving image. These modifications are done through an optimisation algorithm, the selection of which depends on the type of transformation that is applied to the image at each iteration.

One of the most important factors to consider is the amount of parameters to optimise at each turn to avoid increasing the time of computation of the overall registration algorithm (See 2.4.3 for more insights on this topic).

Thus in this work, we focus on using this type of approach for image registration, since the challenges that are commonly presented by multimodal images within the medical sector are similar to those presented by the images derived from MA-XRF scanning of artworks. We compute the mutual information of the overlapping images at each iteration to use as a metric to progressively assess the transformations that are applied to the moving image, and we use the L-BFGS optimiser [88, 89] to modify the transformations that are applied at each turn. We introduce the fundamentals of these steps throughout the sections to follow, and we explain our entire method in detail in Chapter 3.

2.4.1 Computing transformations: Non-rigid transformations

So far we have introduced geometric transformations that allow for modifying the moving image with progressively higher degrees of freedom - with similarity transformations allowing 4 degrees of freedom (see Section 2.3.3), affine transformations allowing 6 degrees of freedom (see 2.3.3), and homography allowing 8 degrees of freedom (see 2.3.3). As mentioned in 2.3.3, the amount of degrees of freedom corresponds to the amount of independent variables (parameters) that can be modified in the transformation that is applied to the image. Ultimately, in all these transformations, straight lines in the original moving image are mapped to straight lines in the registered version of the image. However, if a straight line needs to be mapped to a curved line, we will need to use *non-rigid* transformations, also known as *nonlinear* [1] or *adaptive* transformations [3].

Non-rigid transformations are a group of transformations constituted by various different types of transformations (e.g. spherical and cylindrical warping transformations [3], thin-plate spline [1, 4], B-splines [90], etc.). We are particularly interested in transformations that involve a regular subdivision of the moving image (referred to as *explicit* transformations [3]), as these allow for localised distortions within the moving image. Note that the degrees of freedom of the overall distortion that is applied to the image will be consistent with the amount of internal subdivisions that are set with the transformation to apply.

Two characteristics that are desirable in non-rigid transformations are smoothness and topology preservation [4]. This makes it possible to apply piece-wise smooth transitions in between consecutive regions of the image that would not be possible to achieve with the transformations presented so far. Within the group of explicit transformations, we briefly look into Thin-Plate Splines (TPS [3]) and B-splines [90] transformations. Both implement internal subdivision of the image by means of setting a 2-dimensional grid of control points on top of the image. Each control point is subject to a distortion, dragging the pixels of the images in accordance to the type and field of influence that is set by the transformation to apply.

Thin-Plate splines transformation

Thin-Plate splines (TPS) is a popular transformation used for registration of images with non-linear geometric differences [1], and is used in applications such as *Image Compare* [12]. TPS transformations are part of the *radial basis functions* [2], which can handle local geometric distortions, by means of a mapping function composed of a linear term (an affine transformation) and a weighted sum of radially symmetric logarithmic functions [1]. This is defined [2] as follows:

$$x' = t_x + a_{11}x + a_{12}y + \sum_{k=1}^N c_k g(\mathbf{x}, \mathbf{x}_k) \quad (2.33)$$

and

$$y' = t_y + a_{21}x + a_{22}y + \sum_{k=1}^N c_k g(\mathbf{x}, \mathbf{x}_k) \quad (2.34)$$

where (x', y') are coordinates of the pixel $\mathbf{x}'$ in the transformed moving image I' , $t_x, t_y, a_{11}, a_{12}, a_{21}, a_{22}$ are parameters of the affine transformation involved, c_k are the coefficients corresponding to each control point $\mathbf{x}_k$, and g is the radial term [2, 91], as follows:

$$g(\mathbf{x}, \mathbf{x}_k) = \|\mathbf{x} - \mathbf{x}_k\|^2 \ln(\|\mathbf{x} - \mathbf{x}_k\|). \quad (2.35)$$

It is important to note that radial basis functions are symmetric. Thus, one challenge in this type of transformation is that, when applied to images with asymmetrical local distortions, large registration errors may occur away from the control points - especially if the control point grid is very sparse [3]. This can be overcome by increasing the density of the control point grid, but it may be required to find the parameters of the transformation by solving a large system of equations, at the cost of slowing down the calculation process [3]. An alternative strategy is to look at functions with compact support, such as B-splines, which implement local basis functions keeping deformations (and errors) in the vicinity of each control point [3].

B-splines transformation

B-splines are part of the group of *compactly supported radial basis functions* [3]. As discussed, one of the main differences between functions of this group and TPS, is that the latter lacks the compact support property that is characteristic of B-splines [31]. This allows for the distortions provided by the latter to be more localised, and thus the modifications assigned to each control point in the moving image does not necessarily produce an effect in the overall distortion of the transformed image. This type of transformation is implemented as part of the elastix library [25], where B-splines are used as parameterisation:

$$\mathbf{x}' = \mu_{\mathbf{x}_k, \mathbf{p}_k, \sigma}(\mathbf{x}) = \mathbf{x} + \sum_{\mathbf{x}_k \in N_x} \mathbf{p}_k \beta^3\left(\frac{\mathbf{x} - \mathbf{x}_k}{\sigma}\right) \quad (2.36)$$

where $\mathbf{x}_k$ are the control points, $\beta^3(\mathbf{x})$ is the cubic polynomial B-spline [23, 92, 93] that determines the area of influence of $\mathbf{x}_k$ (see Eq. 2.37 and Fig. 2.19), $\mathbf{p}_k$ are the B-spline coefficient vectors (determining the direction of displacement of each control point $\mathbf{x}_k$), σ is the B-spline control point grid spacing, and N_x is the set of the control points within the compact support of the B-spline at $\mathbf{x}$ [31]. The cubic spline is defined as follows:

$$\beta^3(\mathbf{x}) = \begin{cases} \frac{2}{3} - |\mathbf{x}|^2 + \frac{|\mathbf{x}|^3}{2} & 0 \leq |\mathbf{x}| < 1 \\ \frac{(2-|\mathbf{x}|)^3}{6} & 1 \leq |\mathbf{x}| < 2 \\ 0 & 2 \leq |\mathbf{x}| \end{cases} \quad (2.37)$$

Note that the *compact support* is directly related to the limited influence of $\mathbf{x}_k$ on its neighbouring pixels. This results in $|N_x|$ being small, and thus the transformation of a pixel $\mathbf{x}$ is to be computed from only a few surrounding control points (Fig. 2.20). B-splines transformation proved to be very efficient in terms of computation time and precision of results, thus becoming the transformation of choice in the final steps of the area-based method that is presented in Section 3. The implementation of this type of transformation is further explored in Section 3.5, with a detailed overview of the setting up of the control point grid (see Section 3.5.1) and the parameters resulting from this setup (see Section 3.5.2).

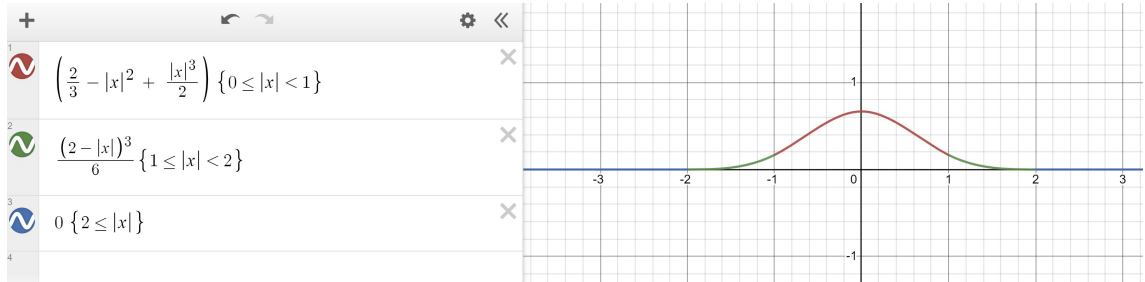


Figure 2.19: Area of influence of control points $\mathbf{x}_k$. Their displacements have a dragging effect in the pixels that are in proximity to each control point, this effect being maximum when the distance to the neighbouring pixels is 0. This is determined by the function 2.36, with particular importance of the one illustrated here, the cubic function 2.37.

Due to the high number of parameters to resolve at each iteration, it is customary that registration with non-rigid transformations is conducted only after the fixed and moving images have undergone an initial affine alignment. This approach is thus followed in the area-based method that is presented in detail in Chapter 3.

2.4.2 Metrics of alignment - An overview

Whilst in feature-based methods it is possible to consider the distance between the corresponding feature pairs as a measure of success of the resulting registration, area-based methods rely on the statistical assessment of the overlapping images to thus assess the quality of the registration

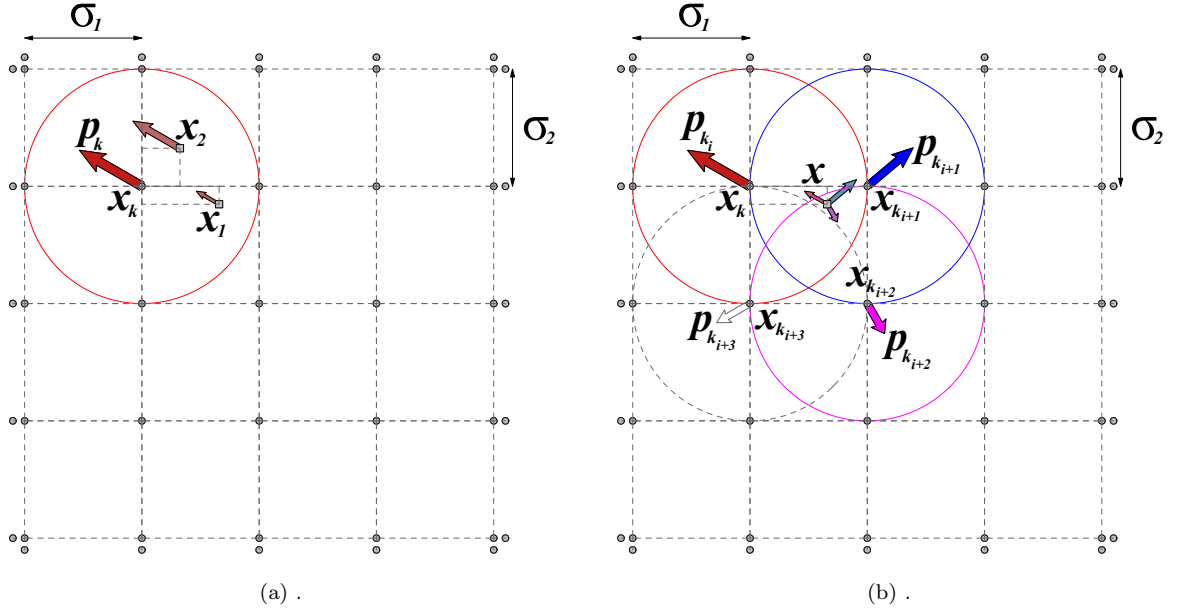


Figure 2.20: Control point grid assigned to a 2D image, of resolution 4 rows x 4 columns, at regular spacings σ_1 and σ_2 . Note that σ_1 and σ_2 can differ in pixel length. For this grid resolution, rows and columns are determined by 5 control points, plus one control point just outside the image at either end of the grid. Thus, we will have 7 control points for each dimension. At each control point $\mathbf{x}_k$, a B-spline coefficient vector $\mathbf{p}_k$ exerts a drag on neighbouring pixels $\mathbf{x}$ that are located within the area of influence of each control point. For different $\mathbf{x}_k$ and $\mathbf{p}_k$, areas of influence are denoted here with circles of corresponding color. These follow equation 2.37, illustrated in Fig. 2.19. The closer pixel $\mathbf{x}$ is located to control point $\mathbf{x}_k$, the greater the effect of $\mathbf{p}_k$ and so is the drag that pixel $\mathbf{x}$ is subject to. The resulting displacement for each pixel $\mathbf{x}$ follows equation 2.36. In part (a), pixel $\mathbf{x}_1$ is further away from control point $\mathbf{x}_k$ than pixel $\mathbf{x}_2$, and thus $\mathbf{x}_1$ is subject to a lesser drag than $\mathbf{x}_2$. In part (b), pixel $\mathbf{x}$ is located within the area of influence of control points $\mathbf{x}_{ki}$, $\mathbf{x}_{ki+1}$ and $\mathbf{x}_{ki+2}$, but lies outside the area of influence of $\mathbf{x}_{ki+3}$. Thus, pixel $\mathbf{x}$ is subject to the drag provoked by the first three control points only, and these constitute the compact support N_X for pixel $\mathbf{x}$.

obtained. Area-based methods are supported by a variety of similarity metrics, such as *gradient difference*, *gradient correlation*, *pattern intensity*, *entropy of the images* [87], *mean squared difference* [31], etc). Each of these metrics assume different relationships between the intensity of the images to align, and thus not all of them are suitable for registration of different image modalities. In this section, we briefly look at commonly used metrics for multimodal image registration: normalised cross-correlation and mutual information.

In the study conducted by Penney et al. [87] for the comparison for similarity metrics in the context of multimodal registration of medical images, it is observed that mutual information "does not assume a linear relationship between the intensities of the images" to align [87]. Note that the images that were tested in their study had been produced in a way that allowed for the corresponding intensities to maintain a relationship close to linear. In their study, it is observed that the other metrics tested (among them, normalised cross-correlation) seem to assume such a linear relationship of the intensities. The relatively poor performance of mutual information that

was obtained is likely attributable to this assumption of a linear relationship of the intensities of the images in that study.

However, this is a characteristic that may favour our case: the fact that mutual information does not assume a linear relationship of the intensities of the images to align allows for the resilience of this metric when it comes to assessing the registration of images with discrepancies in the features they may show (such is the case of images obtained with MA-XRF scanning, which may often show different sub-surface features than those apparent in the target visible image, as mentioned in Section 2.1).

Normalised cross-correlation as a metric for multimodal image registration

The definition of normalised cross-correlation (NCC) that is most commonly used for area-based methods [2, 31, 94] is as follows:

$$NCC(\mu, I_F, I_M) = \frac{\sum_{\mathbf{x}_i \in \Omega_F} (I_F(\mathbf{x}_i) - \bar{I}_F) (I_M(\mu(\mathbf{x}_i)) - \bar{I}_M)}{\sqrt{\sum_{\mathbf{x}_i \in \Omega_F} (I_F(\mathbf{x}_i) - \bar{I}_F)^2 \sum_{\mathbf{x}_i \in \Omega_F} (I_M(\mu(\mathbf{x}_i)) - \bar{I}_M)^2}} \quad (2.38)$$

with the average of the grey-values at the fixed image $\bar{I}_F = \frac{1}{\Omega_F} \sum_{\mathbf{x}_i \in \Omega_F} I_F(\mathbf{x}_i)$

and with the average of the grey-values at the moving image $\bar{I}_M = \frac{1}{\Omega_F} \sum_{\mathbf{x}_i \in \Omega_F} I_M(\mu(\mathbf{x}_i))$

where I_F is the fixed image, I_M is the moving image, and μ is the transformation that is applied to each pixel $\mathbf{x}_i$ of I_M mapped in the domain of the fixed image Ω_F .

Note that NCC is also implemented throughout the feature-based method of Conover et al. [11], particularly in the steps concerning feature matching 2.3.4. NCC is commonly used in area-based methods for registration of multimodal images, although it is noted that this metric yields best results for images where their intensity values appear to be in a linear (or close to linear) dependency [94, 87], making this metric sub-optimal for cases where the intensity values of the images to align appear to be related in a different manner.

Mutual information as a metric for multimodal image registration

In this work, we focus on using the mutual information (MI) of images obtained with MA-XRF scanning of artworks as the metric of alignment for their registration with a novel, automated area-based method that is presented in detail in Chapter 3.

Area-based methods based on MI represent the leading technique in registration of multimodal images [2]. They are widely used for image registration in the medical imaging sector [24] since their introduction by Viola and Wells [95, 96] and Collignon et al. [97]. Originating from information theory, MI is a measure of statistical dependency between two images. An important property of MI is its generality, since it does not assume any relationship between the image intensities [94]. Following the study conducted by Penney et al. [87], MI "assumes that the co-occurrence of the most probable pixel values in the two images is maximized at registration", making it most suitable

for the registration of images obtained with different modalities [87, 94].

The MI between the overlapping areas of images A and B can be calculated in the following way [98, 2]:

$$MI(A, B) = H(B) - H(B|A) = H(A) + H(B) - H(A, B) \quad (2.39)$$

where $H(A) = -E_A(\log(P(A)))$ represents the entropy of the image A (with images treated as random variables), and $P(A)$ is the probability distribution of the intensity values corresponding to the image A . Thus, for the overlapping images A and B , the interpretation of this form is that their MI corresponds to the sum of their entropy values minus their joint entropy $H(A, B)$, at their particular relative positions of overlapping. Changing their relative positions (and thus the quality of their overlapping), will have an impact in the value of their joint entropy $H(A, B)$ and consequentially, change their resulting value of MI. Note that the MI of the overlapping images will be highest when their joint entropy $H(A, B)$ is minimised. Area-based methods seek the maximization of MI as a metric for alignment, assuming that the best alignment of the images is obtained at the peak value of this metric (Fig. 2.21). Intuitively, a greater value of MI indicates that A and B are more mutually dependent which in turn indicates a better alignment of these images.

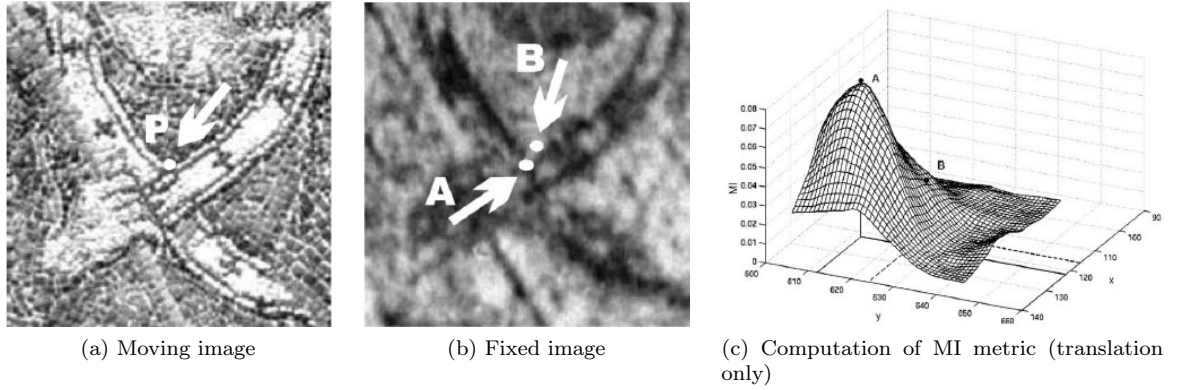


Figure 2.21: Mutual Information (MI). Part(a) shows point P as a reference within the moving image, and part(b) shows the points A and B, the latter indicates the false matching position selected previously by the human operator. Part(c) shows the MI metrics computed in the registration of the moving and fixed images shown in part(a) and (b) respectively, where maximum of MI shows the correct matching position (point A, which is the corresponding position for point P). Images after Zitova and Flusser [2].

In the method presented in Chapter 3, the MI of the images to align is calculated following the formulation provided by the elastix software [25, 31], an open source library for multimodal image registration commonly used in the medical industry. This library uses a form of MI that was introduced by Mattes [23], estimating continuous image distributions using Parzen windows [95]. This metric is known today as MattesMutualInformation [24]. This form is explained as follows:

For two images A and B , mutual information MI can be defined as:

$$MI(A, B) = \sum_{a,b} p(a, b) \log \frac{p(a, b)}{p(a)p(b)} \quad (2.40)$$

This form can be interpreted as the distance between the joint distribution of the pixel intensity values of the images $p(a, b)$ and the joint distribution in the case of independence of the images, $p(a)p(b)$ [24].

The probabilities $p(a)$ and $p(b)$ are calculated by creating a count of the pixel intensity values a and b occurring at each image A and B correspondingly. The probability $p(a, b)$ is calculated by counting the occurrence of a pair of pixel values at corresponding locations across both images. Note that our algorithm is designed to work with grayscale images.

Thus, MI becomes a metric of dependence between the two images that are overlapped, and thus it is assumed that there is maximal dependence between the pixel intensity values of the images when their alignment is correct [24]. The misalignment of the images corresponds to the decrease of this metric. Our algorithm is designed to identify the transformation that obtains the highest absolute value of MI when applied to the moving image for its alignment to the target fixed image.

2.4.3 Optimisers

As mentioned at the beginning of this section, area-based methods may assess the transformation that is applied to the moving image at each iteration by means of computing a metric of the resulting alignment at each turn. The purpose of the method is to converge to an optimal alignment of the images after a number of iterations. Thus at each turn, the transformation may be progressively refined to improve the alignment gradually.

These refinements or modifications are done through an *optimiser*, which is an optimisation algorithm that takes into consideration the parameters of the transformation to evaluate and the metric of alignment that is obtained at each turn.

Area-based methods benefit from a great variety of optimisers. Among others, the following classes of optimisers are commonly used in the medical imaging sector: gradient descend, conjugate descend, Quasi-Newton, and stochastic gradient descent [94]. From this selection, the Quasi-Newton methods are designed to collect information from previous iterations and make use of it to ensure successful convergence. Particularly, methods from this class estimate the inverse of the Hessian matrix ⁴ to define the direction of the search. Within this class, our choice of optimiser is the *Limited-memory Broyden-Fletcher-Goldfarb-Shanno* algorithm (L-BFGS), since it is particularly well suited for optimization problems with many variables [88], as well as being considered to be more efficient than other methods of this class [94]. A study by Kumar [89]

⁴ Hessian matrix is a square matrix containing the second order derivatives, and it can be used for describing the local curvature of a function of many variables [99].

regarding a minimization example for the Rosenbrock function ⁵, shows that the L-BFGS algorithm, being a second-derivative method, converges much faster (and after significantly lesser amount of iterations) than the first-derivative based gradient descent method. We review the implementation of L-BFGS as an optimiser in the presentation of our area-based method in Section 3.5.3.

2.5 Learning Based Methods

As mentioned in the introduction to this chapter (see 2.1), methods for image registration based on deep learning [32, 33] have started to emerge. Methods of this type either follow the strategy of feature-based methods, by learning corresponding pairs of feature points across the images to align [32, 34], or, in the spirit of area-based methods and subject to the type of images in the training set, they may learn a non-rigid transformation to apply to the moving image [35, 38, 101]. Due to the challenges involved in attaining registration of images obtained with MA-XRF scanning to the visible image of the same artwork (see Section 2.2.4), this section focuses on methods based on deep learning that explore area-based strategies for multimodal image registration. We particularly look at methods that learn the deformation vector field to be applied to the moving image following training with a collection of image-pairs of different intensity levels, where each pair is composed of the fixed image and the moving image [36, 37]. Note that methods that follow this approach need images to be segmented to achieve successful learning of the deformation fields [36, 101], and this is something that may not always be possible with technical images of artworks. Another challenge that these methods may present is the need to train the models for a specific size of image pairs, where both fixed and moving images need to be of the same size to train the network. This aspect might be unpractical as datasets obtained from MA-XRF scanning frequently differ in size (including those that are consecutively captured from different areas of the same artwork), resulting in the need for a bespoke network for each image-size, for each image stack to register to the visible image of the artwork.

In this section, we review two popular methods of this type: Voxelmorph [36] (the one that set up the standard) and Synthmorph [37] (making improvements to the standard, and bringing the model closer to what is needed for testing images obtained from MA-XRF scanning of artworks).

2.5.1 Voxelmorph

VoxelMorph [36] is an unsupervised learning-based registration method developed for non-rigid alignment of medical images. We note that traditional area-based methods (i.e. non-learning based methods) optimize a non-rigid transformation that is tailored to a particular pair of images. Conventional area-based methods based on learning models instead rely on ground truth deformation fields or segmentation maps of the images to align [35, 102, 103].

⁵ The Rosenbrock function is a non-convex function that is commonly used to assess the performance of optimization algorithms [100].

The novel approach presented by Voxelmorph consists on a model that learns a registration function from a collection of images of a given class, and once the model is trained, it assigns a non-rigid deformation field to a new input image pair (i.e. an image pair consisting of a moving and a fixed image). The deformation field is then applied to the moving image for its alignment to the fixed image. Both fixed and moving images are of the same size as that of the images in the training and validation datasets. The model consists of a convolutional neural network (CNN) that learns the function by training on a dataset of images of a given anatomical class (e.g. images of coronal slices of brains). Once trained, the model can receive a new pair of images of the same class, and provide a deformation field for their registration in return.

The method assumes two important pre-processing steps that are customary for area-based methods, as follows:

- Both images that are constituent of the image-pair, moving image m and fixed image f , are assumed to be single-channel, grayscale images.
- Both images m and f are affinely aligned prior to being submitted as input to the model for non-rigid registration.

Then, the method works as follows:

- Using a CNN, a function $g_{\theta}(f, m) = \mathbf{u}$ is modelled, where θ are the parameters of the network and $\mathbf{u}$ is the displacement field, after which the deformation field ϕ is computed (Fig 2.22(a)).
- The moving image m is warped to ϕ using a spatial transformation function. The similarity between $m \circ \phi$ and f is assessed together with the smoothness of the deformation field, using an unsupervised loss function based on the local cross-correlation of f and $m \circ \phi$. Note that the authors recommend the use of mutual information in the loss function if the model is intended for registration of multimodal images [36].
- A second loss function is implemented for leveraging anatomical segmentations of the input image pair (Fig 2.22(b)). This loss function is based on the *Dice score*, which makes it possible to assess the overlapping region of two binary images or boundaries [104] (thus the need to perform this second loss function on segmented images). The authors acknowledge significant improvements to the registrations obtained by including this second loss function in the model [36].

The results obtained in the studies by Balakrishnan et al. [36, 105] are very positive for multimodal images, particularly when implementing mutual information as part of the loss functions involved in this model. It is also important to note that, once the model is trained, the speed of registration of a new input image pair is remarkable. However, this method is designed to work with images of the same class (i.e. depicting the same objects) and of a specific size (i.e. the model needs to be trained with images of the same size that will later receive as input for registration).

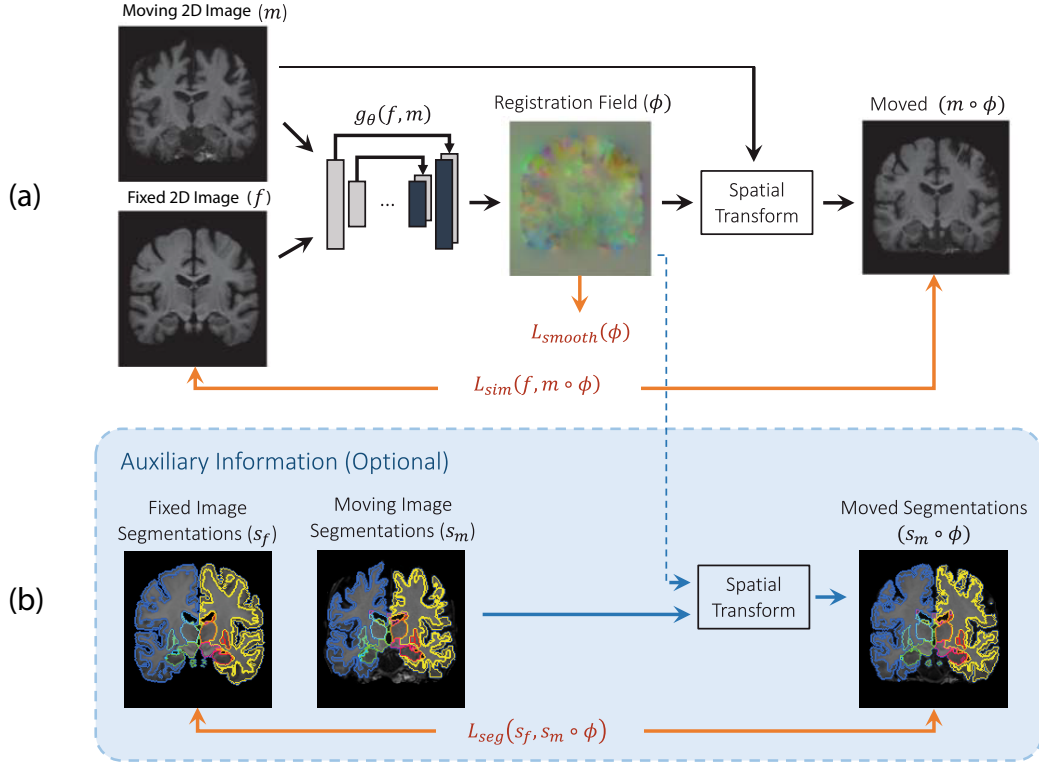


Figure 2.22: Diagram of the Voxelmorph algorithm. Part(a) shows the image pair (m, f) as input to the CNN model $g_\theta(f, m)$. During training, this model outputs a deformation field ϕ from where a transformation is produced and applied to the moving image m . This results in obtaining the transformed image $m \circ \phi$, that is then assessed against the fixed image f by means of the loss function $\mathcal{L}_{sim}(f, m \circ \phi)$. Part(b) shows the process by which the second loss function $\mathcal{L}_{seg}(s_f, s_m \circ \phi)$ contributes to the refinement of the deformation field ϕ by assessing ϕ against the overlapping segmentations of the segmentation map of the fixed image s_f and the segmentation map of the moving image once transformed $s_m \circ \phi$. Note that once trained, the model can output the transformed image $m \circ \phi$, that is then fully registered to the fixed image f . Images after Balakrishnan et al. [36].

In the example of this study, the model was trained with 3731 brain images obtained with MRI scanning, after resampling and cropping them to $160 \times 192 \times 224$. This poses a challenge to our case, as in art investigation, each image pair is unique, and it would not be possible to generate such a large collection of image pairs of the same object to use as the training dataset for the registration of a new pair of images of the same pictorial composition. For this reason, in the next example we look into the learning-based method *Synthmorph* [37]. Building from Voxelmorph, this method works with a training dataset composed of synthetic images that resemble random geometric shapes, and thus presents itself as a more versatile approach for testing different types of images.

2.5.2 Synthmorph

Building from the principles of the VoxelMorph algorithm (described in Section 2.5.1), Synthmorph [37] is an unsupervised learning-based registration method developed for non-rigid alignment of images of different levels of intensity. In this case, instead of training with a dataset of images of the same anatomical shape, the model consists of a convolutional neural network (CNN) that learns a function after training on a dataset of *synthetically* produced image pairs. These images are resemblant of abstract geometric random shapes, and are generated from noise images sampled from a standard distribution (Fig 2.23(a,b,p)). This novel training approach allows for the fact that, once the model is trained, it can output successful registration of input image pairs that do not necessarily belong to any particular class of images.

As with the Voxelmorph approach, both fixed and moving images are of the same size as that of the images in the training and validation datasets.

This method works as follows:

- Firstly, a dataset of synthetically generated images of random shapes is created for training. These images are called *labels* (Fig 2.23(a)), and are created as follows: noise images $p_j (j \in 1, 2, \dots, J)$ are sampled from a standard distribution (Fig 2.23(e,f)). A smooth deformation ϕ is then applied to these images, obtaining images $\hat{p}_j = p_j \circ \phi$ (Fig 2.23(g,h)). From these images, a label s_k is created by mapping onto each location k of s , the highest intensity value across the images $\hat{p}_j$ (Fig 2.23(i)). Next, each of these labels s are subject to smooth transformations so as to obtain the *segmentation* maps s_m and s_f (Fig 2.23(k,l)). They serve as the image pair, constituted by a moving image and a fixed image. By converting these to grayscale and altering their intensities with a seed $\hat{z}$, the final training images are obtained (Fig 2.23(m,n)). These images, together with their corresponding labels, compose the training dataset for the model, that is described next. Note that these steps are the ones belonging to the *Image sampling model* connecting parts(a) and (b) at Fig 2.23.
- Using a CNN, a function $h_\theta(m, f) = \mathbf{u}$ is modelled, where θ are the parameters of the network and $\mathbf{u}$ is the displacement field, after which the deformation field ϕ is computed. Note that the model is trained with the grayscale images only (Fig 2.23(b)).
- The output of the model is the deformation field ϕ_v (with v a diffeomorphism⁶, see Fig 2.23(c)). Note that this deformation field is then applied to the moving *label* image s_m , finally obtaining the warped image $s_m \circ \phi_v$ (Fig 2.23(d)). Note that due to the fact that the model is trained with synthesized label images that are segmented by design, the loss function is composed of a similarity function that measures overlap rather than correlation of image intensities. Thus, in this model the loss function is made of a "soft" version of the Dice metric [107] $\mathcal{L}_{dis}$ and a regulariser $\mathcal{L}_{reg}$ to ensure smoothness of the deformation field ϕ_v [37].

⁶ diffeomorphism is an invertible, differentiable mapping function that is used for non-rigid image registration[106].

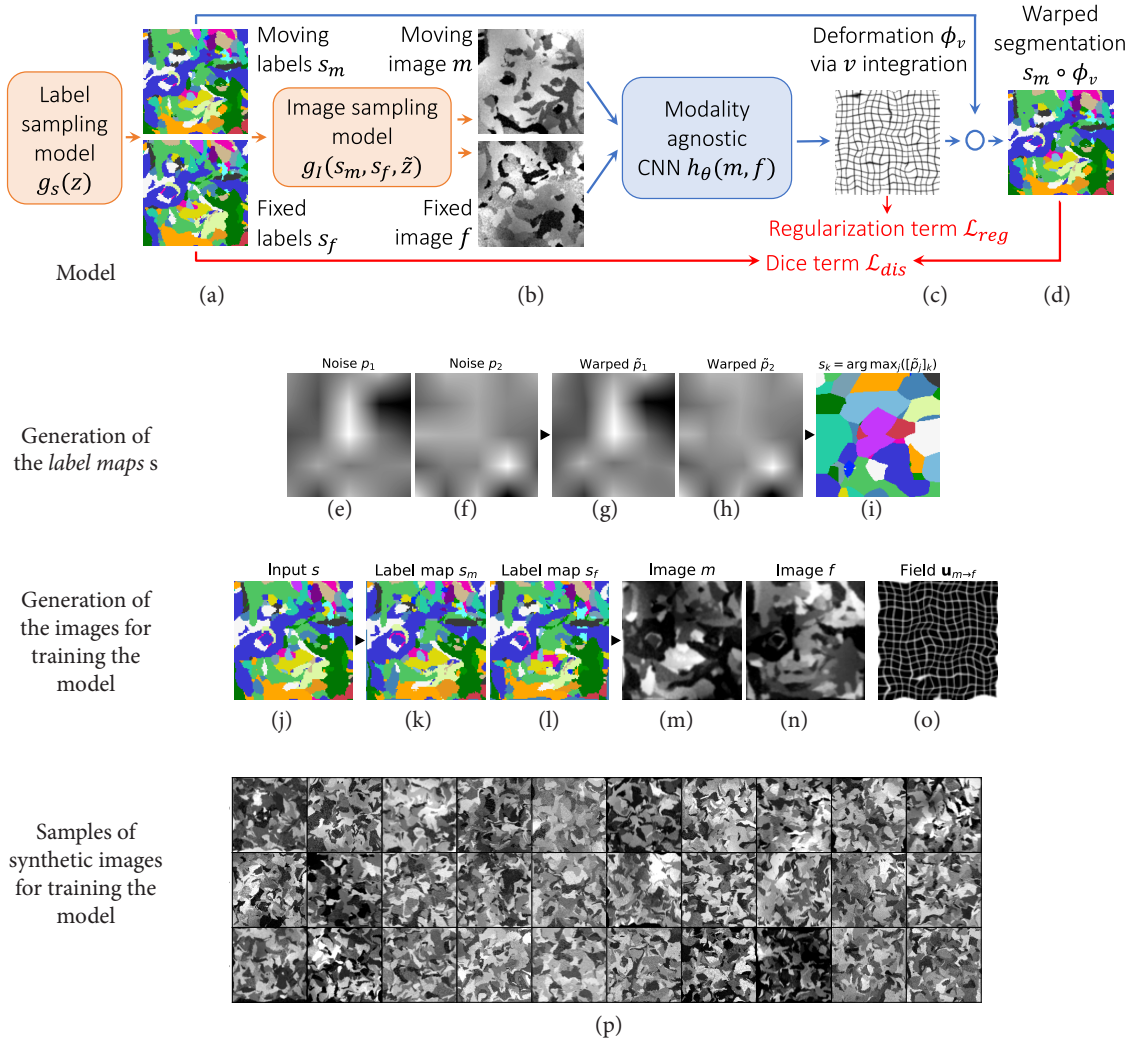


Figure 2.23: This sequence describes the steps of Synthmorph. Parts(a,b,c,d) show the algorithm in full, from receiving the image pair (labels s_f and s_m , part(a)) as input to the model for generating the moving and fixed images (m, f , in grayscale, part(b)), from which the deformation field ϕ_v is obtained for computing the transformation to apply. Note that this transformation is then applied to the image label s_m (part(c)) thus obtaining the transformed image label $s_m \circ \phi_v$ (part(d)). This is then assessed against the image label s_f by means of the loss functions $\mathcal{L}_{reg}$ and $\mathcal{L}_{dis}$. Parts(e,f,g,h,i) show the steps involved in generating the synthetic images that are treated next to create the dataset for training the model, from the original noise images obtained by sampling (parts (e,f)) to the final segmented map s_k (part(i)). Parts(j,k,l,m,n,o) show the following steps, in particular subjecting the images s_k (part(j)) to smooth deformations for the generation of the labels s_m and s_f (parts(k,l)). In turn, these images are subject to grayscale conversion as well as to variations in intensity values, thus obtaining the corresponding grayscale images (m, f) (parts(m,n)). It is from these grayscale images that the transformation to apply to the moving label image s_m is computed (part(o)). Part(p) shows the variety of image pairs (m, f) that compose a typical training dataset. Note that each of these are accompanied by its corresponding label map (as seen in parts (k,l)). Images after Hoffman et al. [37].

This work expands on the previous method of Voxelmorph, not only by developing a model that learns a registration function from training with a collection of images, but also by introducing the option of training with synthetic images that resemble random geometric shapes. This aspect allows the model to work with images belonging to classes other than that of the images that are constituent of the dataset with which the model is initially trained. As with Voxelmorph, once the model of this method is trained, the speed of registration is very fast for every new input image pair. It is also important to note that, since the synthetic images that compose the training dataset are treated so that they acquire different levels of intensity (Fig 2.23(b)), this model once trained is deemed to be *modality agnostic* [37]. In the study conducted by Hoffmann et al. [37], this model is trained with the synthetic images as described in this section, and it is then tested for registration of unseen input image pairs of anatomical shapes with dissimilar levels of intensity, yielding satisfactory registration results.

Thus, the versatility presented by the algorithm of Synthmorph was the main motivation to test this method for the registration of technical images of artworks, namely of images obtained with MA-XRF scanning.

2.5.3 Experiments of image registration using Synthmorph

In this section, results from using the Synthmorph algorithm [37] for registration of technical images of artworks are presented. In particular, this method was implemented for registration of technical images obtained by MA-XRF scanning of the *Portrait of Jane Boughton, née Coningsby* (AGOID.29371, The Thomson Collection at the Art Gallery of Ontario), to the visible image of the same artwork (Note that a series of experiments conducted with this dataset, using our area-based method, is presented in Chapter 4, together with more details about this painting and details about the data acquisition). For the experiments presented here, a pre-trained version of the Synthmorph model was used [108, 109], which is publicly available online by the authors of the Synthmorph algorithm. This model is pre-trained with grayscale image pairs of moving and fixed images of random geometric shapes as the ones shown in Fig. 2.23(p), following the characteristics of the standard model described at Section 2.5.2.

In this case, the Synthmorph model is pre-trained with 2D images of 256 x 256 pixels, and thus designed to operate with images of that size. Thus for this purpose, the technical images of the artwork (of original size of 474 x 470 pixels) were cropped to this size (as per red outline in Fig. 2.24(b)), selecting a region close to the bottom left corner of the full image, whilst maintaining their original resolution (Fig. 2.24(c)). The visible image was converted to grayscale and down-sampled to match the resolution of the technical images, and cropped to size 256 x 256 pixels in the same manner (Fig. 2.24(a,b,c)). The steps of implementing registration with the Synthmorph model are shown in detail in Fig. 2.24.

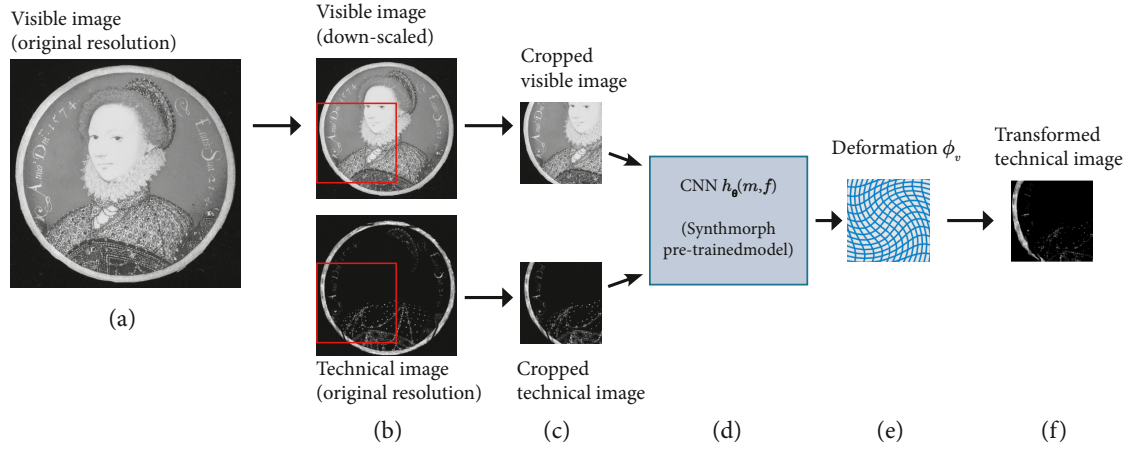


Figure 2.24: This sequence illustrates the steps for the experiments conducted with the Synthmorph algorithm. Part(a) shows the grayscale version of visible image of the artwork, at its original resolution (3098 x 3073 pixels). This image is downsampled to match the native resolution of the technical image obtained with MA-XRF scanning of the artwork (here the *Portrait of Jane Boughton, née Coningsby*), of 474 x 470 pixels (part(b)). These two images, the visible serving as fixed image, and the technical serving as moving image (here the elemental map of gold), are then cropped (as per red outline in image) to match the working image size of the pre-trained model, which is 256 x 256 pixels. These images constitute the input image pair (part(c)). Upon receiving this image pair, the pre-trained model (part(d)) outputs a deformation field ϕ_v (part(e)), which is applied to the technical moving image for its registration to the visible image of the artwork.

Following the setup described in Fig 2.24, we show the registration results of the cropped version of the MA-XRF elemental distribution map of gold (Fig 2.25(a)) and the registration results of the image obtained by combining elemental distribution maps of gold and lead (Fig 2.25(e)). Both served as the technical moving images to be registered to the corresponding visible image of the artwork (Fig 2.25(b,f)). The first technical moving image belongs to the set of original elemental distribution maps obtained from MA-XRF scanning of *Portrait of Jane Boughton, née Coningsby*, shown in Fig. 3.1. The second image is produced based on this set, by combining two significant elemental distribution maps for this artwork, with the aim of providing more pictorial information to the registration model, in the expectation of yielding better registration results. The transformed moving images are shown in Fig 2.25(c,g). Note that arrows in red show the areas where the transformations applied fail to ensure a satisfactory registration to the visible image of the artwork. Fig 2.25(d,h) show the deformation field (x-axis only) that is applied to each corresponding technical moving image.

Valuable insights were gained by implementing this model to the task of achieving registration of multimodal technical images of artworks: although the results obtained from these experiments were not optimal, the differences observed between the deformation fields shown in Fig. 2.25(d,h) highlight the importance of working with methods that ensure consensus among the images ob-

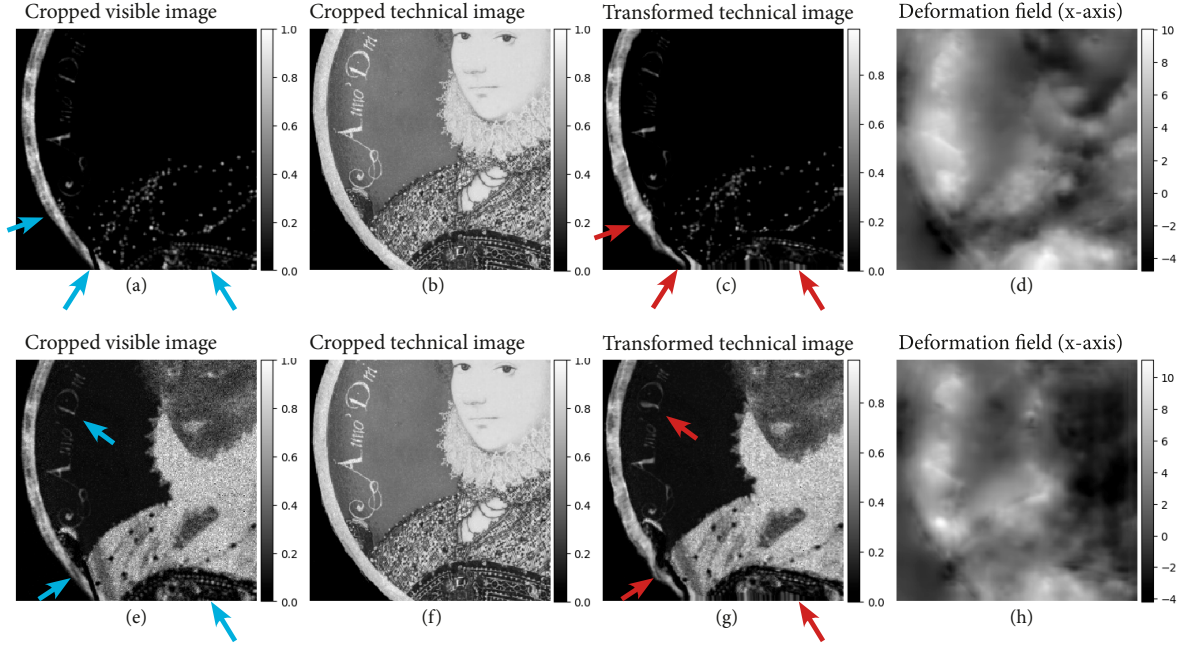


Figure 2.25: This sequence illustrates the results of two registrations conducted with the Synthmorph algorithm: the one for the cropped version of the elemental distribution map of gold (part(a)), and the one for the image of the combined elemental distribution maps of gold and lead (part(e)). Each of these images, together with the corresponding cropped version of the visible image of the artwork (parts (b,f)) constitute the input image pair to input the Synthmorph pre-trained model, which in turn outputs the transformed technical image for each pair (parts (c,g)). Arrows in blue point to the equivalent areas in the starting image to those marked with red arrows as a problem in the transformed image (i.e. where the transformations applied fail to ensure a satisfactory registration to the visible image of the artwork). Parts (d,h) show the deformation field (x-axis only) that is applied to each corresponding technical moving image. Note that the difference between these deformation fields is a call-to-action to ensure consensus among the images that constitute the image stack that is to be registered to the corresponding fixed image.

tained from the same MA-XRF scanning when registering them to the visible image. Note that the deformation field, from where a transformation is obtained and applied to each corresponding technical moving image, is supposed to be the same for all the images obtained from the same MA-XRF scanning, as they are all aligned by construction (see 2.2).

Based on the registration experiments described above and the insights gained from these, we envision the following potential improvements to the Synthmorph algorithm:

- Allow the model to operate with different image sizes, so that the model can receive input image pairs of sizes other than that of the images within its training dataset, as well as enabling the images constituting the image pair to be of different sizes.
- Allow the model to operate with more than one moving image at a time when registering to the fixed image (i.e. allowing to simultaneously assess multiple images, in an orderly arrangement in a stack, so as to register them to the same area of the target fixed image). This is a desirable solution for our case, where multiple images are obtained from the same MA-XRF scanning of the artwork. This would allow consensus to be obtained among the images that constitute the image stack, and to register them with the identified best transformation resulting from the assessment of the image stack against the target fixed image.

Notwithstanding the discrepancies between the deformation fields obtained in Fig. 2.25, the results of these simulations show the Synthmorph model to be remarkably versatile. We envision that, if the above-mentioned short-comings are addressed, it is highly likely that this approach will yield highly satisfactory results.

2.6 Discussion

At the time of submitting this thesis, we understand that there are currently no automated methods designed for the registration of a stack of images with consensus among the constituent images of the same stack. In addition to this, none of the methods currently in use for image registration in the field of art investigation make use of the mutual information (MI) of the images to align as a metric of their alignment. We note that the team of the Restoration of the Ghent Altarpiece uses a method for image registration that takes into consideration the intensity values of the pixels of the images to align. Their approach, however, differs in the aspect that the metric in their method looks for the minimisation of an energy functional that incorporates a dissimilarity measure that is calculated with masks at corresponding pixel neighbourhoods within the images to align, and a measure associated with the curl and divergence of the deformation field [110]. Their method implements this metric for registering *crude crack maps* obtained from the technical images to align (i.e. the X-ray, the visible and the infrared reflectogram images of the artwork to study) [41], and this result is then used for registering the technical images of interest [40]. We also note the area-based method developed by the Conservation and Science Department of the Rijksmuseum [111]. In a similar manner to the method just described above, this approach creates filters that are representative of the images to align. These filters are overlapped, and offsets along the x and y axis are evaluated using normalised cross-correlation as a metric of their alignment. The transformation that yields the best alignment of the filter representative of the moving image is then

utilised for the registration of the original moving image [111]. Apart from these exceptions, the broad majority of methods for multimodal image registration developed for art investigation rely on the presence of salient, well-defined features across the images to align [12, 78, 11, 77, 81, 79, 80].

As mentioned in Section 1.1, MI is commonly used as a metric in area-based methods aimed to provide registration of multimodal images, especially in the medical imaging sector [23, 24, 25]. We borrow this approach to present a novel, automated registration method for multimodal images of artworks, initially designed for the registration of images obtained from the datacube derived from MA-XRF scanning of artworks. These images, considered to be aligned by construction, are arranged into an layered image stack that is treated as the moving image. Our method is designed to identify a unique best transformation to apply to the image stack for its registration to the visible image of the same artwork. The transformation is identified with consensus among all images in the stack. We note that, whilst all images in the same stack (images that constitute the layers of a moving image) are assumed to be of the same size, there are no restrictions on image size of different moving images to register (i.e. one fixed image can act as the target fixed image for the registration of moving images of different sizes). We present this method in the following chapter.

2.7 Summary of this chapter

In this chapter we have seen the core topics that are at the basis of the automated area-based method for image registration of multimodal technical images of artworks - the main subject of this thesis. We have reviewed the principles of collecting data with MA-XRF scanning, the processes involved in producing technical images from this data, and the experimental details regarding the MA-XRF scanning of the artworks that are studied in this work. This was followed by a review of feature-based methods, together with a presentation of the geometric transformations that are commonly associated with this type of methods. A review of area-based image registration methods and an overview of the technical aspects involved in these methods, such as non-rigid transformations, metrics of alignment, and optimisers was then presented. What followed next is a brief review of state-of-the-arts learning-based methods, together with a review of the experiments as part of this study with this type of approach. This chapter concluded with the insights obtained from these experiments, which in turn helped shape the area-based image registration method that is presented in the next chapter.

Chapter 3

An automated, area-based image registration method

This chapter delves into the workflow behind the automated area-based method for registration of multimodal technical images of artworks, which is the main subject of this thesis. In particular, this chapter addresses the implementation details of this method - from setting up the *moving* image (that is, the image or images that need to be distorted by means of a geometrical transformation so as to align to a target *fixed* image), to the steps involved along the process of identifying the best transformation to apply to the moving image. Consisting of 4 steps that we call *searches*, these are explained in detail along with the transformations that are consecutively applied at each search, and examples of their impact along the method. The registrations produced following this workflow are presented in Section 4.

3.1 Introduction

In this chapter, an automated area-based method for multimodal registration of technical 2D images from artworks is presented. This method, originally introduced in [28], can work with a stack of images which is treated as the moving image in need of registration to a target image. The stack can be made of the elemental distribution maps obtained from the same MA-XRF scan of an artwork (as discussed in Section 2.2), but it can also be made of images obtained with other imaging techniques¹. Regardless of the origin of the images in the stack, it is assumed that they are all of the same size, as well as that they are all aligned relative to each other, as, in the case of MA-XRF scans, they are all generated simultaneously during the same instrument data capture.

¹ Note that certain imaging techniques generate a single image output (e.g. visible light photography, X-ray, Infrared luminescence - these are part of the *broad band* imaging techniques, which produce single images from a range of wavelengths of radiation), whereas *spectroscopic* imaging techniques (i.e. methods that capture much more restricted wavelength ranges in a separate manner, e.g. using filters) produce multiple aligned images. Examples of the latter are MA-XRF and reflectance imaging spectroscopy (RIS). In this case, the images can be assumed to be arranged into a stack of images that may be registered jointly.

In particular, this method allows for working with the full stack of images, or with a subset of the most relevant images in the stack. This selection list is tailored to each artwork, since different artists worked with different materials and pigments. For example, there will be paintings where pigments with high content of zinc have a strong presence, and there will be other cases where calcium will be among the dominant elements of the pigments used in the composition. As seen in Section 2.2.1, the presence of such chemical elements can be identified by MA-XRF scanning of the artwork. As an example, MA-XRF scanning of *Portrait of Jane Boughton, née Coningsby* (AGOID.29371, The Thomson Collection at the Art Gallery of Ontario)², one of a group of English portrait miniatures scanned as part of the Unlocking The English Portrait Miniature project at the Fitzwilliam Museum [112], resulted in 15 element distribution maps of interest (Fig. 3.1). These element maps comprise the moving image to be registered to a visible image of the portrait miniature, and are as follows: silver (Ag - $L\alpha$ lines), arsenic (As - $K\beta$ lines), gold (Au - $L\alpha$ lines), barium (Ba - $L\alpha$ lines), calcium (Ca - $K\alpha$ lines), copper (Cu - $K\alpha$ lines), iron (Fe - $K\alpha$ lines), mercury (Hg - $L\alpha$ lines), potassium (K - $K\alpha$ lines), manganese (Mn - $K\alpha$ lines), lead (Pb - $L\alpha$ lines), lead (Pb - $M\alpha$ lines), phosphorus (P - $K\alpha$ lines), tin (Sn - $L\alpha$ lines), silicon (Si - $K\alpha$ lines).

It is important to note that, during the registration process, although each individual image (within the stack of images that comprise the moving image) is individually assessed to find a best transformation to register that image to the visible image of the artwork, the final transformation will be the same for all the images in the stack, since they are assumed to be aligned by construction.

3.2 Overview of the method

The method is initialised with the images to register: a target fixed image and a stack of images, that is treated as the moving image. By applying a transformation to the latter, it is possible to “distort” it to obtain the best alignment between the two images. The goal of this method is to identify the transformation that provides this best alignment.

The method is composed of 4 sequential *searches* (Fig. 3.2). At each search, all transformations within a prescribed range of parameter values are evaluated for registration of the images in the stack to the target fixed image. One transformation at a time is applied to each image in the stack, and the mutual information (MI) between each individual transformed image and the fixed image is obtained. An average of the metrics obtained is computed for each transformation that is evaluated. The transformation with the highest average metric, across all images in the stack, is kept and serves as the starting point for the following step. This process, illustrated in Fig. 3.3, is repeated at each search. Each search is more precise than the previous one, looking at parameters that are progressively more refined: the parameters are either larger in number or they are evaluated at smaller step sizes. This results in evaluating distortions applied to the moving image, which at first are coarse, and become more precise at each consecutive search.

² Images of this artwork are used in diagrams of Figs. 3.2, 3.3, and 3.4.

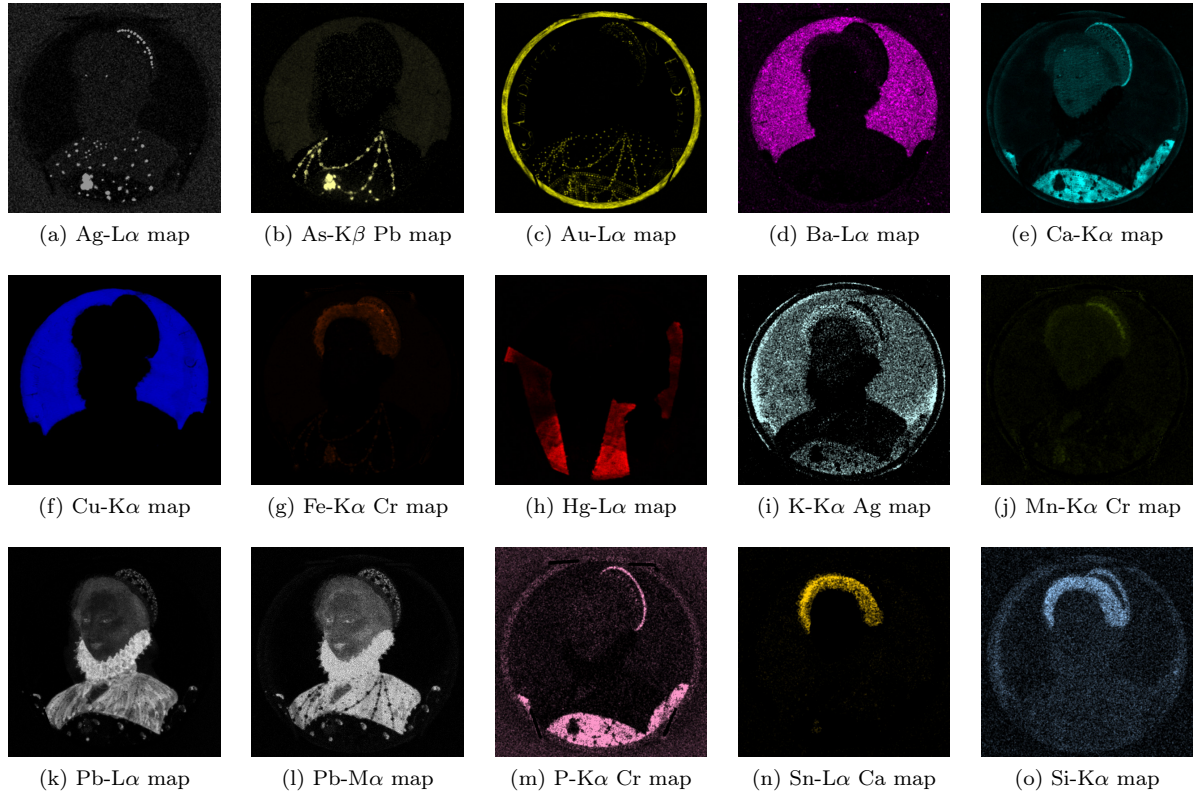


Figure 3.1: Element distribution maps: selection of 15 images derived from the MA-XRF scanning of the *Portrait of Jane Boughton, née Coningsby* (AGOID.29371, The Thomson Collection at the Art Gallery of Ontario). These compose the image stack which we treat as the moving image. Note that some of these images are subject to pixel averaging (a, b, d, i, j, m, n, o), or are processed by having another element subtracted (g, j, m).

Throughout the first three searches, progressively more detailed similarity and affine transformations are evaluated (see 3.4). Finally, non-rigid transformations for subtle distortions are evaluated at the last stage (see 3.5). As discussed in Section 2.4.1, the process of identifying the most suitable non-rigid transformation involves the optimisation of a large number of parameters at each iteration until converging to the right solution. This process is computationally expensive, whilst transformations with low number of parameters are much faster to compute. Thus, it is customary that registration with non-rigid transformations is conducted only after the fixed and moving images have undergone an initial affine alignment. This stepped approach allows the method to speed up the overall registration process whilst obtaining progressively better metrics at each search.

For visualisation of results, overlays of the target image and the transformed moving image are created. This is illustrated in Fig. 3.4 where for each stage of the method, the best transformation identified has been applied to one image of the stack (here the gold map (Au-L α element lines)) and the resulting image has been overlaid onto the fixed target image (visible image), so as to illustrate the progressive refinement of the alignment between the images. The accompanying

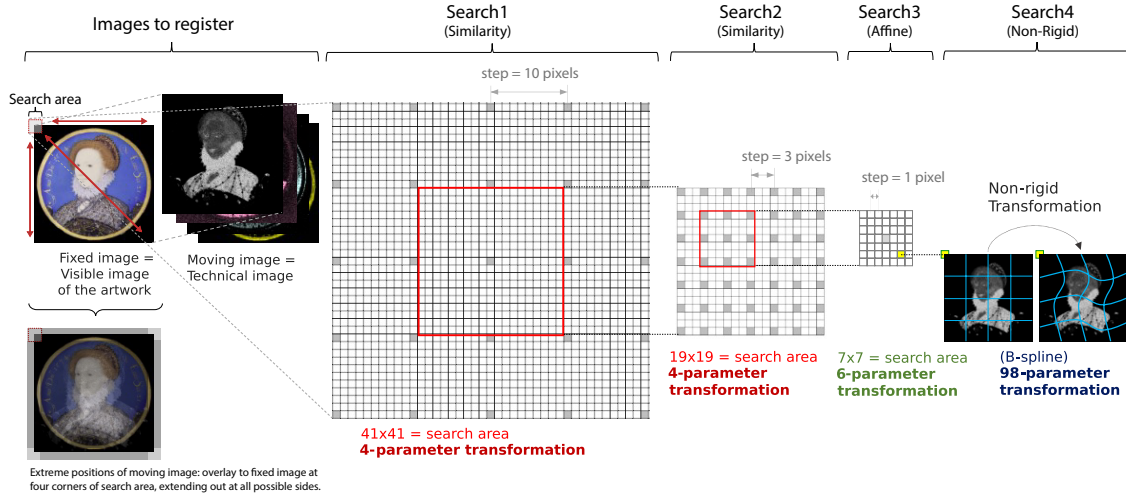


Figure 3.2: This sequence illustrates the initial images to register (fixed and moving image) and the 4 searches at progressively increased resolution - coarse at first, and progressively more refined. At each stage, a best transformation is identified and applied to the moving image, and this becomes the starting point for the next search. Through these searches, the best transformation for registration is identified with consensus.

metric is indicative of the average mutual information obtained by the transformation across all images in the stack, that is incrementally optimised at each stage of the method.

3.3 Implementation of the method

The method presented here aims to find one best transformation per moving image (that is here the stack of images derived from a single MA-XRF scanned area of the artwork at its original resolution) so as to register it to its corresponding region within the target image. This is usually the visible image of the artwork: a single-capture, high resolution photographic image of the full-visible artwork. Note that the resolution of the visible image of the artwork is usually higher than that of the MA-XRF scan. Thus, at the beginning of the workflow, the visible image is scaled down, manually, to match approximately the resolution of the moving image.

It is assumed that the user knows the approximate initial location of the moving image. Following this, an initial location may be assigned to the moving image, so that it is positioned as close as possible to what is believed to be the *correct* location of the moving image with respect to the target fixed image. Next, a distance of 20 pixels is set from this location (in each direction top, bottom, left, right) thus defining a *search* area of 41 x 41 pixels. Note that this distance of 20 pixels is consistent with the parameters of the first search - in particular, the translation step sizes along the x and y axis. These parameters, together with the amount of steps to evaluate in the first search, constitute the "search-space", that is the region that will be progressively analysed to identify the best transformation to apply to the moving image for its registration to the target

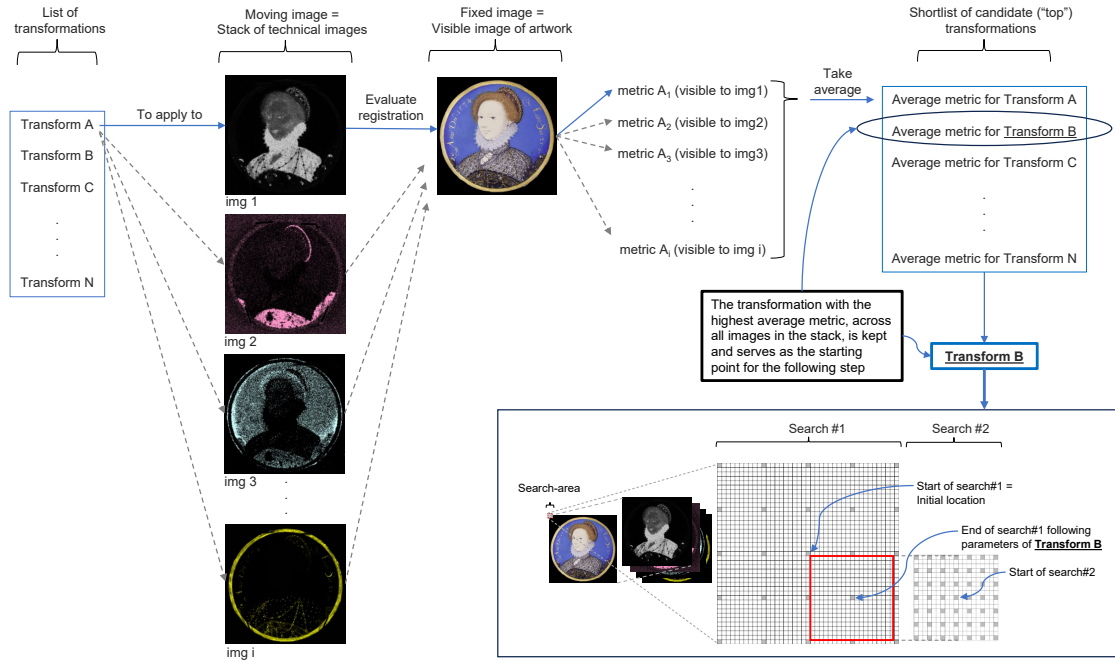


Figure 3.3: This sequence illustrates the process of identifying the best transformation at each search, with consensus among all images in the stack. From a list of transformations (here, Transforms A to N), one transformation at a time is applied to all images in the stack. The transformed images are then evaluated for registration to the target fixed image, and the corresponding metrics are obtained (here, metrics A_1 to A_i). The transformation that yields the highest average metric across all images in the stack is the one that will be selected as the starting point for the following search. Note that the number of transformations to evaluate ("N") depends on the number of parameters at each search, and on the range of parameter values to assess¹. Note that from the list of metrics that is obtained (in example, A_1 to A_i , B_1 to B_i , .. N_1 to N_i), only a few of them will bear a significant value, as some transformations will displace the images into further misalignment and return low metrics when evaluated at any image in the stack². Thus, the list of transformations for which to obtain an average metric of significance can be shortened to those transformations that return a high metric when evaluated at an individual image in the first place. In our method, this list is composed of the top 40 transformations³ that return the highest mutual information when evaluated at each individual image in the stack.

¹ in example, following the characteristics of the first search in Fig.3.2, 4 parameters are assessed (t_x , t_y , rotation and scaling), at a range of 5 values for t_x , 5 values for t_y , 3 values of rotation and 3 values of scaling, thus $N = 5 \times 5 \times 3 \times 3 = 225$.

² in example, any transformation that rotates an image of the stack in the opposite direction than that of the intended final alignment, may result in a very low metric. Thus there is no need of further computing the average metric that this transformation would yield across all images in the stack.

³ we take the 40 top transformations as this has proven to be a good compromise between accuracy and speed of computation.

fixed image (see Fig. 3.5 - note that a different painting, "*The Virgin of the Rocks*" (NG1093, The National Gallery, London), is used in the diagrams of this figure illustrating two possible "search-spaces").

Once the initial location is assigned to the moving image (a pixel-location within the target fixed image), the search-space is constructed around that location. Note that our convention refers

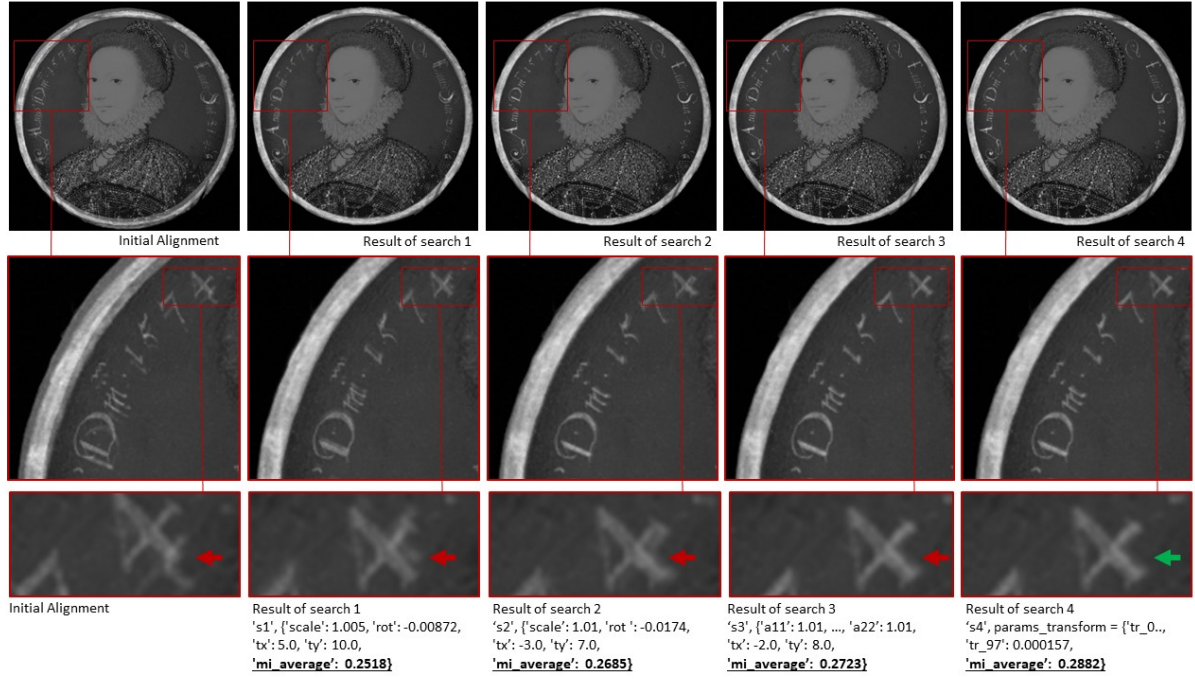


Figure 3.4: Overlays of target (visible) image and transformed moving image (gold (Au-L α element lines) maps with the corresponding applied transformations that are obtained at each stage of the method). This sequence illustrates the metrics obtained at each step, reflecting the incremental maximisation of mutual information. The top row shows the full extent of the images overlaid, and the middle and bottom row show progressively more highly magnified details from the regions indicated with red boxes on the row above.

to the position assigned to the top-left corner of the moving image³.

In the case that both the fixed image and the moving image capture the full extent of the artwork (e.g. as for *Portrait of Jane Boughton, née Coningsby*, shown in Figs. 3.2, 3.3, and 3.4), the top-left corner of the moving image may be positioned at the top-left corner of the fixed target image. Thus, the search-space is constructed around this location (0,0) and the method is ready to start. However, it might be the case that the moving image captures only a partial area of the entire artwork (e.g. as for the case study of “*The Virgin of the Rocks*”, shown in Fig. 3.5). In this case, once the initial location of the moving image is assigned, the full-visible image is cropped to the smaller corresponding region only (“Region of interest”, see Fig. 3.5). Namely, it is cropped to match the size of the moving image, plus 41 pixels along the x and the y axis respectively. For example, if the moving image is 100 pixels by 150 pixels in size, the visible image of the painting will be cropped to a region of 141 pixels by 191 pixels in size. Then, at initialisation, the moving image is placed at the centre of the region of interest. The search space is then set at the top-left corner of the moving image, for allowing the displacement of the moving image up to 20 pixels in all directions (top, bottom, right, left, corresponding to positive and negative directions along x and

³ in example, the moving image is translated by positioning its top left corner at the pixel-location within the target fixed image. The search-space is thus constructed around this pixel-location, with its centre at the top-left corner of the moving image.

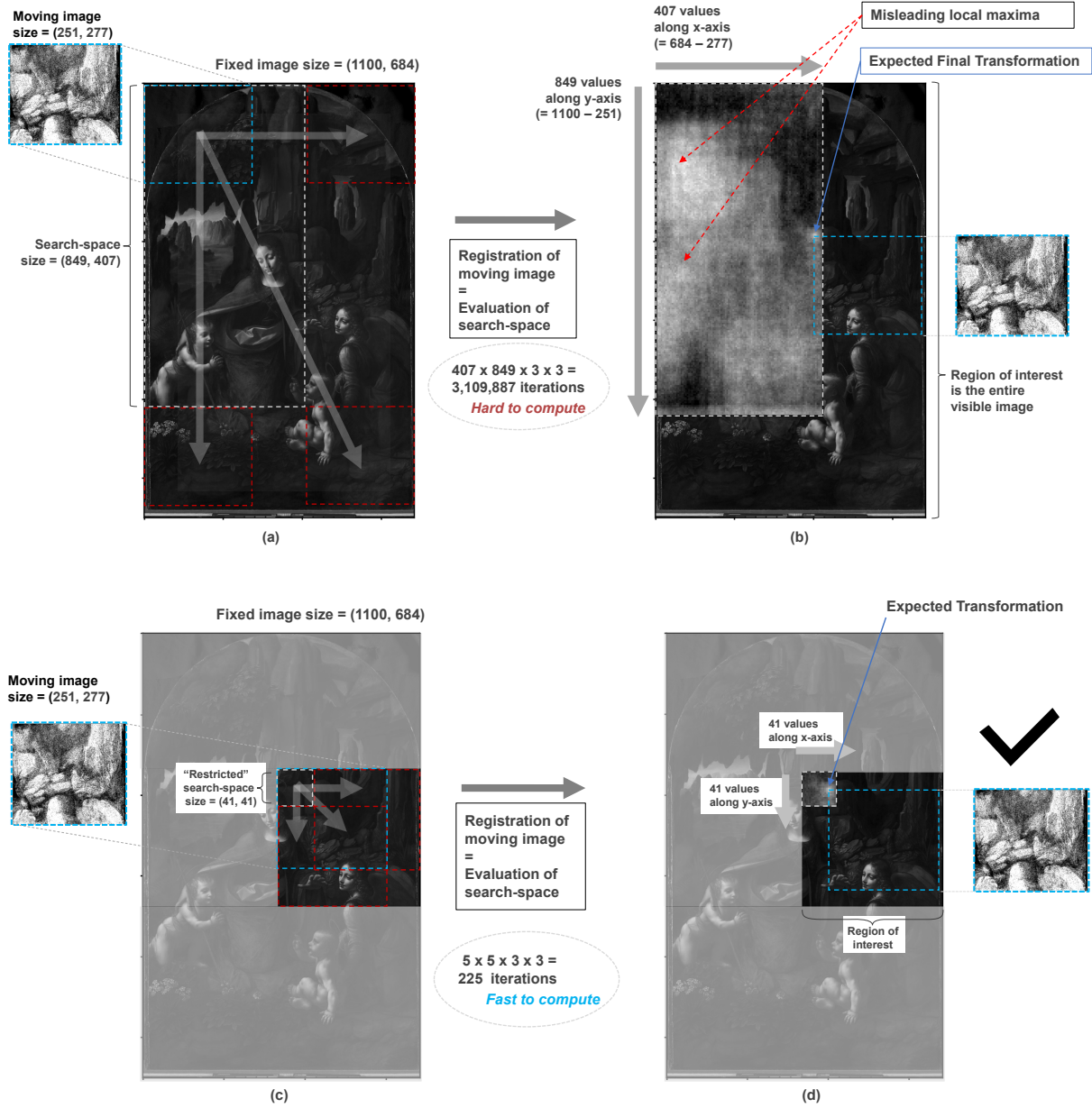


Figure 3.5: Registration of element maps resulting from MA-XRF scanning of area above the head of the angel of the painting "The Virgin of the Rocks" (NG1093, The National Gallery, London) [60]. In part(a) the moving image is evaluated across the entire visible image, resulting in a large search-space (the box in blue indicates the initial position of the moving image, and boxes in red indicate the extreme positions of the moving image, relative to the fixed image - same as in part (c)). This is computationally expensive and introduces the risk of asserting the registration at a misleading local maxima. In part(b) the search-space is depicted as a plot where regions of bright values are indicative of transformations with high metric values, and dark regions indicate where poor registration metric values are obtained (with blue box indicating the expected final position of the moving image - same as in part (d)). In part(c) the moving image is evaluated across a *region of interest* within the visible image, resulting in a restricted search-space and allowing for faster computations at each step of the method. Part(d) shows the restricted search space where 5 values of t_x , 5 values of t_y , 3 values of rotation and 3 values of scaling are parameters of the N transformations to evaluate, with $N = 225$. In all figures, the elemental map of lead (Pb-L lines) is representative of the moving image.

y axis) whilst containing the moving image within the region of interest (i.e. the corresponding, cropped area of the target fixed image). Once the moving image is assigned an initial location in relation to the fixed target image, the searches may begin.

3.4 Similarity and affine transformations

The first search consists in evaluating similarity transformations across 25 locations (i.e. grey locations at search 1 in Fig. 3.2), within a region of 41 x 41 pixels of the fixed image, to determine at which of these locations the best transformation for all images in the stack can be found. The transformations to evaluate in the first and second searches are defined [31] as:

$$\mathbf{y} = \mu_{s,\theta,t_x,t_y}(\mathbf{x}) = sR(\mathbf{x} - \mathbf{c}) + \mathbf{t} + \mathbf{c} \quad (3.1)$$

with s a scalar, R a rotation matrix of size 2×2 , $\mathbf{c}$ the centre of rotation, and $\mathbf{t}$ the vector for translation. Here, $\mathbf{x}$ is the initial position of a pixel in the input image, and $\mathbf{y}$ is the final position of the same pixel after the transformation is applied.

The parameters for the function μ consist of the angle of rotation θ , the isotropic scaling factor s , and the vector $\mathbf{t}$ for translation along x and y axis [31]. This means that the image is treated as an object, which can translate, rotate, and scale isotropically. The rotation matrix is parameterised by the angle θ in 2D. Please refer to Equations 2.12 to 2.18 for the relationship between these parameters and the matrix from of the similarity transformation function.

The list of the transformations to evaluate at each image in the stack results from the combination of these 4 parameters (scaling of the image, rotation, translation along x axis, and translation along y axis), each of them restricted to a discrete range of values (namely: 3 values for rotation, 3 values for scaling, 5 values for translation along the x-axis, and 5 values for translation along the y-axis). Note that the ranges of parameter values were chosen to balance speed of computation and precision of the results obtained. The first search allows for relatively large displacements (step size for translations in both the x-axis and y-axis is 10 pixels); whereas the step size for rotation and scaling is restricted to 0.5 degrees and 0.05 respectively. These step size values are added and subtracted from the “default” set-up values for the moving image (initial location along x and y axis, and initial values for rotation angle and scaling factor - typically 0.0 and 1.0), resulting in the discrete range of parameter values of the transformations to evaluate. Thus, the search is “coarse” in the sense that it evaluates transformations at positions that are 10 pixels apart each time. Nonetheless, all parameter values within the above-mentioned discrete range are evaluated, making this search “exhaustive” in that aspect. Subsequent searches have smaller step sizes for translations, reducing the sparsity of the search at each turn - which can be also considered as “increasing the resolution” of the search for the best transformation at each turn.

Each transformation in the list is applied to each image in the stack and evaluated for regis-

tration to the fixed image. The metric used for registration is the *mutual information* between the fixed image and the transformed moving image when overlapped. A higher value of the metric indicates a better registration of the images (see 2.4.2). This “coarse-exhaustive” search is done for each image in the stack. The list of all the transformations evaluated on all images is generated, and then sorted by descending value of metric obtained for each transformation. A shortlist of the top 40 transformations is kept as candidate for further evaluation - namely, the 40 transformations that yield the highest values of mutual information at the registration of any individual image in the stack to the fixed image. For each candidate transformation in this shortlist, the average mutual information is calculated. This is obtained by applying the candidate transformation to each image in the stack for its registration to the fixed image. The transformation with the highest average mutual information is kept as the best transformation resulting from this search, and will serve as the starting point for the following search (Fig. 3.3).

The second search investigates the neighbouring area that follows from the best transformation obtained at search 1. Starting from the parameters of this transformation, the above-mentioned process is repeated for each image of the stack, evaluating similarity transformations at a higher number of locations within a smaller region of the fixed image (namely, 49 locations within a region of 19 x 19 pixels, i.e. grey locations at search 2 in Fig. 3.2 and Fig. 3.3). Thus, as step sizes decrease in relation to the previous search (3 pixels for translations in x-axis and y-axis), the transformation of the moving image is now evaluated more precisely, allowing for the second search to be of a higher resolution than the previous one. As in search 1, the transformation that returns the highest average mutual information is the one selected as the best transformation, and thus informs the starting point for the following (third) search. Note that this second search will either yield a better transformation than the one obtained so far, or will keep the existing one as the best transformation resulting from these first two searches. Consequently, after a new search is concluded, the value of the average mutual information will increase or remain the same.

The third search investigates the neighbouring area that follows from the best transformation obtained at search 2. Starting from the parameters of this result, affine transformations are evaluated across 49 locations within a region of 7 x 7 pixels of the fixed image (i.e. all locations at search 3 in Fig. 3.2). These transformations are defined [31] as follows:

$$\mathbf{y} = \mu_{s_x, s_y, \theta, m_1, m_2, t_x, t_y}(\mathbf{x}) = A(\mathbf{x} - \mathbf{c}) + \mathbf{t} + \mathbf{c} \quad (3.2)$$

where the matrix A allows for translation, rotation, scaling along x and y axis, and shear along x and y axis. A is a 2x3 matrix composed of the elements $(a_{11}, a_{12}, t_x, a_{21}, a_{22}, t_y)$. The parameters of the transformation μ consist of the scaling factor s_x along x-axis and s_y along y-axis; the angle of rotation θ , the shear factor m_1 along x-axis and m_2 along y-axis, and the translation t_x along x-axis and t_y along y-axis [31]. Please refer to Equations 2.21 and 2.22 for the relationship between these parameters and the elements of matrix A . This type of transformation allows for the image

to be treated as an object, which can translate, rotate, and be subject to anisotropic scaling and to anisotropic shear.

For each image of the stack, the number of transformations to evaluate results from the combination of these 6-parameters, each of them restricted to a discrete range of values. This search not only takes in smaller step sizes than search 2, but also explores more versatile transformations with 6 degrees of freedom, allowing for more subtle distortions than those evaluated in the first two searches. Step size for translations along x and y axis is 1 pixel whilst the step values for its remaining parameters allow for a rotation step size of 0.05 degrees, and for a scaling factor step size of 0.005. These step size values are added and subtracted from the transformation parameters obtained at search 2, resulting in the new discrete range of parameter values of the transformations to evaluate. As in search 1 and search 2, a list of all the transformations evaluated on each image of the stack is generated, and then sorted by obtained metric value. A shortlist of the 40 transformations with highest mutual information at any given image in the stack is kept as candidates for further evaluation. Each candidate transformation is then evaluated on all images, and the average mutual information is computed for each transformation. The transformation that obtains the highest average mutual information is kept as the best transformation from this search, and will serve as a starting point for the following (fourth) search.

3.5 Non-rigid transformations

This fourth search consists in evaluating non-rigid (B-spline) transformations (see Section 2.4.1). The objective of this search is to find a best transformation to be added as a final “adjustment” to the moving image, on top of the best affine transformation obtained at search 3. Note that this follows the assumption that the artworks we work with are nearly flat objects, only needing a small non-rigid transformation to help adjust the final alignment. This type of transformation, unlike the ones implemented in the previous searches, entails a parameterised control-point grid that allows for smooth, continuous, piece-wise distortions to the moving image. These non-rigid adjustments may be necessary due to non-flat support for paintings (e.g. warped panel, twisted canvas strainer, buckled paper) or effects linked to the capture devices such as lens distortions, and it would not be possible to achieve them with the similarity or affine transformations applied so far.

The implementation of a parameterised control-point grid is built from the notion that the moving image could be subdivided and treated as a series of regular tiles. Each tile could then be individually registered to its corresponding location within the target fixed image, where each tile would benefit from a distinctive affine transformation. Composing each of these affine transformations, at the location of each tile, would correspond to the effect of a non-rigid transformation applied to the entire moving image (Fig 3.6). This type of transformation is implemented as part of the elastix library [25], where B-splines [90] are used as parameterisation as described in

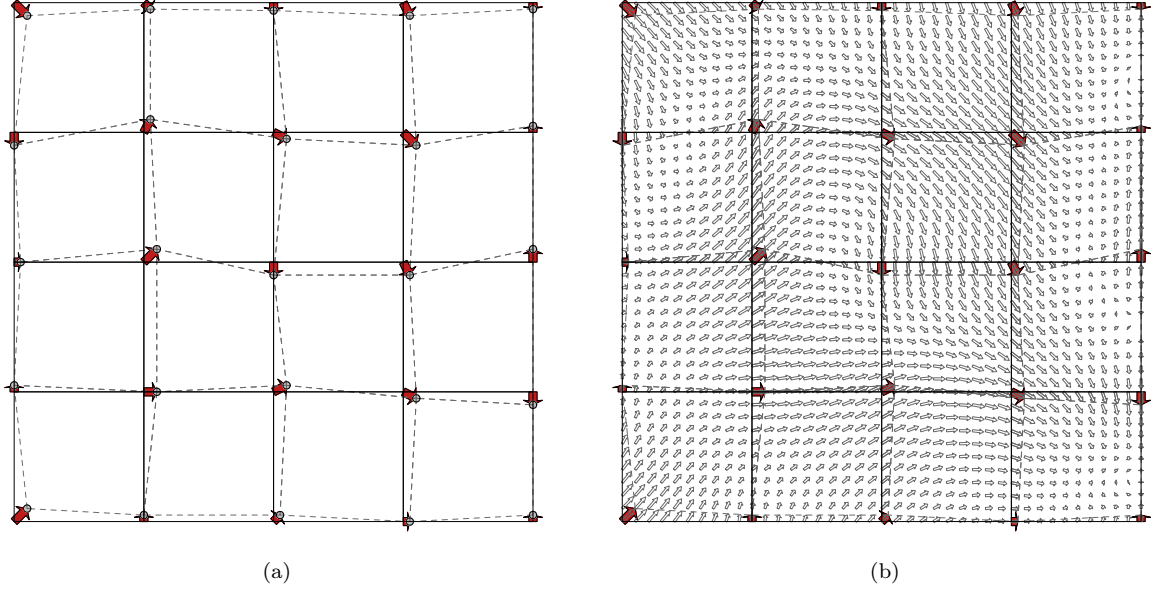


Figure 3.6: This sequence shows the moving image subdivided into regular tiles, where each tile benefits from a distinctive affine transformation. Part(a) shows the vectors (at each corner of the subdivisions) representative of the displacement that would result from applying affine transformations to each tile. Part(b) shows the vector field resulting from the displacements intended at each node of the image, corresponding to the effect of a non-rigid transformation applied to the entire moving image.

3.5.1 B-splines transformations: control-grid point

The control-point grid starts as a regular grid. The number of subdivisions is set in the algorithm, and is consistent with the spacing of the control-point grid. During the optimisation process, the positions of these control points can be changed. Each control point has a region of influence within the image, and can be displaced individually, affecting their neighbouring pixels only.

In the experiments shown here, the control-point grid is set to a resolution of 4 rows x 4 columns. We have found this grid resolution to be a good compromise between performance (i.e. the degrees of freedom explored with this grid resolution) and speed of computation. Note that the number of parameters M increases quadratically with the resolution of the control point grid. Therefore, increasing the resolution may quickly result in a very high number of parameters that can be computationally expensive to optimise. In our case, since the artworks we work with are considered to be almost planar and with very small non-rigid distortions, working with a low-resolution control-point grid has allowed us to obtain good registration results.

3.5.2 B-splines transformations: number of parameters

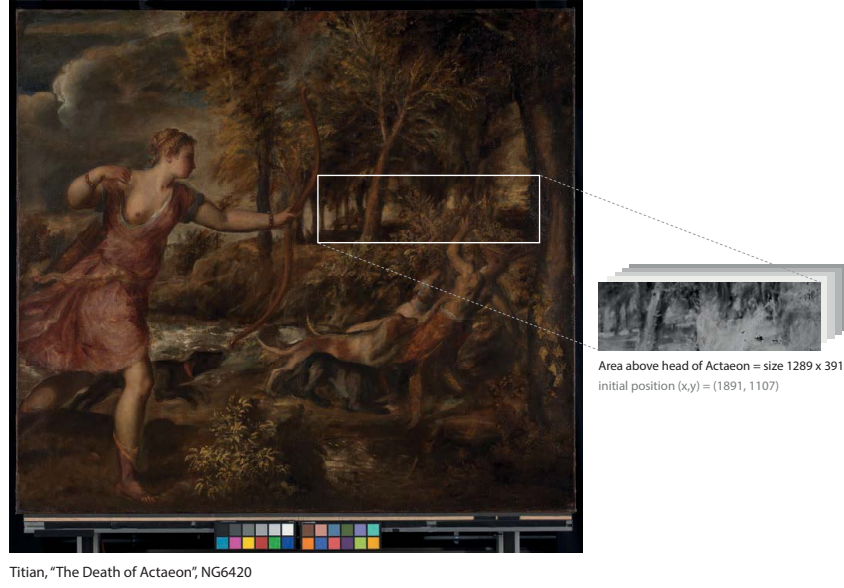
In this type of transformation, the number of parameters to optimise at each iteration is calculated as $M = (P_1 \times \dots \times P_d) \times d$, with d the image dimension (in this case, $d = 2$) and with P the number of control points at each dimension, $P = (P_1, \dots, P_d)$. In the experiments presented here, we assign a grid of 4 rows x 4 columns to a 2D image. For each dimension we will then have 5 control points to denote the grid, plus one control point just outside the image at either end of the grid. Thus, we will have 7 control points for each dimension, and the number of parameters $M = (7 \times 7) \times 2 = 98$ parameters, which is the amount of B-spline coefficient vectors $\mathbf{p}_k$ (for its components in x and y axis, see Fig. 2.20). Note that these are the parameters that are optimised in each iteration of this fourth search.

As mentioned in Section 2.4.1, the number of parameters (independent variables) of the transformation corresponds to the amount of degrees of freedom that can be modified in the moving image. The higher the number of parameters, the more computationally expensive it is to optimise the transformation to apply.

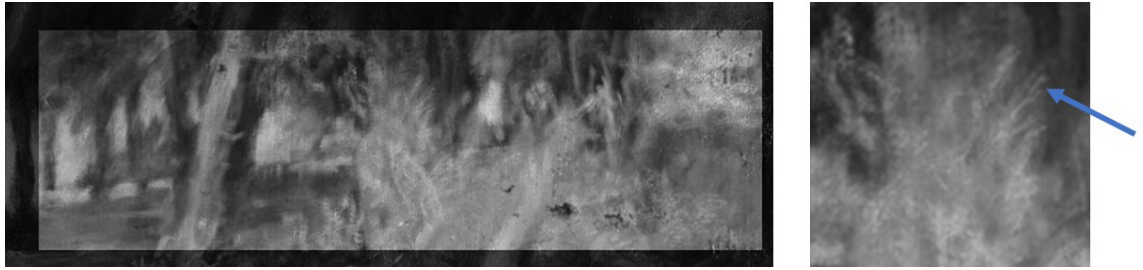
3.5.3 B-splines transformations: L-BFGS optimiser

It is important to note that this type of transformations involves the optimisation of a much higher number of parameters than in the previous searches. Whilst similarity and affine transformations look at optimising 4 and 6 parameters respectively, using the above mentioned settings we now look at optimising 98 parameters at each turn. In this scenario, working with an exhaustive search is no longer feasible, as it would take too much computing memory. Instead, for this search, the optimiser Limited-memory BFGS (L-BFGS) is used [94, 88], see Section 2.4.3. This optimisation algorithm approximates the Broyden–Fletcher–Goldfarb–Shanno algorithm (BFGS) using a limited amount of computer memory, and is a popular algorithm for parameter estimation. Storing only a few vectors that represent an approximation of the inverse Hessian matrix implicitly, the L-BFGS method is particularly well suited for optimization problems with many variables. In this way, this fourth search entails over 50% of the overall computation time. The improvements to the registration due to the implementation of this type of transformation is shown in Fig 3.7. Note that a different painting, “*The Death of Actaeon*” (NG6420, The National Gallery, London), is used in the diagrams of this figure. This artwork was chosen to validate the implementation of this type of transformations, as it is considered particularly problematic in terms of registration. This is due to the fact that the technical images were acquired non-consecutively, over an extended length of time, allowing for changes in position of the optical device, and leading to different distortions even in adjacent images.

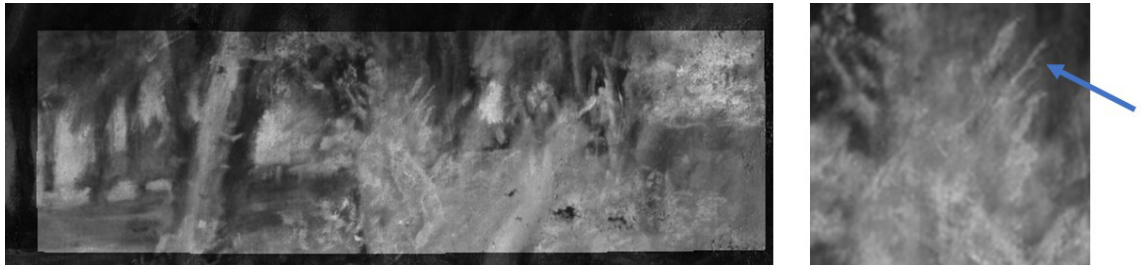
Note that non-rigid registration is known to be the most computationally-expensive step in any registration method [2, 36, 31], and efforts in reducing the computation time of this type of registration have resulted in exploring not only the implementation of faster optimisers to handle



(a) "*The Death of Actaeon*" (NG6420, The National Gallery, London), and lead map (Pb-L lines) representative of scanned area of wooden region above Actaeon's head of the painting.



(b) NG6420, Scanned Area - registration obtained after search 3 (affine transform applied).



(c) NG6420, Scanned Area - registration obtained after search 4 (affine plus non-rigid transform applied).

Figure 3.7: Registration of element maps resulting from MA-XRF scanning of area of wooden region above Actaeon's head of the painting "*The Death of Actaeon*" (NG6420, The National Gallery, London): overlays of target visible image with lead map (Pb-L lines). Part(a) shows the registration to visible image only. Part(b) shows the registration obtained after 3 searches, with final affine transformation only, and part(c) shows the registration obtained after 4 searches, with final composite transformation of affine plus non-rigid transformation. Blue arrows indicate details where improvement of the registration is most noticeable.

large numbers of parameters [113, 114, 94], but they have also led to the development of learning-based methods [36, 37, 35, 38] (see Section 2.5). This step, intended as a final refinement to the

registration of the moving image, is not always required, as the method can achieve satisfactory results at the end of search 3. As mentioned at the beginning of this section, this area-based method follows the assumption that the artworks we work with are nearly flat objects, only needing a small non-rigid transformation to help adjust the final alignment. However, there have been cases where the registration provided by this last search did not help enhancing the alignment obtained in search 3, making inefficient use of computation time or even degrading the alignment obtained until that point.

Thus, this last search allows for investigation of a final non-rigid best transformation, starting with an initial transformation as per the one obtained from search 3. Each non-rigid transformation to evaluate is applied as a composite transformation, together with the best transformation resulting from search 3. For each composite transformation that is evaluated, the mutual information between the fixed image and the transformed image is obtained. This is done for each image of the stack. Following the procedure for searches 1-2-3, a list of all the composite transformations evaluated on all images is generated, and then sorted by descending value of metric obtained. A shortlist of the first 40 composite transformations (those with highest metric at any given image) is kept as candidates. Each of these composite transformations is then evaluated on all images, and the average mutual information is computed for each composite transformation. The composite transformation that provides the highest average mutual information across all images is kept as the best resulting one from this last search, and will serve as the final transformation to apply to the full image stack for registration to the target image.

3.6 Discussion

Throughout the implementation of this automated area-based method, the individual images that constitute the image stack are crucial to identify good candidate transformations. The more pictorial, relatable information to the target image they may contain, the more suitable candidate transformations may be identified, since they will obtain a higher value of mutual information when applied to the individual image. In the case of MA-XRF scans (and other technical images/image modalities that reveal non-surface features), the images derived may show different information from one another as well as from that of the visible image, since each of them represents the quantity map of each chemical element detected by the MA-XRF scan.

If the artwork was created in a way such that the materials and pigments being used were relatively homogeneously distributed across the painting, it would be likely that a number of images within the stack contained pictorial information that is well relatable to the visible image of the painting (Fig. 4.9). On the other hand, if the artwork was created with different materials for distinctive features or regions of the artwork (such is the case with manuscripts and portrait miniatures), different materials may be clustered only locally throughout the artwork (Fig. 3.1). Thus, the information in each image may only be relatable to localised, clustered areas of the

artwork. In Section 4.1.2 we show results of our method when applied to this type of artworks. Additionally, in Section 4.1.3 we show how, in these cases, it is possible to enhance the image stack by adding synthetic images resulting from merging two or more individual, distinctive images from the same stack. By combining the individual images, the information presented by the newly merged image can facilitate the identification of transformations to register various distinctive features at once. Another strategy to identify suitable transformations is addressed in Section 4.2.2. Here, instead of enhancing the information provided by the moving image, an enhanced version of the target fixed image is created by taking the full-visible image of the painting, and adding the pixel values of an initially registered technical image with well-defined features. Thus, this new composed image may now serve as the fixed target image for the registration of the image stacks corresponding to the scanned areas surrounding the initially registered technical image (Fig. 4.12).

It is important to note that in a painting there are more paint layers and pigments than what is typical in manuscripts or portrait miniatures. For this reason, technical images of paintings may have features that cannot be seen in the visible image as they may be covered by other layers. Moreover, paintings may have features that, although expected to be seen in the corresponding MA-XRF elemental maps, do not appear because of *shielding* effects - for example, a heavy atomic element such as lead may shield the presence of other lighter elements that may be present, simultaneously, in the same pixel location⁴. Another reason why features may not be shown in technical images can be that the pigment used for these features may be based on materials that contain elements that cannot be detected by MA-XRF (this may be the case for some pigments based on organic colorants, for example, red or yellow lake pigments, and most black pigments). As a consequence, there may be passages that are a clear colour in the visible image, but that colour is from elements that the MA-XRF does not detect for what they will not appear in the MA-XRF element maps.

In such cases, the image stacks may contain images with potentially misleading information for either feature-based or standard area-based methods. The strength of the method presented in this section relies on the consensus that is achieved among all the images in the stack, allowing to overcome the presence of images of this kind yet ensuring that these images are correctly registered to their corresponding area within the visible image of the artwork.

3.7 Summary of this chapter

In this chapter we reviewed the workflow behind the automated area-based method for image registration of multimodal technical images of artworks, which is the main subject of this thesis. Building from Chapter 2, this chapter addresses the details involved in the automated registration of images obtained from MA-XRF scanning of artworks - from setting up the moving image (in

⁴ The occurrence of shielding effect may result in the signal recovered at that location by the MA-XRF scanning to identify the presence of lead and miss the signal of the elements that are shielded by the heavier element

this case, the obtained technical images arranged into an aligned image stack), to the steps used along the process to identify a unique best transformation to apply to all the images comprising the stack. This was followed by a detailed explanation of the four *searches* comprised along the method, together with the transformations that are consecutively applied at each search. This section concluded with a discussion of the insights that were gained during the development of this method. Examples of its application are presented in the next chapter.

Chapter 4

Experiments

In this chapter we present the experimental setup and results of our automated area-based method for the registration of element maps resulting from MA-XRF scanning of 2D artworks. We divide these experiments into two groups: registration of images that capture the entire 2D artwork (this may be the case for small-sized artworks that can be captured at a reasonable resolution in one scanning), versus registration of images obtained from multiple scanned areas (this may be the case for medium-sized and large artworks ¹). In the second case, it may be necessary to combine the technical images resulting from the multiple scanned areas to create a single composite image of the entire artwork. The combination of these images can be guided by the visible image and it is referred to as *mosaicking*. Note that it can happen that only the scanning of a detailed small area may be needed from a large artwork. In this case, it might still be necessary to register that area in relation to the visible image to show it in its correct position in space.

4.1 Registration of images that capture an entire 2D artwork

In this first set of experiments, we show results for *Portrait of Jane Boughton, née Coningsby* (AGOID.29371, The Thomson Collection at the Art Gallery of Ontario) [112]. We apply our method for the registration of the element maps obtained from a single MA-XRF scanning to the visible image of the same artwork.

Firstly in this section, we show registration results of a moving image that is composed of the original images obtained from MA-XRF scanning. Secondly, we show registration results using a moving image that is composed of the original images obtained from MA-XRF scanning of the entire artwork, plus an additional *synthetic* image. This additional image is obtained as a combination of

¹ In example, the painting of the case study presented in Chapter 3, “*The Virgin of the Rocks*” (NG1093, The National Gallery, London) is 120 cm x 189.5 cm in size, and it was scanned in areas of about 48.7 cm x 44.3 cm (such as the scanning area above the head of the angel, shown in Figs. 3.5, 2.6). Note that, although the maximum scanning area of this instrument (MA-XRF scanner M6 JETSTREAM, Bruker Corp.) is 60 cm x 80 cm, the scanning of the artwork may be conducted in areas of smaller dimensions as these may depend on the resolution required, the resulting duration of the scan and the overall dimensions of the painting.

two or more of the original images of the stack, enhancing the quality of the registration obtained. Finally, for the purpose of comparing results, we conduct the registration of these images with the automated feature-based method developed by Conover et al. [11], presented in Section 2.3.4.

4.1.1 Experimental details

We work with a selection of 15 of these element maps (see Fig. 3.1), which were obtained by the teams at The Hamilton Kerr Institute and The Fitzwilliam Museum [112], as follows: a MA-XRF scanner (M6 JETSTREAM, Bruker Corp.) equipped with a 30 W Rh tube with polycapillary optics operated at 50 kV and 600 μ A and dual 60 mm² SDD detectors set to a 275 kcps threshold was used to scan the object. The scan was completed with the instrument head at a distance of approximately 12 mm from the surface of the painting, resulting in an X-ray spot size of 100 μ m. The pixel size was also set to 100 μ m (setting the spatial resolution of the element maps obtained) and a pixel dwell time of 10 ms was used. Note that, using these settings, it took about 60-80 minutes to scan each of these artworks of size circa 5 cm x 5 cm. The data was processed using the Bruker ESPRIT software, using the in-built deconvolution function. The resulting images are 8-bit colourised tiffs that have in some cases undergone further manipulations involving subtractions², intensity adjustments and/or pixel averaging.

4.1.2 Image stack composed of element maps obtained from MA-XRF scanning only

For the registration of *Portrait of Jane Boughton, née Coningsby* with our area-based method, the image stack is composed of the above-mentioned selection of 15 element maps (Fig. 3.1). These images are 470 x 470 pixels in size. The output of this method is an optimised unique transformation, obtained with consensus among all images in the stack, that can be applied to the entire image stack. See Fig. 4.1 for the image setup that applies to this first experiment.

The method is started by assigning an initial location to the moving image, so that it is positioned as close as possible to the target fixed image. Note that the initial location of the moving image is provided as an input by the user. At this stage, the misalignment of these images may be highly noticeable when visualising their overlay (Fig. 4.2(b)).

As described in Section 3.2, the method is composed of four stages called *searches* (Fig. 4.3). At each of these stages, a series of transformations are evaluated across all the images in the stack, and a unique best transformation is selected with consensus among these images. Thus at each stage, the moving image is progressively better aligned to the target fixed image. Initially, the moving image is subject to transformations that provide coarse approximations. These are followed by more detailed alignments, as more suitable transformations are identified after each search. The final non-rigid transformation allows for the more subtle, overall alignment of the

² Please refer to details in Fig. 3.1, in particular maps corresponding to parts(g, j, m).

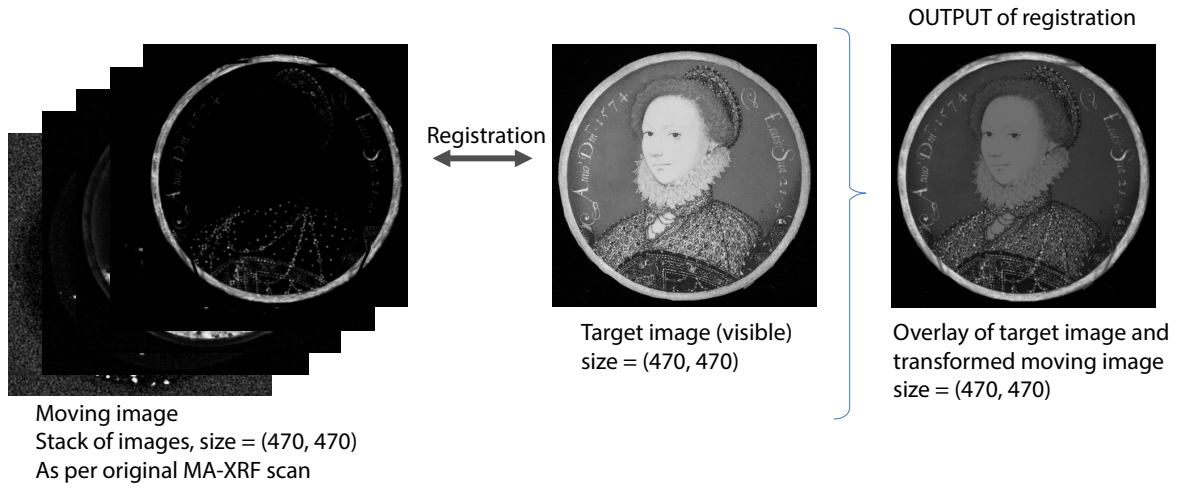


Figure 4.1: Initial setup of working images. The moving image consists of a stack of images that are selected from the MA-XRF scanning. These are registered, one at a time, to the target (visible) image of the artwork. The output is a unique best transformation, that is identified with consensus and can be applied to the full image stack. Overlays are created for inspection of results - here, the element map of gold is shown for reference.

images of the artwork. Please refer to Sections 3.4 and 3.5 for further information of these stages of the method. Overlays showing the initial alignment and final alignment obtained in this case are shown in Fig. 4.2(b,c).



Figure 4.2: Registration of MA-XRF scanning of *Portrait of Jane Boughton, née Coningsby*: (a) Au-L α map and (b) Overlay of target visible image with Au-L α map, shown at the commencement of the registration method as the *initial alignment*; and (c) Overlay of target visible image with transformed image of Au-L α map, shown as reference of the *final alignment*.

In the case of these Portrait miniatures, where the artwork and its technical images are relatively small in size, the improvement of the registration offered by the last step is only barely noticeable, and thus the first 3 searches are normally sufficient. However, the final non-rigid transformation provides noticeable enhancement to the registration of the images of larger artworks (i.e. Fig. 4.7). This type of transformation is especially useful when the artwork is set on a non-flat or undulating

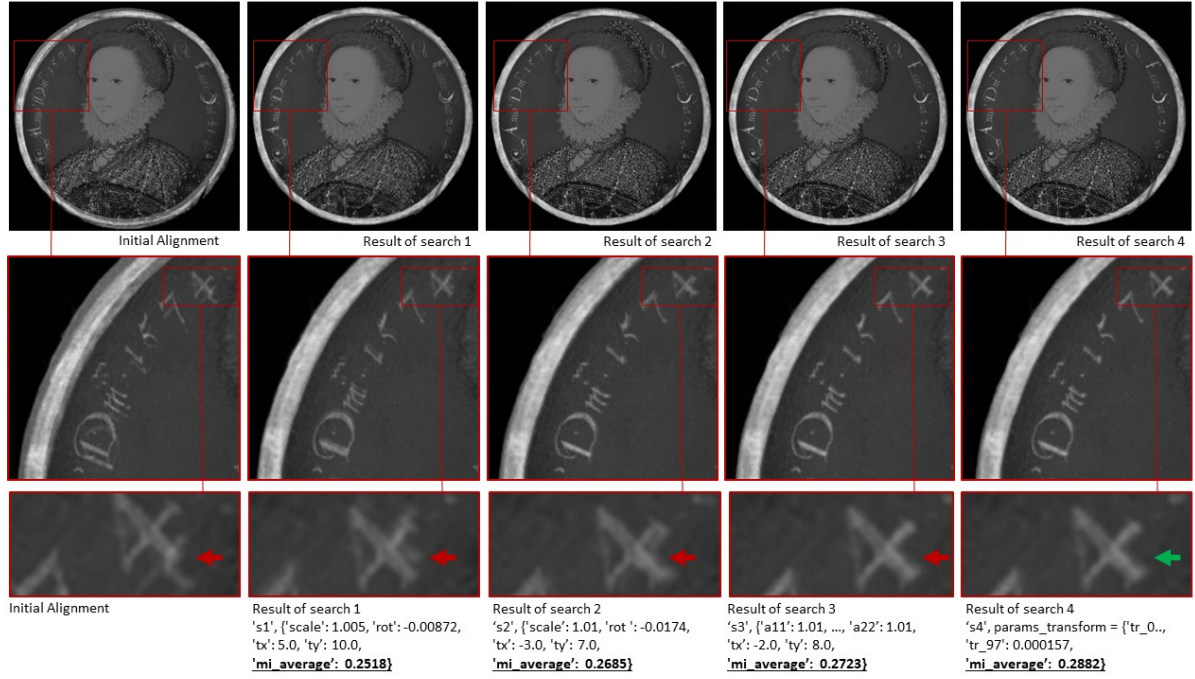


Figure 4.3: Overlays of target (visible) image and transformed moving image (gold (Au- $L\alpha$ element lines) maps with the corresponding applied transformations that are obtained at each stage of the method). This sequence illustrates the metrics obtained at each step, reflecting the incremental maximisation of mutual information. The top row shows the full images overlaid, and the middle and bottom row show progressively more highly magnified details from the regions indicated with red boxes on the row above.

support (see Section 3.5). Thus, it is a choice of the user to implement this fourth search, as it is also the most time consuming stage (involving about 50% of the total computation time).

As discussed in Section 2.2.1, the images obtained from MA-XRF scanning show the different chemical elements present in the artwork, and thus these images may show different features of the pictorial composition. In the case of Portrait miniatures, the pigments and other materials (such as gold leaf) utilised for the different regions of the artwork are quite different from one another, resulting in limited overlap between different element maps. As a consequence, each element map may highlight areas of the artwork that may be *exclusive* to that map (for example, notice the difference between the elemental maps of gold and lead in Fig. 4.4). Thus, some candidate transformations identified by the method may be efficient for the registration of the element maps showing, e.g., features of the clothing shown in the artwork only, whereas other candidates may be effective at the registration of the inscriptions - while not accounting for the features in the clothing at all. In the following experiment, we show that including an additional image resulting from combining individual images containing distinctive information of the artwork can help enhance the final registration.

4.1.3 Image stack composed of images obtained from MA-XRF scanning plus additional synthetic images

In the case of the second experiment, the image stack is composed of the 15 images used in the previous experiment plus an additional *synthetic* image. This synthetic image is created by combining two or more images from the same stack that have distinctive information from different elements of the pictorial composition. Here we combine the element maps of lead (Pb-M α lines) and gold (Au-L α lines). See Fig. 4.4 for the image setup that applies to this approach, and Fig. 4.5 for registration results.

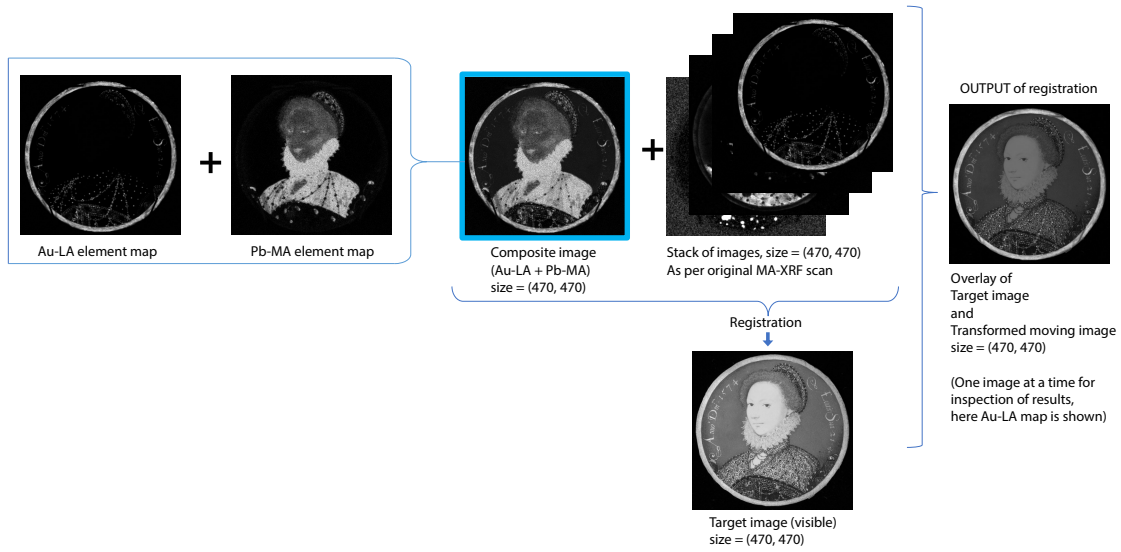


Figure 4.4: This sequence shows the creation of the *synthetic* image, created by combining the lead and gold element maps. This composite image is added to the stack of images obtained from MA-XRF scanning. This augmented image stack constitutes the moving image, that is registered to the visible fixed image. The output is a unique best transformation, that is identified with consensus and can be applied to the full image stack.

Belonging to the same image stack, the selection of which images to combine is based on the features they show. The more distinctive they are, the better candidate transformations will be identified by the method. In this case, the image of the element map of lead shows features of the clothing and facial features of the subject shown in the artwork, whereas the element map of gold shows details of the inscriptions in the artwork. The selection and number of images to combine is a choice of the user, as multiple combinations are possible.

Again in the case of the Portrait miniatures, where the artwork and its technical images (470 x 470 pixels at original resolution) are relatively small in size, the improvement of the registration offered by adding a synthetic image is only barely noticeable, and thus the results obtained for the registration of the *standard* moving image composed of only the original images obtained from MA-XRF scanning might suffice. However, this technique has shown noticeable improvements

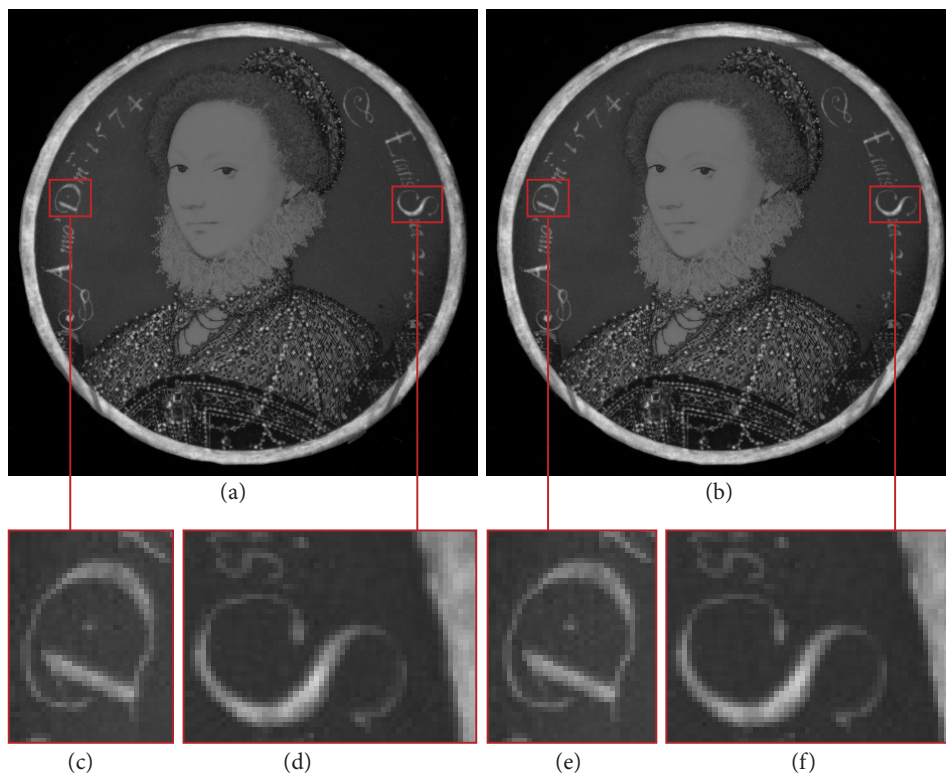


Figure 4.5: Registration of MA-XRF scanning of the *Portrait of Jane Boughton, née Coningsby*: overlays of target visible image with Au-L α map. Final alignment obtained with the original stack of 15 images only (a), and final alignment obtained with the enhanced stack of 15 images plus additional synthetic image (b). In red is shown a close-up detail of the *S* inscription, showing slight improvement of the registration by the addition of the synthetic image to the image stack.

to the registration of images of higher resolution. Thus, we proceed to show results for a case where, instead of reducing the resolution of the visible image of the artwork to match the original resolution of the MA-XRF scanning (see Section 3.3), the resolution of the technical images is increased to match the resolution of the visible image of the artwork (see Fig 4.6).

In the following experiment, the images for registration are the upscaled (i.e. enhanced resolution) version ³ of the element maps obtained from MA-XRF scanning of the *Unknown Man, formerly called Edward, Baron Herbert of Cherbury* (AGOID.29377, The Thomson Collection at the Art Gallery of Ontario) [112]. Note that the resolution of these element maps is increased so as to match the resolution of the visible image of the artwork, from 470 x 470 pixels to 2991 x 3660 pixels. Fig. 4.6 shows the setup of the resulting images for registration. Part(a,b,c) show the enhanced resolution version of the element maps of lead, copper and gold respectively. These images are combined to produce the *synthetic* image ⁴ (part(d)) that is added to the image stack

³ technical images were enlarged using the "Preserve Details" method in Photoshop 2021 to match dimensions of the visible image of the artwork - from their original size of 470 x 470 pixels to 2991 x 3660 pixels.

⁴ In this case, technical images are combined after being set to 1/3 of their intensity value, so that the image resulting from the combination of the three technical images does not surpass the limit pixel value of 255 for 8-bit images,

(part(e)) for registration to the target fixed image, which is the greyscale version of the visible image, at its original resolution of 2991 x 3679 pixels (part(f)). Thus, the moving image consists of the selected images obtained from the MA-XRF scanning, plus the additional *synthetic* image. The output is a unique best transformation, that is identified with consensus and can be applied to the full image stack. Part (g) shows the overlay of the visible image and the registered elemental map of gold.

A comparison of the results of the registration conducted with the *standard* moving image (i.e. composed of the image stack of 15 images obtained from MA-XRF scanning), versus registration conducted with the augmented moving image (i.e. composed of the image stack of 15 images plus the synthetic combined image) is shown in Fig. 4.7.

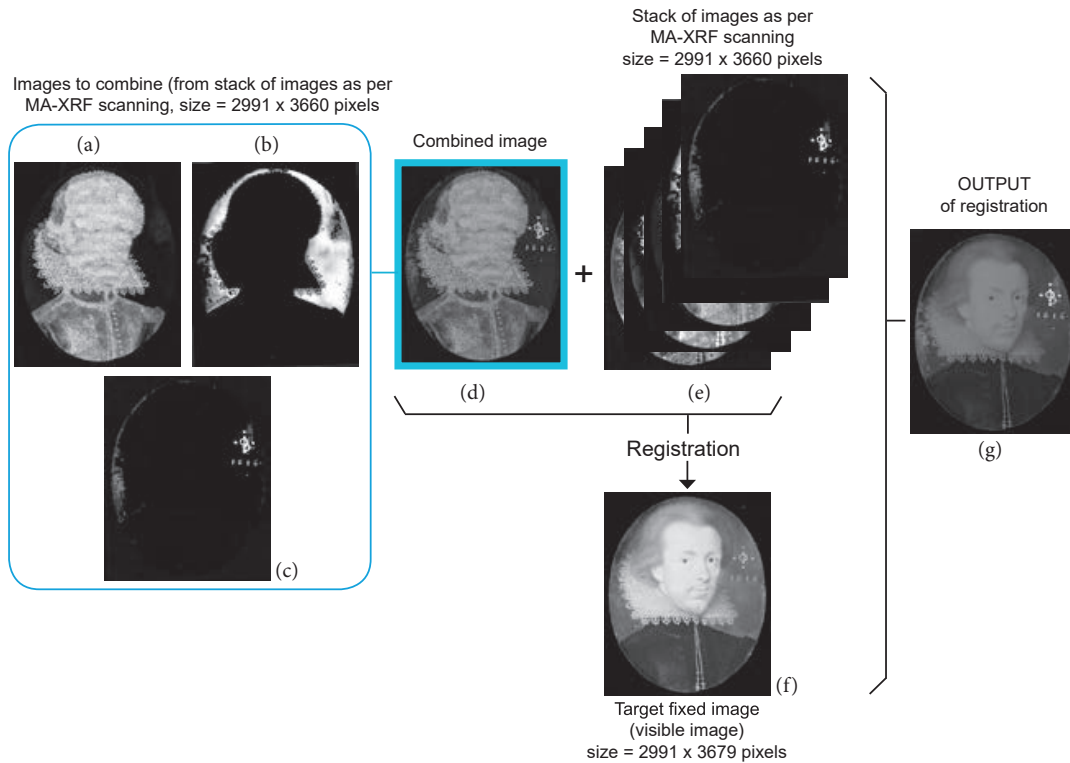


Figure 4.6: Registration of MA-XRF scanning of the *Unknown Man*, formerly called *Edward, Baron Herbert of Cherbury* (AGOID.29377, The Thomson Collection at the Art Gallery of Ontario) [112]: setup of the images for registration. Parts (a,b,c) show the enhanced resolution version of the element maps of lead ($\text{Pb-M}\alpha$), copper ($\text{Cu-K}\alpha$) and gold ($\text{Au-L}\alpha$) respectively. These images are combined to produce the *synthetic* image (part (d)) that is added to the image stack (part (e)) for registration to the target fixed image, which is the greyscale version of the visible image, at its original resolution (part (f)). Note that here the elemental maps obtained from the MA-XRF scanning of the artwork have been altered by increasing their original resolution to match the resolution of the visible image of the artwork. Thus, the moving image consists of the selected images obtained from the MA-XRF scanning, plus the additional *synthetic* image. The output is a unique best transformation, that is identified with consensus and can be applied to the full image stack. Part (g) shows the overlay of the visible image and the registered elemental map of gold. Images after The Hamilton Kerr Institute and The Fitzwilliam Museum [112].

while maintaining the individual details of each elemental map in the combination.

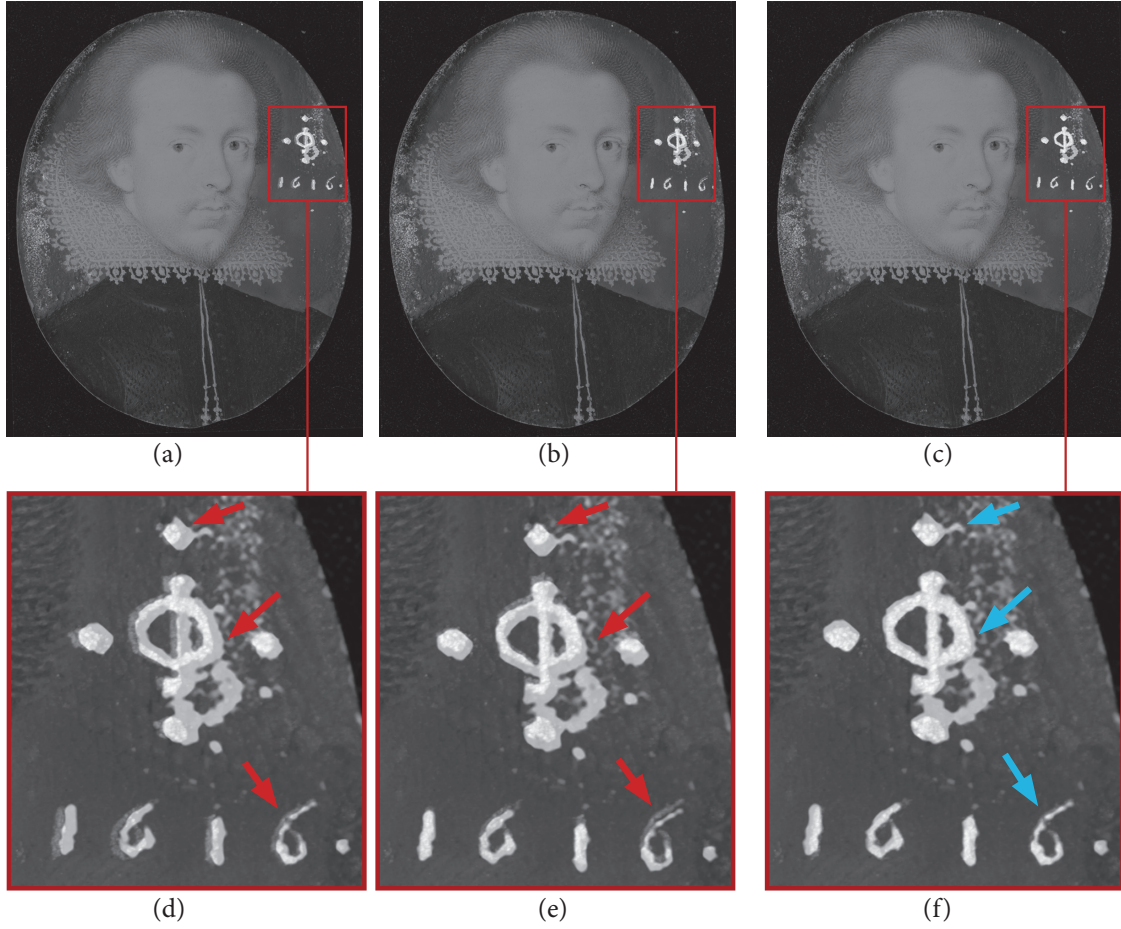


Figure 4.7: Registration of MA-XRF scanning of the *Unknown Man*, formerly called *Edward, Baron Herbert of Cherbury* (AGOID.29377, The Thomson Collection at the Art Gallery of Ontario) [112]: overlays of target image (visible image at its original resolution of 2991 x 3660 pixels) with element map of gold (enhanced resolution version). Parts(a, d) show the overall initial alignment of these images and a detail, respectively. Parts(b, e) show registration results (final alignment, after the fourth search of our method) obtained with the *standard* moving image of 15 images only, and parts(c, f) show corresponding registration results obtained with the *augmented* moving image composed of 15 images plus the *synthetic* additional image, resulting from the combination of the elemental maps of gold, lead and copper. Arrows in red show the misalignments that are overcome at the registration of the augmented moving image, shown here with blue arrows. Images after The Hamilton Kerr Institute and The Fitzwilliam Museum [112].

4.1.4 Comparison of results: registration with feature-based methods

Following the previous two experiments, in this section we show results for the registration of the images obtained with MA-XRF scanning of the *Portrait of Jane Boughton, née Coningsby* using feature-based methods. In particular, we show results after conducting registration of these images with the method proposed by Conover et al. [11], which follows the procedure presented in Section 2.3.4.

Since this method is designed to register only one image at a time (e.g. does not cater for

obtaining consensus among multiple images in a stack), we proceeded to test three element maps that show patterns and shapes that bear significant resemblance with the visible image of the artwork (Fig. 4.8 (a), and images to register, element maps of gold (Au- $L\alpha$), lead (Pb- $M\alpha$) and copper (Cu- $K\alpha$), Figs. 4.8 (b, d, f) respectively).

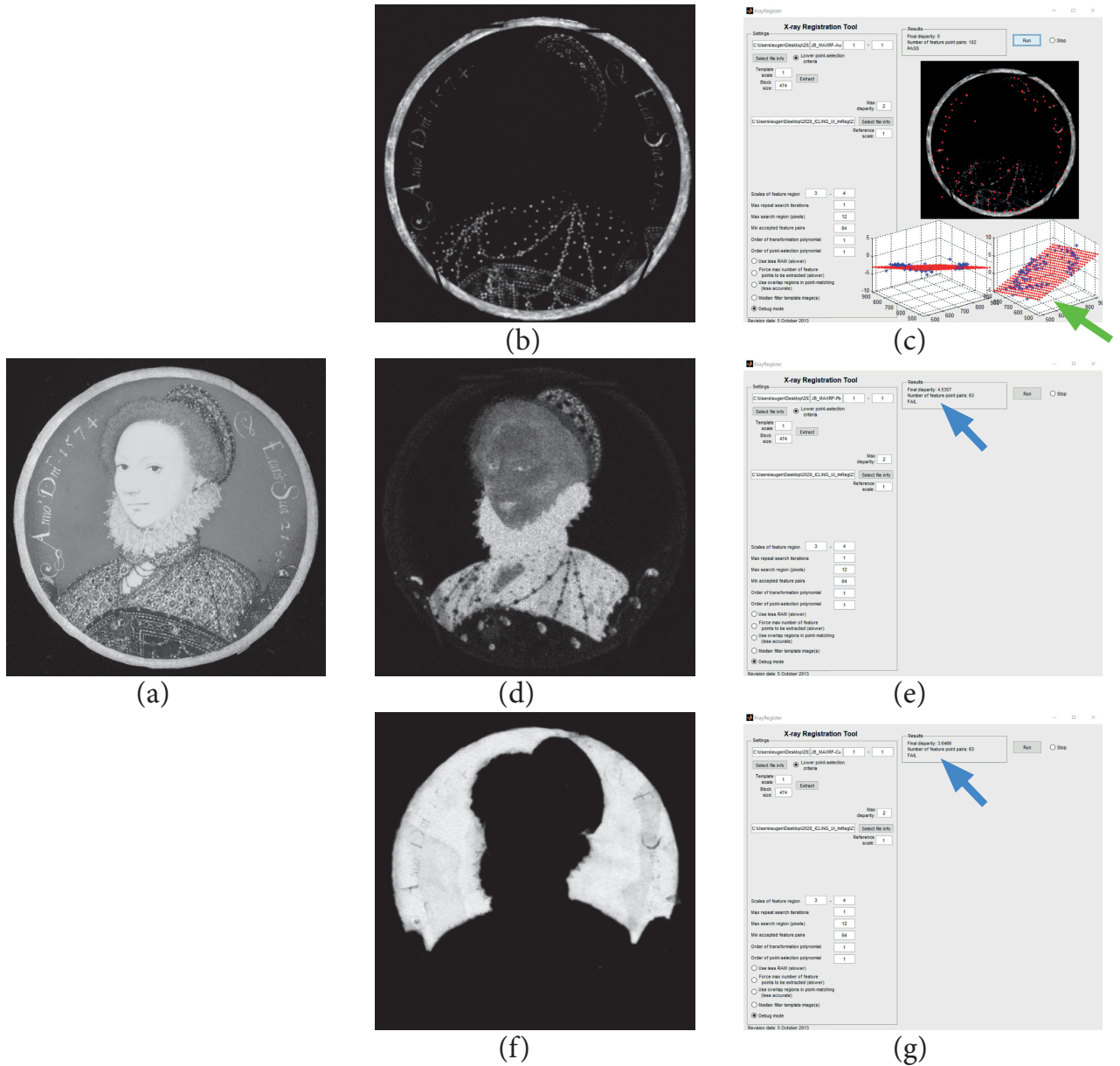


Figure 4.8: Registration of MA-XRF scanning of the *Portrait of Jane Boughton, née Coningsby*: Registration of element maps of gold (Au- $L\alpha$), lead (Pb- $M\alpha$) and copper (Cu- $K\alpha$), to the visible image of the artwork, shown in part (a). Parts (b, c) show the elemental map of gold and the resulting registration with the selection of feature pairs from the image (indicated with green arrow), where the registration was successful. However, Parts (d, e) and (f, g) show the elemental maps of lead and copper together with their "Results" message box where the registration fails to identify feature point pairs within the maximum disparity threshold (indicated with blue arrows).

This registration method gave good results for the element map of gold. In this case, the

algorithm was able to successfully identify matching feature points within the threshold required (set to 2 pixels for the case of this exercise, see *Max disparity* in Fig. 4.8 (c)). However, in the case of the elemental maps of lead and copper, the algorithm failed to identify matching points within this threshold ⁵ (Fig. 4.8 (e, g)). Our belief is that these elemental maps show large areas of similar intensities with no distinctive or salient features that could be identified in the same way that a minute detail such as craquelure would be (see Section 2.3.4). Whereas, in the case of the element map of gold, there is an abundance of point-like features that can be related to corresponding salient features in the visible image of the artwork ⁶. Thus, if addressing registration of technical images obtained from MA-XRF scanning, the success of this method would depend on the registration obtained for specific images within the stack that constitutes the moving image, and would not be able to establish a consensus among the images as proposed here, as not all the images would be able to identify valid transformation candidates.

4.2 Registration and mosaicking of images that capture partial areas of a 2D artwork

In this second set of experiments, we focus on the case study presented by the painting “*The Death of Actaeon*” (NG6420, The National Gallery, London) [115]. We consider the images obtained from a series of MA-XRF scanning of neighbouring areas of the artwork ⁷, for their registration to the visible image of the same artwork. The goal is to create technical images resembling the full extent of the scanned areas of the painting, by combining the corresponding element distribution maps of each scanned area that is registered to the visible image. This process is called *mosaicking* (see Section 1.1.2).

For the case of this artwork, the selected images are 12 element maps, namely: potassium (K - K lines), calcium (Ca - K lines), titanium (Ti - K lines), manganese (Mn - K lines), iron (Fe - K lines), cobalt (Co - K lines), copper (Cu - K lines), arsenic (As - K lines), barium (Ba - L lines), mercury (Hg - L lines), lead (Pb - L lines) and lead (Pb - M lines), (Fig. 4.9). Using the area-based method presented in Section 3, we show registration results for six scanned areas (Fig. 4.10). The resulting image stacks for each scanned area are of different sizes, ranging from about 150 x 950 pixels, to about 950 x 950 pixels. Here the pixel size was set to 580 μm , setting thus a lower spatial resolution of the element maps obtained than that of the Portrait miniatures (where pixel size was set to 100 μm), as well as than that of the visible image of the artwork.

⁵ A threshold of 2 pixels is a commonly used value to work with. Especially in this case, where the size of the technical images is 470 x 470 pixels, the threshold should not be higher than this value.

⁶ It is important to note that these technical images, of 470 x 470 pixels, are of a much lower resolution than other image modalities such as X-radiographs, hyperspectral images or infrared reflectograms, which are normally registered successfully with this method.

⁷ The MA-XRF scanning of this painting follow the areas as described in Fig. 4.10, allowing for an overlap of circa 5 cm of the margins of any two consecutive areas. Note that this is a fixed overlap rather than a percentage of the scanned area.

It is also important to note that the scanning of this painting was done in two campaigns, and this resulted in two different scanning setups leading possibly to a different alignment of the MA-XRF scanner relative to the painting. For this reason, the registration of these technical images was considered a particularly challenging case.

The resulting MA-XRF datacubes were processed to create the element maps (i.e. the moving image) using the "fast" automatic deconvolution algorithm developed by Su Yan et al. [27]. Note that this algorithm produces element distribution maps for a *set* of line-groups of each detected element (i.e. groups $L\alpha$ and $L\beta$ are combined into a single group L for the elements detected along the corresponding energy line (L)). The resulting images compose the image stacks that are treated as the moving images.

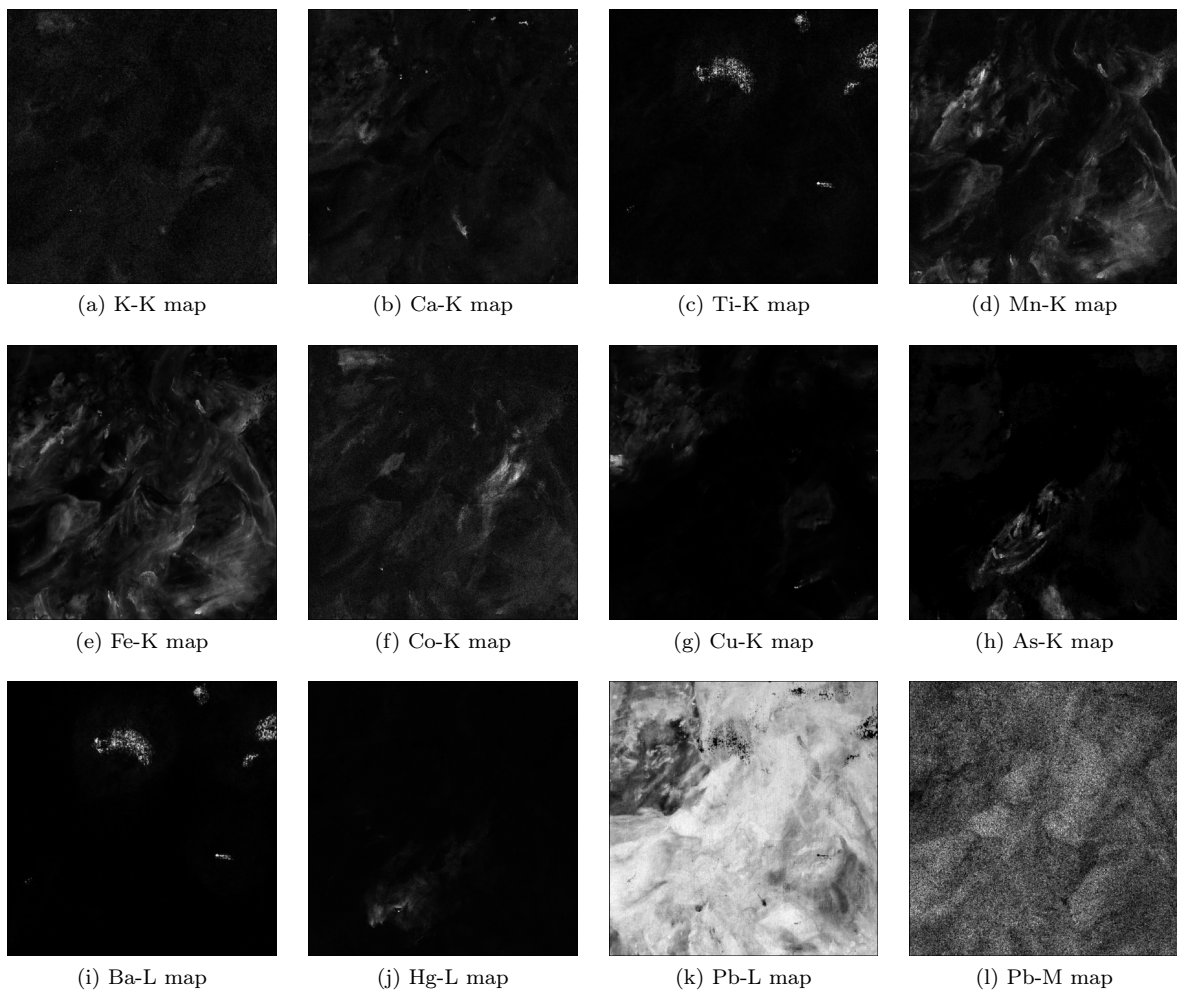


Figure 4.9: Element distribution maps: selection of 12 images derived from the MA-XRF scanning (area d02, see Fig. 4.10) of the “*The Death of Actaeon*” (NG6420, The National Gallery, London). These compose the image stack which we treat as the moving image.

A possible approach is to independently register each scanned area to its corresponding region of the visible image of the painting. In this case, for each scanned area of the painting, an image

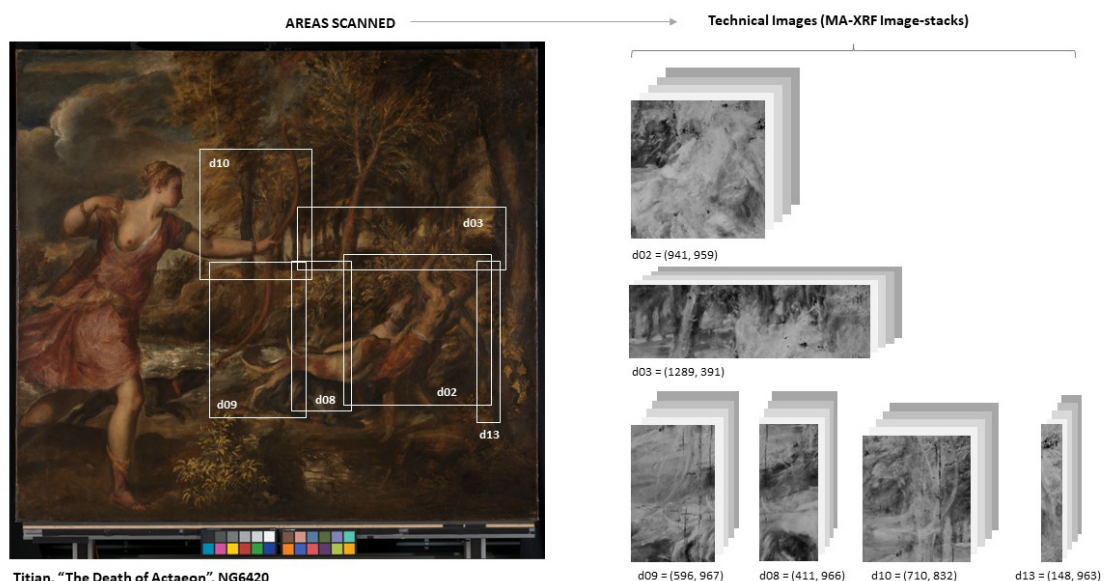


Figure 4.10: Full-visible image of the painting, scaled down to match approximately the original resolution of the scans (with white boxes indicating the positions of the scanned areas), and technical images (covering different regions of the artwork) derived from MA-XRF scanned areas d02, d03, d08, d09, d10, d13 (moving images, with element maps of lead (Pb-L lines) shown as representative top image of each image stack). Note that the numbering of the datacubes serves as an indicator that these areas were scanned in a non-consecutive manner.

stack is composed of the above-mentioned selection of element maps and thus treated as the moving image for its registration to the visible image of the painting only.

However, when collecting technical images from different areas of the same artwork, it may occur that some element maps contain more pictorial information than others. Some areas may cover regions of the painting where the composition might feature sharper edges, or contain well-defined, recognizable pictorial structures. If these well-defined structures are apparent in both moving and fixed images, a higher value of mutual information can be obtained during the registration process, thus helping to identify candidate transformations that may lead to a satisfactory alignment of the images. On the other hand, there might be cases where element maps derived from scanned areas contain features that might not be present in the visible image of the painting. In such cases, the features that may be apparent in the MA-XRF elemental maps of a given scanned area might relate more successfully to the features shown in the corresponding elemental maps of a contiguous scan (namely, in the overlapping regions of these scanned areas), rather than to the features of the corresponding region of the visible image. Thus, the registration results can be leveraged by using the information of neighbouring scanned areas. This is done by initially registering the images derived from scanned areas that are larger in size and with well-defined features, and then producing a *combined image* that will serve as the target fixed image to register the remaining, surrounding scanned areas. Thus in this experiment we *enhance* the target fixed image (by combining the visible image of the artwork with the information of a selected elemental

map), which contrasts with the approach of enhancing the image stack with a synthetic image as implemented in the previous experiment (described in Section 4.1.3).

4.2.1 Registration of image stacks to the visible image of artwork

In this example, the first image stacks to be registered to the visible image are the ones corresponding to areas d02 and d09, as shown in Fig. 4.11. Using the area-based method presented in Section 3, a best transformation was found and applied to these image stacks. In the case of this painting, not only image stacks d02 and d09 contain images with well-defined features that are apparent in the visible image of the artwork, but also they are surrounded by other scanned areas, as they are located at the centre of the visible image of the painting. Thus, their initial registration may help to inform the registration process of the remaining, neighbouring image stacks. Note that this is a painting with several changes in the composition, resulting in the elemental maps showing information that is not visible at the surface of the final artwork [115].

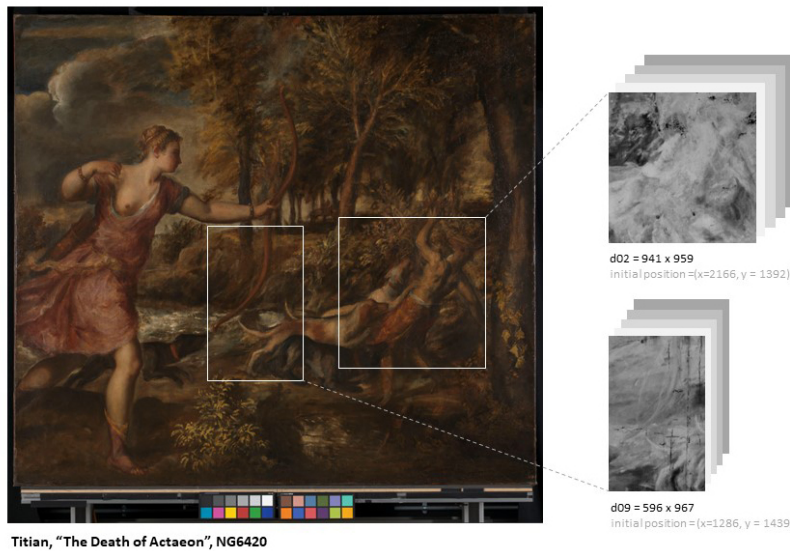


Figure 4.11: Initial registration of image stacks d02 and d09 to the visible image. This can be used to inform the registration process of the remaining, neighbouring image stacks. Element maps of lead (Pb-L lines) are shown.

4.2.2 Registration of image stacks to the visible image of artwork with additional information of neighbouring image stacks

Once the initial image stacks are registered to the visible image of the painting, a new target fixed image based on the *original* visible image of the painting is produced. This is done by combining the visible image of the painting with additional information from the registered image stacks d02 and d09. For the case of this painting, we select the element map of lead (L-lines) since it has

recognizable pictorial structures⁸. We combine these images by taking the pixel values of this element map at the locations of the “transformed” image stacks d02 and d09, and add these to the pixel values at the corresponding locations within the visible image of the painting. In this way, a new *combined* image of the painting is produced.

This combined image may now be used to register the image stacks corresponding to the scanned areas surrounding d02 and d09 (Fig. 4.12). Using the area-based method presented in Section 3, image stacks of areas d03, d08, d10, and d13 are registered to this combined image.

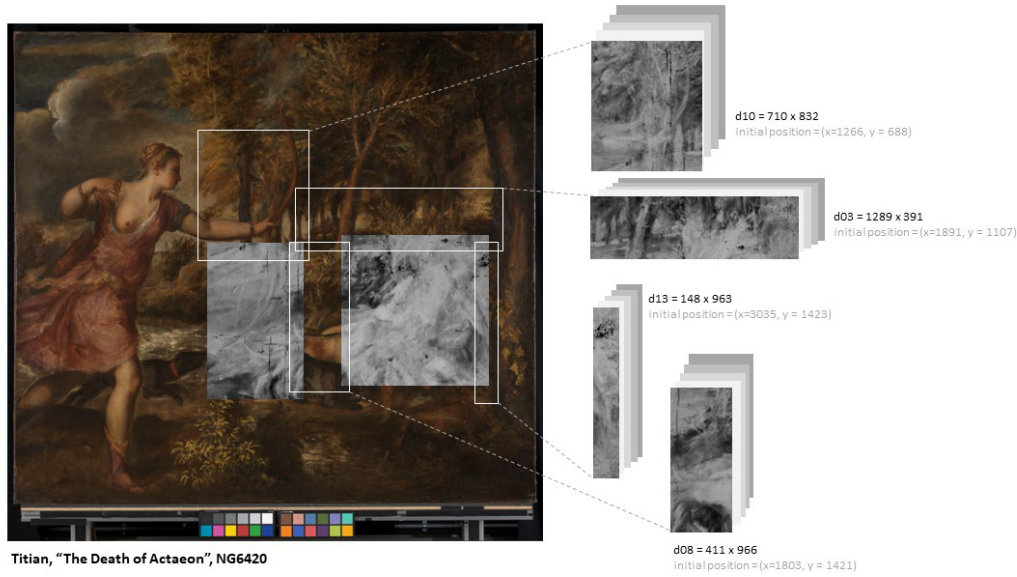


Figure 4.12: Registration of image stacks d03, d08, d10 and d13, to their corresponding regions of the visible image plus additional information of image stacks d02 and d09. The element maps of lead (Pb-L lines) are shown for reference as the first image of the neighbouring image stacks.

The method is applied to all remaining neighbouring image stacks, one at a time, so that they become registered to the full visible image of the painting. For each of these image stacks, the target image is created as follows: firstly, the full visible image of the artwork is combined with the initially registered elemental maps of choice as described above. Then, from this combined image representative of the entire artwork, a series of regions corresponding to each individual scanned area are selected to serve as the individual target fixed images, respectively. For this purpose, the combined image becomes an “enhanced” version of the visible image of the painting, by combining the latter with additional information from the images of interest (in this case, the elemental distribution map of lead of the initially registered scanned areas d02 and d09).

Once the remaining image stacks are registered to the target fixed image, it possible to create

⁸ Note that the distribution map of lead is also a good one to see the pentimenti as lots of the changes done by the artist are to do with positioning, for example, of the figures or limbs of characters.



Figure 4.13: Registration of MA-XRF scanning of “*The Death of Actaeon*”: *mosaic* images of lead maps (Pb-L lines) of scanned areas d02, d03, d08, d09, d10 and d13 registered to visible image plus additional information of registered scanned areas d02 and d09. The mosaic image is shown overlayed to greyscale version of visible fixed image.

a *mosaic* image of the technical images belonging to all six scanned areas (Fig. 4.13), and assess not only their individual registration to the visible image of the painting, but also the overlapping areas between any two consecutive technical images. The benefits of using the information of neighbouring image stacks can be seen in the comparisons presented in Fig. 4.14 and Fig. 4.15. By initially registering the image stacks d02 and d09, the continuity between any two consecutive registered technical images is noticeably improved.

4.3 Runtime performance

The experiments presented in this chapter were conducted with an Intel core i7-7700 CPU @ 2.80GHz, and 32GB RAM. Registration with our area-based method requires under 5 minutes using this CPU, for a stack of 15 images of about 500 x 500 pixels in size (as it is in the case of the experiments conducted with the Portrait miniature of *Portrait of Jane Boughton, née Coningsby* (AGOID.29371, The Thomson Collection at the Art Gallery of Ontario) [112], shown in

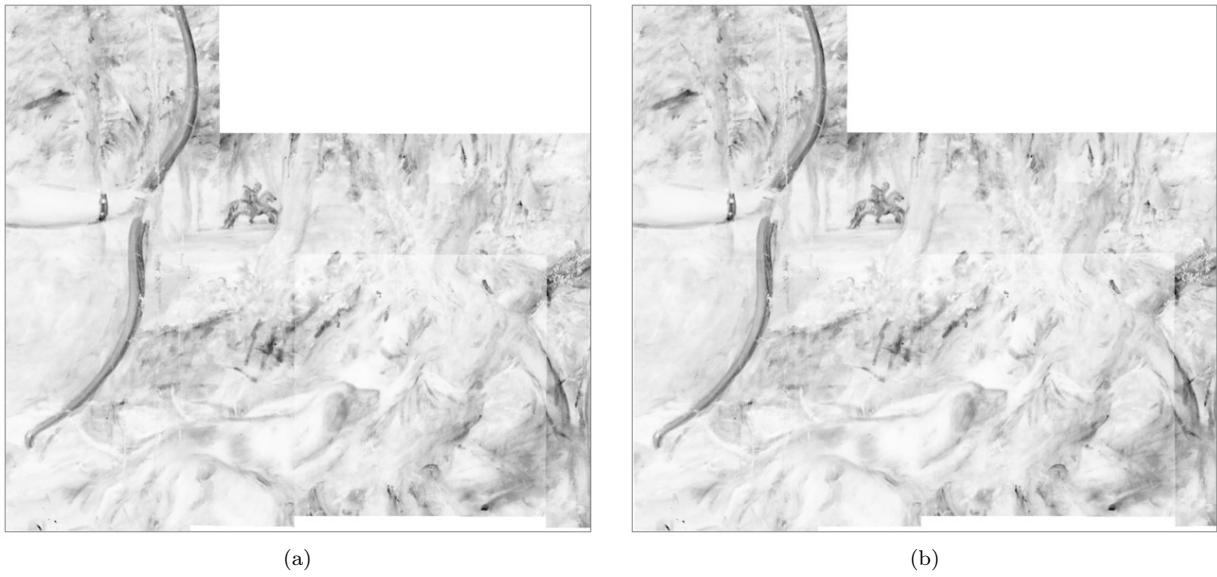


Figure 4.14: Registration of MA-XRF scanning of “*The Death of Actaeon*”: *mosaic* images of iron maps (Fe-K lines, inverted for clarity at visual inspection) of scanned areas d02, d03, d08, d09, d10 and d13 registered to visible image. Comparison of registration to target image with information of visible image only (a), and registration to target image with information of the visible image plus additional information of registered scanned areas d02 and d09 (b). Note that these mosaicked images were obtained by simply layering the registered moving images one on top of another. See Section 5.4 for a detailed description of visualisation of results.

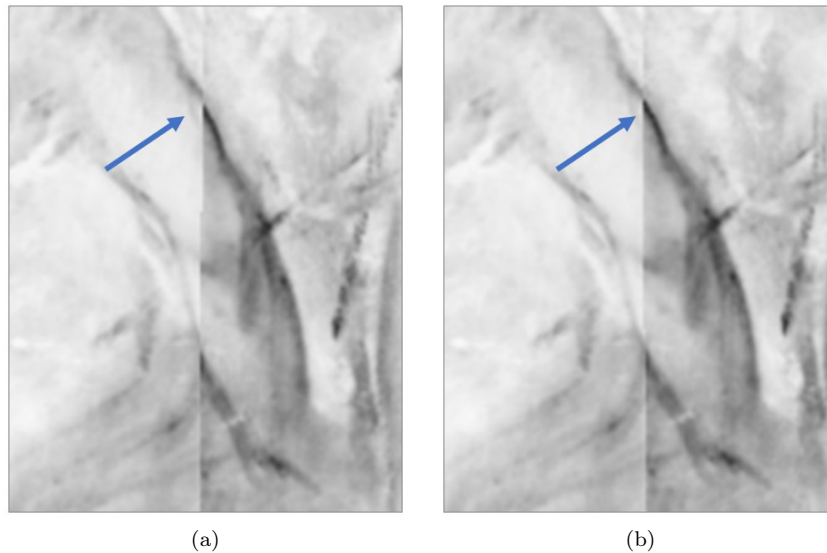


Figure 4.15: Detail of registration of MA-XRF scanning of “*The Death of Actaeon*”: *mosaic* images of iron maps (Fe-K lines, inverted for clarity at visual inspection) of scanned areas d02 and d13 registered to visible image. Comparison of registration to target image with information of visible image only (a), and registration to target image with information of the visible image plus additional information of registered scanned areas d02 and d09 (b). Blue arrows indicate details where improvement of the registration is particularly noticeable.

Section 4.1). This is a guideline only, since the time required for registration is naturally subject to the size of the image stack and resolution of the working images.

4.4 Discussion

Throughout the experiments shown in this section we note that this area-based method is highly sensitive to the initial location that is assigned to the moving image at the beginning of the process. The task of finding this best initial location (assigning a translation along x and y axis respectively, an initial angle of rotation and an initial scaling factor so that the moving image is positioned as close as possible to its best alignment to the visible fixed image) may require a lot of the user's attention. The outcome of the method depends much on this aspect: the better the initial location of the moving image, the more accurate the registration to the visible image will be. Thus, setting a good initial position is key to the success of the registration ⁹.

Whether using an automated method, or conducting the registration manually, is it customary that an initial location ¹⁰ would be found by means of inspecting the images to align in an image processing software package such as GIMP [15] or Photoshop [14]; and evaluating their overlap by manually arranging the images until an initial location, that the user is satisfied with, is found. This can be time-consuming and entail extra steps in the registration workflow. Thus, providing the user with an integrated and effective way to identify a good initial position was one important factor to consider in the process of developing the Graphical User Interface that is presented in detail in the next chapter.

4.5 Summary of this chapter

In this chapter we reviewed the practical application of the workflow presented in Section 3 for registration of technical images of artworks obtained from MA-XRF scanning to the visible image of the same artwork. This chapter started by addressing the registration of technical images that capture the entire artwork in one MA-XRF scanning. Within this case, we reviewed results of registration using a *standard* image stack composed of the element distribution maps obtained from MA-XRF scanning only, as well as results of registration using an *augmented* image stack composed of the same images plus an additional *synthetic* image resulting from the combination of two or more images from the original, standard image stack.

⁹ Note that, whilst the user needs to assign the parameters for an best initial location, the final outcome of the method is a composite transformation. This is composed of an affine transformation (resulting from the method's third search, see Section 3.4) plus a non-rigid transformation (resulting from the method's fourth search, see Section 3.5) that is applied to the moving image for its registration to the visible image. This composite transformation is "built" on top of these initial parameters assigned by the user for the best initial location.

¹⁰ Note that this requirement stands for most methods, especially the ones used in Heritage Science, such as the method developed by Conover et al. [11], ImageCompare [12], or the registration method developed by the Conservation and Science Department of the Rijksmuseum [111].

This was followed by addressing the case where the artwork is captured in more than one MA-XRF scan. Here, the resulting image stacks need to be registered individually to the visible image of the artwork, for the purpose of generating a *mosaic* technical image (here a set of MA-XRF element distribution maps) showing the full extent of the scanned areas of the artwork. Within this second case, we reviewed results of registration using the *standard* visible image as the target fixed image, as well as results of registration using an *augmented* visible image resulting from the combination of the original visible image plus the information from MA-XRF element distribution maps that are initially registered to it.

This chapter concluded with a discussion of the runtime performance, and the insight that finding a good initial location of the moving image within the visible fixed image is a determining factor for a successful registration. Yet this can be a time-consuming task, and this being a manual step, it may depend much on the skill of the user. With the aim of simplifying this process of setting up the initial locations, as well as unifying all of the steps involved in using this automated area-based method, a Graphical User Interface (GUI) has been developed to provide a user-friendly browser-based tool. This GUI is presented next in Chapter 5.

Chapter 5

Graphical User Interface (GUI)

This chapter showcases the graphical user interface (GUI) developed for the area-based registration method presented in Chapter 3. It starts by describing the steps of the workflow required to perform the registration of a set of images: from the selection and alignment of the input images, to the creation of a *settings* file containing all the corresponding information (i.e. images to align and their assigned initial positions) that is required to perform the registration with our automated method. The chapter then discusses the requirements and constraints that motivated the technical decisions that are at the basis of this application to ensure it best meets end-user needs. The overarching goal was to develop an application that could run in the browser of any computer, e.g. a *browser-based* application. We then conclude this chapter by discussing further work that would be ideal to address within the application such as the inclusion of image processing tools for more detailed image analysis (e.g. histograms to help with the visualization of individual images) or manipulation (e.g. Gaussian filters, pixel averaging) to obtain smoother transitions between overlapping consecutive images of potentially different intensity values in mosaicked or composite images.

5.1 Introduction

The purpose of developing this GUI is to simplify the use of the area-based registration method described in Chapter 3 by making the functionality accessible to the user through the browser (thus removing the need for the user to interface with the code directly), as well as to facilitate the process of setting up the initial locations of multiple image stacks relative to a target fixed image in a streamlined workflow. Thus, developing the GUI as a *browser-based* application allows any user to perform a registration, without prior training in coding or image processing techniques. Additionally, it allows the user to access the application from any computer, without the need to install it on a personal computer. An overview of the steps implemented by our GUI is presented in Fig. 5.1.

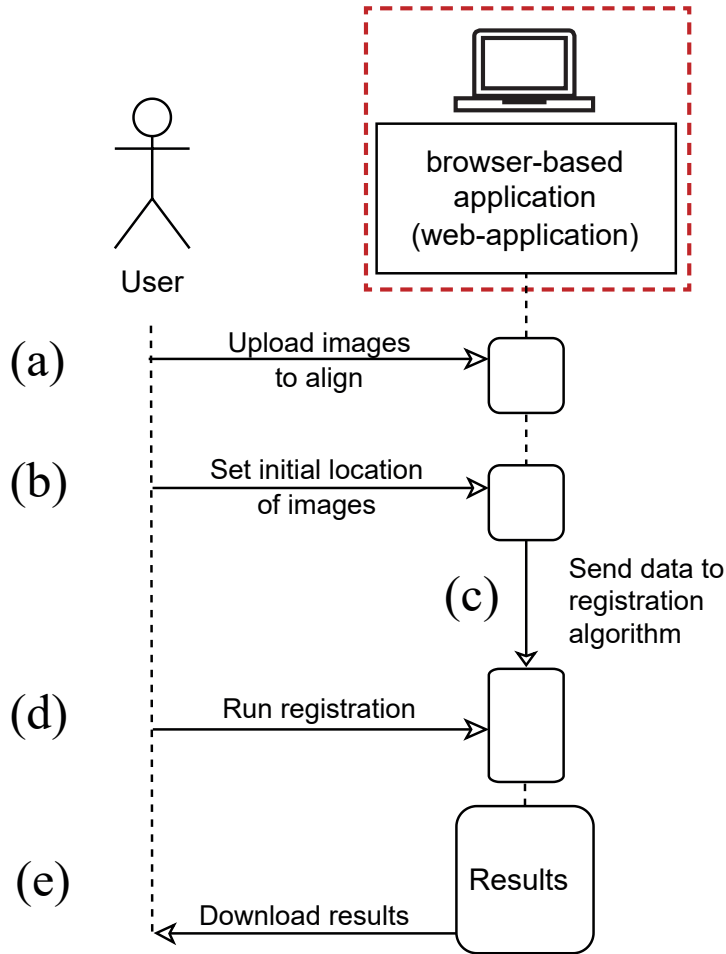


Figure 5.1: Graphical user interface (GUI): Overview of the steps involved in our GUI. Part(a) illustrates the user uploading the target and moving images; part(b) illustrates the user setting the initial parameters of the images to align. Once data is then passed to the registration algorithm (part (c)), the user may instruct the execution of the registration (part(d)). Finally, the user can view and download the results and registered images.

As mentioned previously in Section 4.4, the initial locations of the images of interest would be normally set up manually in an image processing software package such as GIMP, Photoshop or Inkscape. Once the user is satisfied with the position of the moving image relative to the fixed image, the parameters would be noted down manually and passed as input to the automated registration method of choice. Alternatively, the user would proceed with the registration in the same software in a manual manner.

To execute our automated area-based method (see Section 3.4), an initial one-off input containing the parameters of the initial locations assigned by the user for each moving image as well as for the fixed image is required. In order to facilitate this vital initial alignment step and pass the starting information to the algorithm (and thus avoid having to manually transfer this information and editing the underlying code of the algorithm), the GUI is designed so as to allow the user to

select the images to align and designate the initial parameters to apply to them - such as translation along x and y axis, rotation and scaling. These parameters constitute the initial locations of the images to align. Once these parameters are defined for all the images of interest, they are saved into a *settings* file, which is then used as the input to start the registration process.

The development of the GUI has been undertaken in two stages. A first GUI was developed that undertook the above-mentioned setting-up steps (i.e. creation of the *settings* file) only and this is described in Section 5.2 and Appendix A.

A more sophisticated, second GUI that is able to undertake the full workflow was subsequently developed, into which all of the functionality of the initial first GUI has been incorporated (see Fig. 5.6). This is described in section 5.3. The technical reasons for this approach to the development of the GUI are described in Section 5.1.1.

5.1.1 Software and accessibility constraints: client-server architecture

As we wanted to develop a GUI that would allow any user to perform a registration (i.e. without prior training in coding or image processing techniques), and to be able to access the application from any computer (i.e. without the need to install it on a personal computer), we focused on the development of a browser-based application, with an intuitive interface that would be freely accessible from any computer connected to the internet.

However, our area-based method is written in python, leveraging SimpleITK, which is an image processing library developed for the medical imaging sector and is written in C++. This imposes some difficulties to directly run the registration algorithm in a browser, due to the fact that python and C++ do not run in a browser environment. This issue motivated us to build an intermediary that executes the registration algorithm on behalf of the browser-based application. This, in turn, implies the need for a *client-server* architecture¹. Please refer to Fig. 5.2 for an overview of this setup.

The browser-based application constitutes the *client*, and the intermediary (the piece of software executing the area-based method) constitutes the *server* application. This is a flexible architecture that allows the *server* to be installed on any computer: it can be installed on the user's local computer, or on a remote computer (i.e. in the cloud or on another on-premise computer). Note that this type of architecture allows the user to instruct registrations as a queue of work to be executed by the server in series or in parallel, independently to the activity of the user in the browser-based *client* part of the GUI.

¹ In this case, a *client* refers to a piece of computer software that interacts with a *server* (which is another piece of software which can reside in another computer system, and may liaise with the client by means of a network). These are the fundamentals of the client-server model of computer networks. [116]

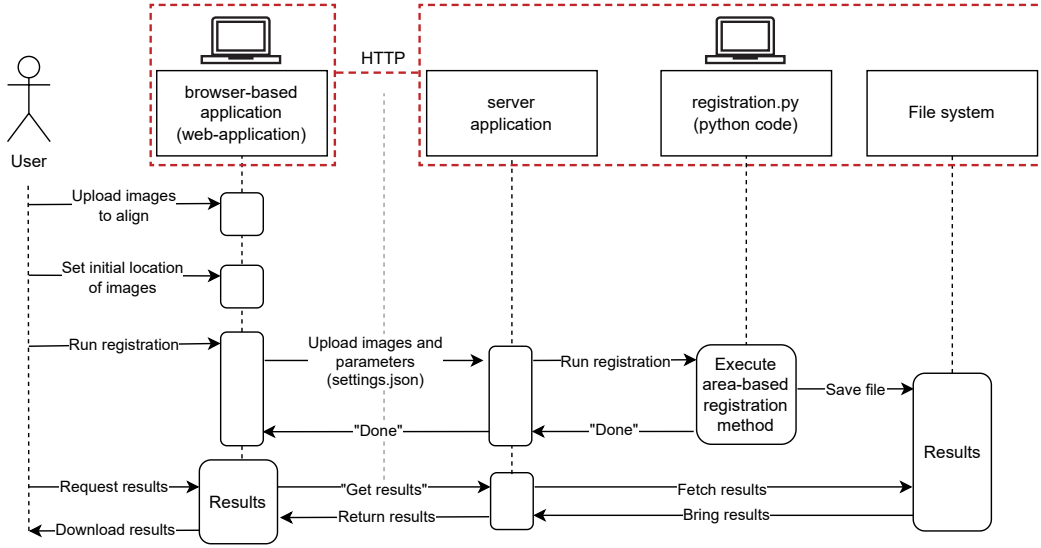


Figure 5.2: Graphical user interface (GUI): client-server architecture. This sequence shows the interaction of the user with the client (browser-based application), which in turn establishes communications with the server application for executing the registration and saving results in the corresponding file system. In this setup, the user instructs the client to run the registration. The client, in turn, instructs the server to do this on its behalf. Once the registration is completed, the server notifies the client, which in turn notifies the user. The user can then proceed to request and download the results from the client.

5.2 Browser-based *client* application: interface for selecting images in the browser and setup of initial locations

Initially, the GUI developed consisted of a browser-based application that allowed the users to select the images to align and designate the initial parameters to apply to them - such as translation along x and y axis, rotation and scaling. Once these parameters are defined for all the images opened in the GUI, they can be saved into a *settings* file that can then be used as an input to start the registration process. Please refer to Fig. 5.3 for an overview of the first GUI, and to Appendix A for a detailed user guide on how to navigate this first version of the GUI *client* application.

Our GUI facilitates the process of setting up the initial locations of multiple image stacks relative to a target fixed image by enabling the possibility of setting the initial location of multiple moving images in the same interface. The user can thus visually inspect all the images jointly and automatically export these parameters, together with the information of all images selected, to a single *settings* file. This, in turn, facilitates the process of potentially evaluating alternative configurations (such as different initial locations of the moving images) as the *settings* file can also be saved as work-in-progress and continued later at another session.

Please refer to Appendix A.1 for a detailed overview on how to assign the initial parameters to the images to align and create the *settings* file. As mentioned earlier, the purpose of developing this first GUI is to simplify the process by which the images to register are set up with their

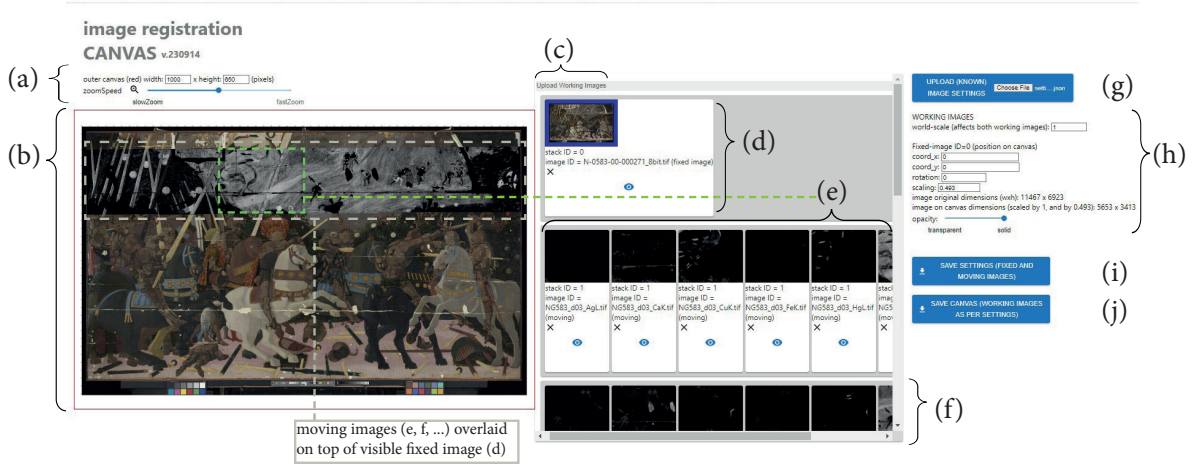


Figure 5.3: Initial graphical user interface (*first GUI*). This image shows the interface of the browser-based application. Here, this interface shows the setup of the initial locations that are assigned by the user to each moving image in relation to the target fixed image (here the visible image of the painting “*The Battle of San Romano*”, NG583, The National Gallery, London [117]). Part(a) shows the dimensions (set by the user, in this case 1000 x 650 pixels) of the *canvas*, shown in (b), which is the window with red edges for visualising the workspace with the images that are selected for alignment. Part(c) shows the *uploading working images* button, which enables the user to select, firstly, the fixed image, and after that at each consecutive time, the images that constitute each moving image. Part (d) shows the fixed image and its settings, and parts (e,f) show the moving images (represented in the canvas by the element maps of lead (Pb-L), on top of the visible image) and its settings. Note that each moving image is an image stack (resulting from each individual scan area), composed of individual images. For visualisation, these individual images are displayed in a row in the GUI (parts (e,f)). However, when registered they are placed one on top of the other, since they all belong to the same image stack. All images belonging to the same stack have to be of the same size, and are assumed aligned by construction. Note that the fixed image (which is the first one to be selected) remains in the background, and each consecutive moving image is placed *on top* of it. Part(h) shows the dialog boxes where the user can input the parameters for the image that is selected in the workspace (in this case, the fixed image is selected and this is noted by the blue-edge around its representative thumbnail). Part(i) shows the *save settings* button which collates all the information pertinent to the images as displayed in the canvas (i.e. it collates the actual images and the parameters assigned to each of them by the user) and downloads this information into a *settings* file, which is a *JavaScript Object Notation (JSON)* format file used by the server to then execute the registration of these images following the method described in Section 3. Part(j) shows the *save canvas* button which allows the user to save the image currently visible in the canvas window, in a similar manner to that of creating a screenshot of the current configuration of the workspace (showing, in particular, the fixed and moving images following the parameters assigned). Part(g) shows the *upload known settings* button, by which the user can open a settings file and continue the initialisation or change the locations of a previously saved configuration.

initial locations prior to commencing the registration method. The GUI thus becomes a platform for visual inspection of these parameters, which inform the first search of the area-based method and thus are key to the success of the registration (as discussed in Section 4.4). Following the description provided at Fig. 5.3, the user can operate in the GUI in the following way:

- Define the working space, called *canvas*. This is a window shown with red edges on the left side of the web application (Fig. 5.3 parts(a,b)).
- Select fixed image (Fig. 5.3 part(c)). The GUI is designed to treat the first image received as the target fixed image. This is treated as a single image in the background and this is usually the visible image of the artwork (Fig. 5.3 part(d)).
- Select moving images (Fig. 5.3 part(c)). The GUI is designed to compute multiple moving images, where each individual moving image is, in fact, an image stack composed of one or multiple images (Fig. 5.3 parts(e, f)). It is considered that all images in the same stack are aligned by construction.
- Set up initial locations for the fixed image and for each moving image (Fig. 5.3 part(h)). The initial location is set up by defining the translation along x and y coordinates, an angle of rotation (around the top left corner of the image) and a scaling factor. Note that these parameters are the same for all the individual images in the stack that compose the moving image. Any individual image of the stack can be made translucent upon selection (clicking on *eye icon* underneath the corresponding thumbnail) to help with the positioning of the image stack, relative to the target fixed image in the background.
- Once the initial locations are assigned, the user can save this information into a *settings* file (Fig. 5.3 part(i)). This file serves as the single input to the registration method, as it contains the information of each image opened in the session, with their corresponding initial locations (i.e. the parameters to inform the first search of the area-based method as described in Section 3.4). Note that this file can also be saved as work-in-progress and continued later at another session. See Fig. 5.4 for a detailed overview of this file.
- Once the method concludes the registration, the results can be accessed from the browser or alternatively saved in local folders determined by the server.

5.2.1 Format of images for registration and limitations on image size

Throughout the development of this GUI we have focused on TIFF (8-bit and 16-bit) and PNG image formats, since these are commonly used in art investigation due to the fact that these are lossless compression formats. Note that the method also supports other image formats such as BitMap and JPG since they are supported by the browser.

Note that, in principle, there is no limit on how many images a user can open in the GUI. However, due to the memory-limit in current browsers such as Chrome or Safari which is about 2 Gb (on the 64 bit browser version), the total amount of images selected may need to be restricted. In our experiments, working with images totaling 500 MB has provided good performance in terms of display refresh rate and incorporating changes to the GUI settings.

```

settings.json
{
  fixed_image : {
    fixed_image.tiff,
    parameters : {
      translation(x), translation(y), scaling factor s, angle of rotation r
    }
  },
  moving_images : [
    {name: moving_image_1,
      image_stack : {
        elemental_map_1.tiff, elemental_map_2.tiff, ..., elemental_map_n.tiff
      },
      parameters : {
        translation(x)1, translation(y)1, scaling factor s1, angle of rotation r1
      }
    },
    {name: moving_image_2,
      image_stack : {
        elemental_map_1.tiff, elemental_map_2.tiff, ..., elemental_map_n.tiff
      },
      parameters : {
        translation(x)2, translation(y)2, scaling factor s2, angle of rotation r2
      }
    }
  ]
}

```

Figure 5.4: This image represents the structure of the *settings* file that is created by the GUI application following the parameters assigned by the user to the moving and fixed images. This file can be saved at any point of the assignment and accessed for modifications at a later instance, or can be used to perform a registration with the area-based method in an automated manner. Note that the fixed image entails only one image and its corresponding parameters, whereas the moving images consist of an image stack and the corresponding parameters affecting all images in the stack. For the purpose of simplicity, only two moving images (*moving_image_1* and *moving_image_2*) are shown in this example. Importantly, this file, read by the registration algorithm, is both machine and human readable - the user can inspect its content via any standard free text editor such as Geany (<https://www.geany.org/>) or NotePad++ (<https://notepad-plus-plus.org/>).

5.3 Execution of the area-based registration method

Once the parameters of the initial locations are defined and saved into a *settings* file, the registration with our area-based method can start with this file as the only input.

The registration method can be executed by a server application (following the client-server architecture described in Fig. 5.2 and corresponds to the second version of the GUI, see Fig. 5.6), or alternatively, it can be executed directly by the user in the same local computer (following the

workflow illustrated in Fig. 5.5), corresponding to the initial *first* GUI as illustrated in Fig. 5.3).

In any of these cases, the intermediary (either the server or the user) instructs the method with the *settings* file, and the method runs in an automated manner. Once the registration is completed, the method notifies the intermediary. The method then saves the results in a designated folder in the local file system for future retrieval. The user can then retrieve these results by downloading them from the browser or by accessing them from the above-mentioned file system folder directly.

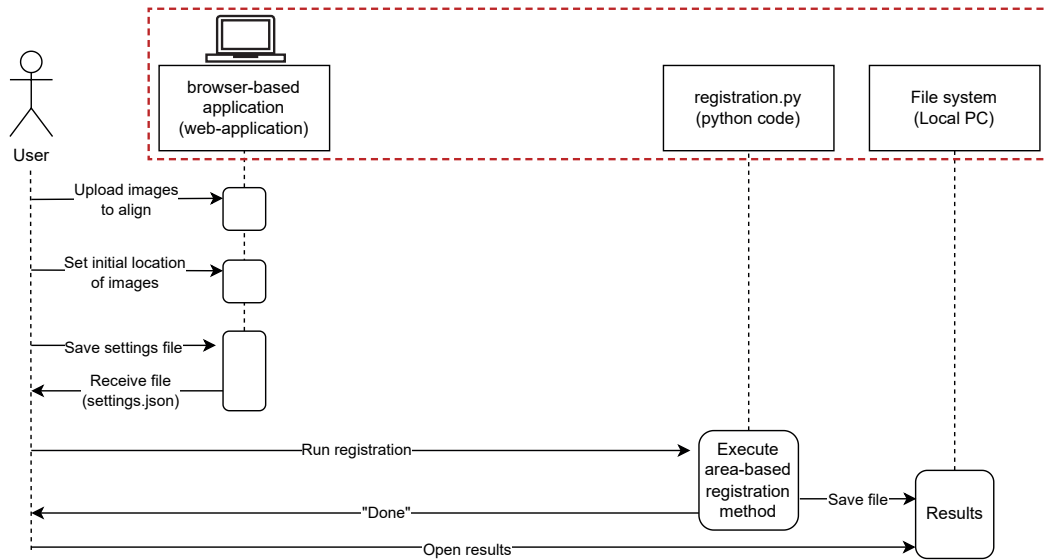


Figure 5.5: Graphical user interface (GUI). This block diagram shows the interaction of the user with the browser-based application to create a *settings* file, that can then be used to perform a registration with the area-based method in an automated manner. In this setup, the user instructs the method to run the registration. The method, in turn, runs the registration and saves the results in the corresponding local file system. Once the registration is completed, the method notifies the user. The user can then proceed to fetch the results from the corresponding local folder.

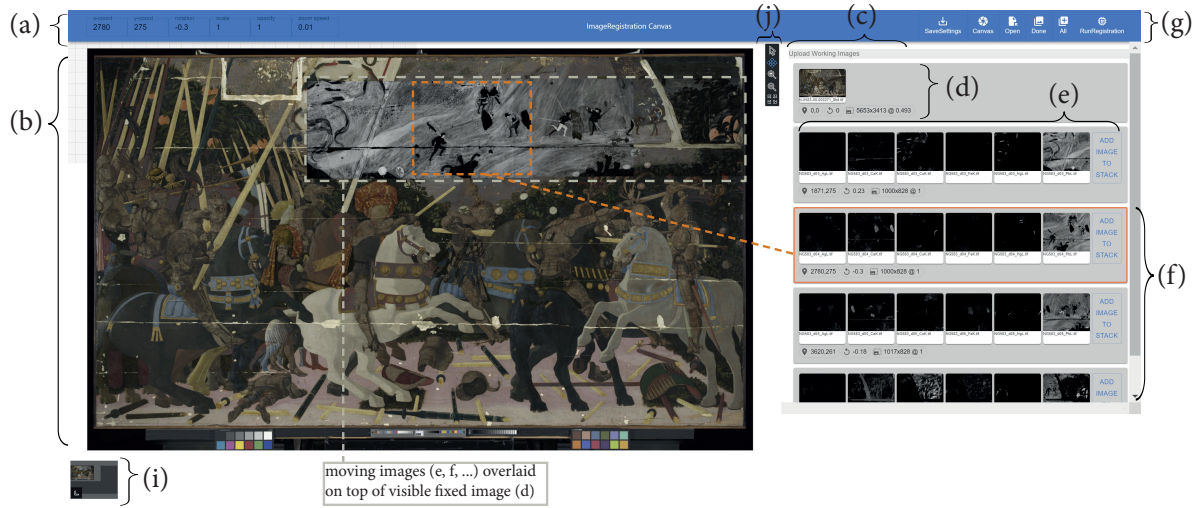


Figure 5.6: Graphical user interface (*second GUI*). This more sophisticated version is able to undertake the full registration workflow, incorporating all of the functionality of the initial first GUI. Here, the user can set up the initial locations of the images to align (here the technical and visible images of the painting “*The Battle of San Romano*”, NG583, The National Gallery, London [117]), run the registration and access the results. Part(a) shows the parameters of the image selected (edged in orange) as they are set by the user by assigning values into these dialog boxes, or by dragging the image in the *canvas*, shown in (b), which is the area for visualising the workspace with the images that are selected for alignment. Part(c) shows the *uploading working images* button. Part (d) shows the fixed image and its settings, and parts (e,f) show the moving images (represented in the canvas by the element maps of lead (Pb-L), on top of the visible image) and its settings. Note that each moving image is an image stack (resulting from each individual scan area), composed of individual images. For visualisation, these individual images are displayed in a row in the GUI (parts (e,f)). However, when registered they are placed one on top of the other, since they all belong to the same image stack. All images belonging to the same stack have to be of the same size, and are assumed aligned by construction. Note that the fixed image (which is the first one to be selected) remains in the background, and each consecutive moving image is placed *on top* of it. Part(g) shows the main commands of the application, in particular, the *save settings* button, the *save canvas* button and the *upload known settings* button. These buttons have the same functionalities as in the first version of the GUI (see Fig 5.3). This second GUI includes a *RunRegistration* functionality, supported by buttons that show all the registration jobs that are currently in progress and those that have completed registration successfully (i.e. the *All* button and the *Done* button). Finally, this version of the GUI features a small *mini-map* (part i), showing a small-size image of the overall image layout, giving the user a sense of orientation when inspecting the images in this interface.

5.4 Visualisation of results

The results produced by our area-based method consist of the transformations applied to the moving images (i.e. text-files indicating the values utilised for each parameter of the transformations applied in the final search of our area-based method - see Section 3.5), and the corresponding registered images (i.e. image files that are representative of the *transformed* moving images). The transformed moving images are saved in PNG format (as a stack of individual element distribution maps with transparent "padding" to match the size of the visible fixed image, see Fig.5.7 (a,b)),

so that they can be overlapped to the visible image of the artwork in a layered manner (see Fig.5.7 (c,d)). In this way, the user can select an element map from each transformed moving image (i.e. the same image across all registered image stacks) and combine them to create a *mosaic* technical image of the element distribution map for the desired element (see Section 1.1.2). Due to the layered nature of the resulting mosaic image, the user has the choice to toggle on and off and/or alter the transparency of each layer to inspect the quality of the registration obtained. The user can also combine the transformed moving images without the fixed visible image, thus obtaining a mosaic technical image with plain background (see Fig.5.7(e)).

Please refer to Appendix A for a detailed overview on how to access these results, and in particular, to Fig. A.19 for an illustration of the layout of the layered mosaic image. In Fig.5.3 and Appendix A, the GUI that was developed initially is presented and discussed but the same functionality also exists within the subsequent GUI that also calls and runs the registration algorithm.

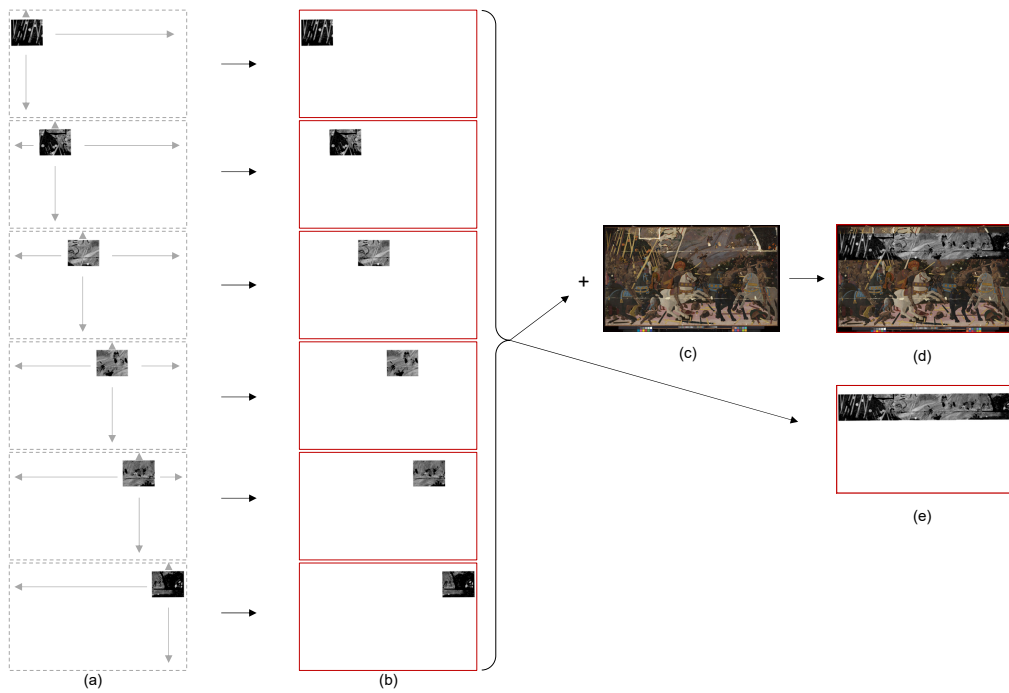


Figure 5.7: Graphical user interface (GUI): Visualisation of results. Part(a) shows the *transformed* moving images (element maps of lead (Pb-L) with additional transparent padding to match the size of the visible fixed image (visible image of the painting “*The Battle of San Romano*”, NG583, The National Gallery, London [117]) in correspondence to their relative location with regards to the visible fixed image. Part(b) shows the individual transformed moving images with additional transparent padding. These can be used as layers to overlay on top of the visible fixed image (shown in (c)). Part(d) shows the resulting mosaic technical image overlaid on top of the visible fixed image. Part(e) shows the transformed moving images without the fixed visible image, thus obtaining a mosaic technical image with transparent background.

5.5 Discussion

Throughout the development of the area-based method presented in Chapter 3 and the experiments conducted in Chapter 4, we noted that image registration (especially of images produced by different modalities) entails a series of steps that are traditionally conducted in different interfaces. This can be cumbersome and time consuming. Thus, throughout the development of the two versions of the GUI, we focused on creating a unified interface to allow the user to perform as many tasks as possible in a streamlined manner. For this purpose, we note that including a selection of image processing tools in the GUI would allow the user to investigate the results obtained in a more thorough manner, without the need of moving the images to other interfaces to obtain further insights. Thus, it would be ideal to incorporate tools for analysis (e.g. histograms to help the visualisation of individual images) or manipulation of the images obtained (e.g. Gaussian filters, pixel averaging) to help obtaining smoother transitions between overlapping consecutive images of potentially different intensity values in mosaicked or composite images. This would enable the user to further inspect the images in a single interface as well as assisting with the assessment of the registration obtained.

5.6 Summary of this chapter

In this chapter we discuss two graphical user interfaces (GUI) developed to simplify the use of the automated area-based method that was previously introduced in Chapter 3. This chapter presents the requirements and constraints behind the implementation of our area-based method, which in turn motivated the client-server architecture that constitutes the basis of this browser-based, user-facing applications. It also details the workflow required to perform the registration of a set of images using the GUI: from selecting the images to align, to the creation of a *settings* file containing all the corresponding information (i.e. images to align and their assigned initial positions). Finally, this chapter presented an overview of the possible workflows associated to executing our area-based method for the registration of the images selected in the GUI, either by the user (*first* GUI) or by means of a server application (*second* GUI). In both cases, the *settings* file serves as the only input for this purpose. This chapter concluded by discussing potential additional tools for processing the images displayed in the GUI.

Chapter 6

Conclusion

In this thesis, we have presented a novel, automated registration and mosaicking method for use in heritage applications. Rather than implementing a feature-based approach, our area-based method works with the incremental maximisation of the average mutual information of the images to align. This method is optimised for analytical imaging spectroscopy applications (where the corresponding data may be processed and arranged into image stacks), supporting the registration of a stack of images that, treated as the moving image, can be aligned to a target fixed image by identifying a unique best transformation to apply to the moving image with consensus amongst all images in the stack.

To facilitate use of this registration method, we have developed a graphical user interface (GUI) to enable the user to easily set up the initial parameters of the images to be aligned, based on simple visual alignment using a browser-based tool that can be accessed from any computer.

This GUI was developed with the aim of streamlining the processes related to setting up the initial parameters of the images to align and then running their registration via the same interface. In the first version of the GUI (*initial alignment application*, see Fig. 5.3), the user can save the settings assigned to the images to align and conveniently compare different configurations prior to executing the registration process (as illustrated in workflow in Fig. 5.5). The full version of the GUI allows the user to perform these tasks, and additionally, execute the registration process and access and download the results (i.e. the transformed moving images) from the same browser-based platform (as illustrated in Fig. 5.6 and workflow in Fig. 5.2).

Originally developed for registration of images obtained from MA-XRF scanning to the visible image of the same artwork, this area-based method can be broadly implemented for registration of images obtained with various different modalities, such as infrared reflectography, ultraviolet-induced visible fluorescence imaging, raking light photography, and X-radiography - with these various image modalities either serving as the moving or fixed images.

Our image registration method provides satisfactory results, in particular for the registration of the technical images obtained with MA-XRF scanning of “*The Death of Actaeon*” (NG6420,

The National Gallery, London), as presented in Section 4.2. In this case, considered particularly problematic, the technical images were acquired in a non-consecutive manner over an extended length of time, resulting in different distortions even in adjacent images due to the changes in position of the optical device.

6.1 Summary of thesis achievements

Key aspects of our automated area-based method include:

- The use of mutual information as a metric for registration of multimodal images for use in the cultural heritage sector - with emphasis on the alignment of technical images obtained from MA-XRF scanning of painted artworks to the visible image of the same artwork.
- The implementation of sequential, progressively refined alignments of the moving image, finalising with a non-rigid transformation so as to provide a detailed registration of the moving image, leading in this way to a fast and reliable method.
- The identification of a unique best transformation to apply to a stack of images with consensus among all the images in the stack - that, in turn, constitutes the moving image. This is done by evaluating all candidate transformations across all images in the stack so as to keep the one that returns the highest average value of mutual information as the best transformation for that image stack.
- Mosaicking of technical images by sequentially registering multiple neighbouring moving images. This is achieved by implementing the initial registration of the scanned areas that are in the centre of the final mosaicked image, and then making use of the results to inform the registration of the remaining neighbouring scanned areas.
- The introduction of a graphical user interface (GUI) to help simplifying the use of our area-based registration method, by making it possible for the user to access the application from the browser and set up the initial parameters of the images to align in a visual manner prior to commencing the registration method.
- The automated nature of the registration method, permitting the user to run the registration immediately after setting the initial locations of the moving images relative to the target fixed image. The latest version of the GUI allows the user to perform all these tasks and, additionally, execute the registration process and access and download the results (i.e. the *transformed* moving images) from the same browser-based platform.

6.2 Future work

The area-based method presented in this thesis may be further expanded to address the registration of other image types, in particular of images obtained with *hyperspectral imaging*, also known as *reflectance imaging spectroscopy* (RIS) [18, 19, 17]. By scanning an artwork with this modality, the standard approach makes it possible to obtain reflectance images by sampling points along each of the two corresponding spectra: the visible and near-infrared (VNIR) spectrum of typically 400-1000nm, and the short-wave infrared (SWIR) spectrum of typically 1000-2500nm. Scanning with this modality results in obtaining a datacube¹ of two spatial dimensions (which set the size of the reflectance images) and one spectral dimension. It is considered that these reflectance images are aligned by construction². The amount of images collected at each of these spectra depends on the sampling to which the instrument may be set, typically collecting one narrow-band image for every few nanometers of the wavelength range³. This modality may thus yield image stacks that are constituted by a high number of images, and applying our registration method with consensus among all images in any such stack would take prohibitively long to complete, more so as we envision our method as an agile implementation to identify a unique best transformation to apply to the entire datacube.

Thus, in order to speed up the registration of image stacks obtained with this modality, one possible approach to explore in future work is to reduce the amount of images in the stack, by taking images at, for example, 50 nm intervals throughout the spectrum (instead of using all of the images that could be created based on the full spectral resolution of the instrument). Another possibility would be to combine the default images that are obtained at the spectral resolution of the instrument into a reduced number of synthetically produced images. For example, the full set of images that would be created -based on the full spectral resolution of the instrument- could be organised into consecutive groups of 20 neighbouring images each. Then, these image could be combined together into one single image per group (i.e. binning the images after the capture stage). These operations would make it possible to reduce the amount of images in the original stack (effectively binning the spectrum at a "coarse" rate and thus reducing its resolution).

However, due to the characteristics of the images obtained with this modality, it would be necessary to evaluate the performance of these operations and the potential necessity of applying further distortions so as to ensure that the registration is achieved correctly. In particular, for the

¹ Also referred to as *image-cube* [17].

² Extensive work on filter-based multi-spectral imaging was conducted by David Saunders, John Cupitt and Haida Liang during their time at the National Gallery, since the original images produced were not absolutely aligned by construction [20, 21, 22]. Note that this work led to the development of the software package NIP2 [16]. However, reflectance images obtained along a wide spectrum may present optical effects and other complications due to capture settings of the instrument, such as *barrel* and *keystone* distortions associated with "push-broom" scanning mode [118]. As a consequence of this, the reflectance images might not be perfectly aligned throughout the full extent of the spectrum. We note that this situation needs further study to ensure alignment across the reflectance images in the stack prior to conducting image registration of datacubes of this type.

³ In the case of the setup implemented at the National Gallery, the equipment has high spectral resolution and provides an average spectral sampling of circa 3 nm. This results in obtaining 516 narrow-band images over the SWIR spectra of range 1000-2500 nm [19].

registration of images proceeding from the VNIR range, it might be necessary to apply geometrical corrections due to lens distortion (i.e. barrel correction [118]) before applying the unique best transformation that is identified for registering the entire image stack. In the case of the SWIR range, the images obtained from this spectrum may be subject to more severe distortions, since distortions due to barrel effect may increase with longer wavelengths. Thus, distortions of this type may need to be corrected first. Additionally, images of this spectrum may display more details relating to sub-surface features due to longer wavelengths. This aspect might also affect the registration results, and strategies such as identifying transformations related to the images generated by shorter wavelengths may need to be implemented to ensure resemblance to features displayed in the visible target image.

Note that images obtained with HSI are usually closer in spatial resolution to the photographic visible image of the same artwork, becoming computationally expensive to handle. This aspect sits in direct contrast when considering that, for the registration of images obtained with MA-XRF scanning, it is customary to downscale the resolution of the visible target image to match the resolution of the technical moving image (see Section 2.1).

Another technique worth exploring in future work is enhancing the HSI datacube by means of adding synthetic images, as explored in this thesis for the registration of images obtained with MA-XRF scanning in Section 4.1.3. When performing registration of images obtained with HSI, it is common to employ additional images generated with principal component analysis [119] (PCA), minimum noise fraction [120] (MNF) or false-colour images [121] (by assigning different images to the R, G and B channels of a newly combined image). In the case of our image registration method, we envision that these could be added to the original HSI image stack so as to enhance the moving image, and thus help to identify a unique best transformation to apply to the full stack of HSI images for their registration to the visible image of the same artwork.

It is important to note that at the time of submitting this thesis, the area-based registration algorithm described in this thesis provides satisfactory registration results for the sets of element map images obtained with MA-XRF scanning and the individual images obtained by illuminating an artwork with ultraviolet light, visible light at a raking angle and infrared reflectograms. However, we have noticed that the method is less effective for images obtained with X-radiography. In particular, our method fails to provide satisfactory registration results where the depth of the artwork makes it difficult to reconcile the details captured at the front and at the back of the object, due to parallax effects [122]. Examples of this situation are the tests conducted with technical images of paintings on thick wooden panels, such as when conducting registration of X-ray technical images of NG1093 [60] and NG1155 ⁴ [123]. It is important to note that we have not yet conducted registration experiments with HSI images. For these reasons, we envision future

⁴ In the case of NG1093 (*The Virgin of the Rocks* by Leonardo Da Vinci), the painted wooden panels has a wooden 'cradle' applied. In the case of NG1155 (*The Assumption of The Virgin* by Matteo di Giovanni), X-ray opaque paint is roughly applied to the back of the panel, making these cases particularly challenging.

efforts may be directed to expand the work presented in this thesis to address the registration of the technical images of artworks obtained with these modalities.

With regards to further improving the current performance of the registration algorithm supporting our area-based method, we envision future work could explore the following aspects:

- Computing the mutual information of more than two images at a time. Throughout the method presented in Chapter 3, the mutual information is computed from two images at a time. In particular, it is computed from the target fixed image and one individual image at a time, the latter belonging to the image stack that is treated as the moving image.
- Automating the selection of images (elemental maps) that constitute the image stack that is treated as the moving image for registration to the visible image of the artwork. Throughout the method presented in Chapter 3, these images are selected by the user, but they could be chosen following a series of indicators such as the entropy of the individual images or other measures of the images to align.
- Reducing noise of images to align by applying segmentation or other image processing techniques. Throughout the method presented in Chapter 3, the mutual information is computed from sampling pixel values from the entire image. Reducing the sampling area to a region of interest within the image to evaluate can lead, potentially, to discard areas where information may be "trivial" and concentrate efforts to sample from the most significant areas of each image at a time.

With regards to the GUI presented in Chapter 5, it might be of interest to users to incorporate publicly accessible image-viewers into this application, such as comparative viewers [124, 125] that make it possible to zoom and track the moving and fixed images together, in the style of overlay viewers and curtain viewer applications. Further additions to this application could include image processing tools for analysis (e.g. presenting histograms to help with the visualisation of individual images) or manipulation (e.g. Gaussian filters, pixel averaging) to obtain visually smoother transitions between overlapping consecutive images of potentially different intensity values. Not only these additions would help improving the quality of the final mosaicked image, they would also enable the user to further inspect the images in a single interface as well as it would further facilitate the assessment of the registration obtained.

Appendix A

User guide: image registration using the first version of the browser-based image registration GUI and running the registration algorithm from a local installation

The purpose of this appendix is to help the user to navigate the initial implementation of the browser-based graphical user interface (*first* GUI) presented in Chapter 5. To this end, we present a user guide with the workflow for conducting image registration with our area-based method (as presented in Chapter 3) using our dedicated GUI application and python locally (i.e. installed in the same computer from where the user is accessing the browser-based application, as illustrated in Fig. 5.5).

Our GUI application provides the user with a workspace ("canvas") where, once selected, the images to align are displayed. These images must be saved in the local computer. The user can open a first image (usually the visible image of the artwork, which will serve as the target fixed image). Following this, the user can open either a single image to register or a set of multiple images (usually the technical images of the same artwork), which will be automatically arranged into an image stack, and will serve as the first moving image. Then, the user can open a second image or set of multiple images (image stack), which will be automatically arranged into a separate new image stack, and will serve as the second moving image. This process may be repeated for all moving image stacks that need to be registered to the target fixed image. There is no limit to the amount of moving images that can be opened in the GUI, nor there is a limit to the number

of images that constitute each image stack (other than the browser memory-limit, as discussed in Section 5.2.1).

Once the images are opened in the GUI, the user can designate the initial parameters to apply to them - such as translation along x and y axis, rotation and scaling. These parameters constitute the initial locations of the images to align, which the user can designate by inspecting these images in a visual manner with our GUI. For this purpose, the user can explore the initial overlapping of the images by altering the transparency of each image in the image stack.¹ As discussed in Section 5.1, for each moving image as well as for the fixed image, these parameters will be the starting point for the first search of our area-based method (see Section 3.4).

Once the user is satisfied with the initial overlapping of the images, the parameters defined for all the images opened in the GUI can be saved into a *settings* file and used as input to start the registration process.

A.1 Assigning initial parameters for image registration

This first part of this user guide looks into opening the images to align, assigning their initial parameters and creating a *settings.json* file² with this information (in particular, this file will contain the images to align and the parameters designated by the user). How to use this file as the input to the area-based method will be covered in the second part of this user guide (see A.2).

Fig. A.1: Open the GUI application from a browser (e.g. Chrome).

The browser will show the landing page of the application, where the *canvas* and the zoom speed are displayed. Once there, the user can select to modify the size of the canvas (red window in the GUI) where all images will be displayed (here set to 1000 x 600 pixels; this dimension can be set according to the user's preference). By clicking on "*Upload Working images*", the user can select the target fixed image (usually the visible image of the artwork). This can be a 16-bit or 8-bit image saved as tiff, png or jpg.

¹ Note that modifying the transparency of the images is only for visualisation purposes, and does not affect the images exported for registration with our area-based method

² Note that in this *first* GUI, this file needs to be named in this way (*settings.json* as the python code will look for this particular file to collect the corresponding image data).

image registration

CANVAS v.230914

outer canvas (red) width: 1000 x height: 600 (pixels)
zoomSpeed slowZoom fastZoom

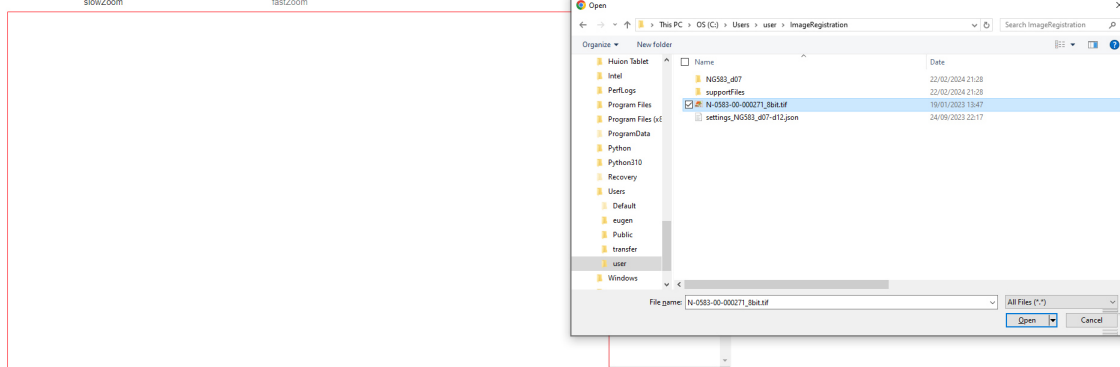


Figure A.1: Graphical user interface (GUI): landing page of the application. Here, the user can select the first image to open in the canvas, which will serve as the target fixed image.

Fig. A.2: By opening this first image, the user can start setting the parameters that will inform the registration. The fixed image can be scaled to match the resolution of the technical images that will be registered (here, scaling is set to 0.493, as the resolution of the technical images is lower than that of the visible image). Note that in the current image-registration python file (v.7, 20-12-2023), it is assumed that the fixed image remains at ($x = 0$, $y = 0$) of the canvas, and to a rotation angle of 0 degrees.

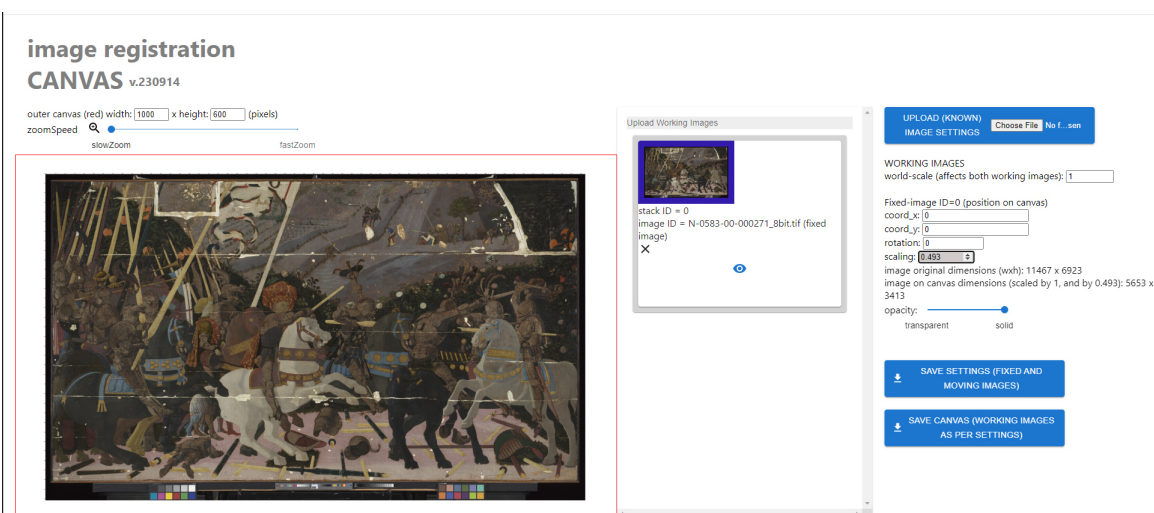


Figure A.2: Graphical user interface (GUI): Once the first image is opened, the user can manually designate the parameters that apply to this image.

Fig. A.3: Once the fixed image is opened, the user can click again on “Upload Working images” to select the stack of images that constitute the moving image. These can be 16-bit or 8-bit images saved as tiff, png or jpg format. Note that it is assumed that all images in the same stack are of the same dimensions and are aligned by construction. The user can assign values of translation, scaling, and rotation (in degrees) to the moving image. Note that these parameters are the same for all images in each image stack.

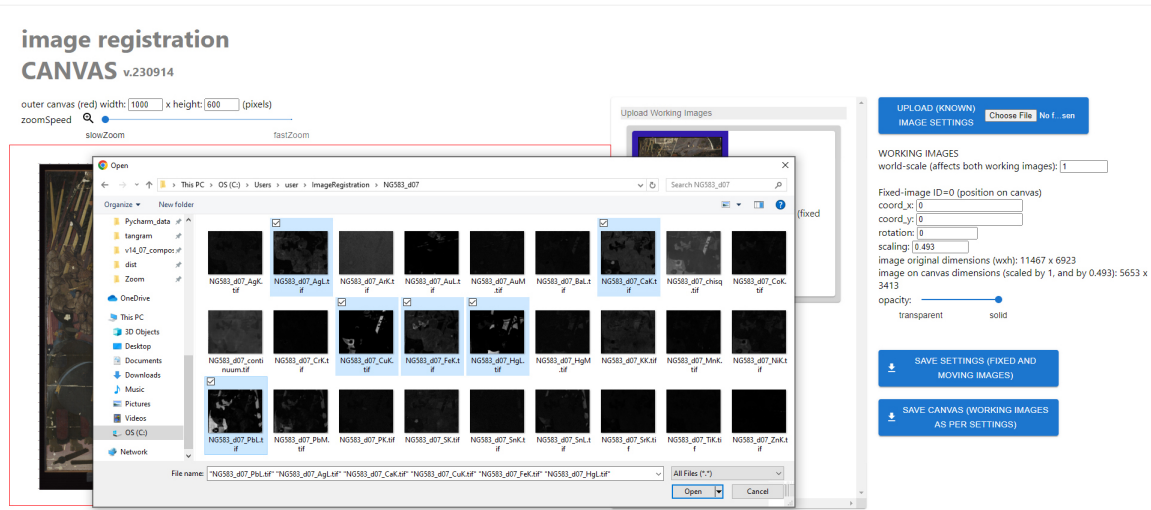


Figure A.3: Graphical user interface (GUI): Once the first image is opened in the GUI, the user can select the moving image. This can be composed by one or multiple images that, selected simultaneously, are automatically arranged into an image stack that serves as the moving image.

Fig. A.4: By clicking on the tiles representative of the moving image, the user can set the parameters. When selected, the images that compose the moving image are edged in orange.

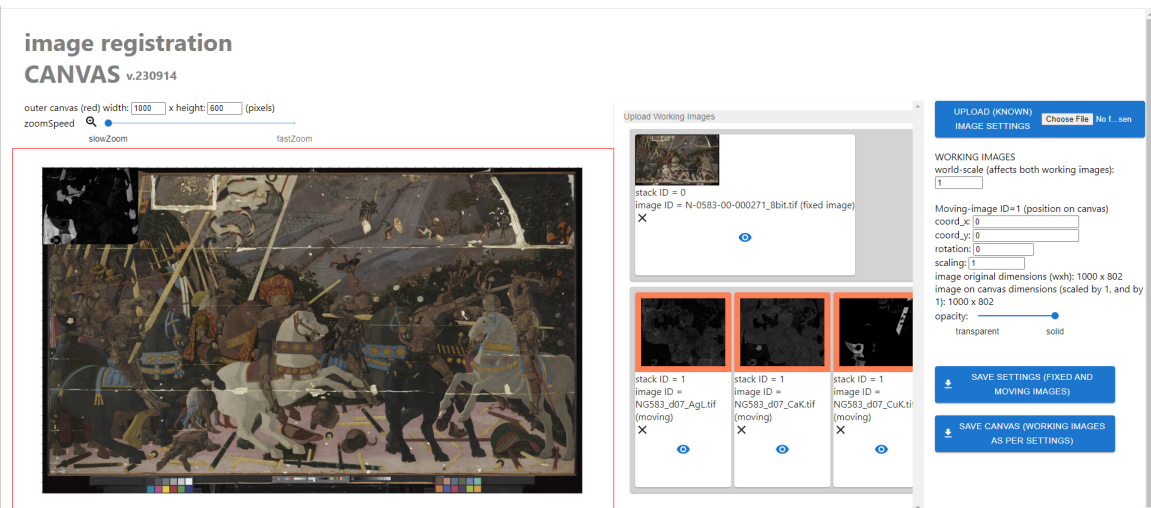


Figure A.4: Graphical user interface (GUI): Once the moving image is opened, the user can select it and designate the parameters that apply to it. The parameters designated for the moving image apply to all images in the stack.

Fig. A.5: Once these parameters are set, the user can save the *settings.json* file by clicking on the blue button in the bottom right corner “save settings (fixed and moving images)”. This file can then be used as input to run the area-based image registration method by running the python file *imreg_python__read-json-settings.py* in Anaconda Prompt terminal (see Section A.2).

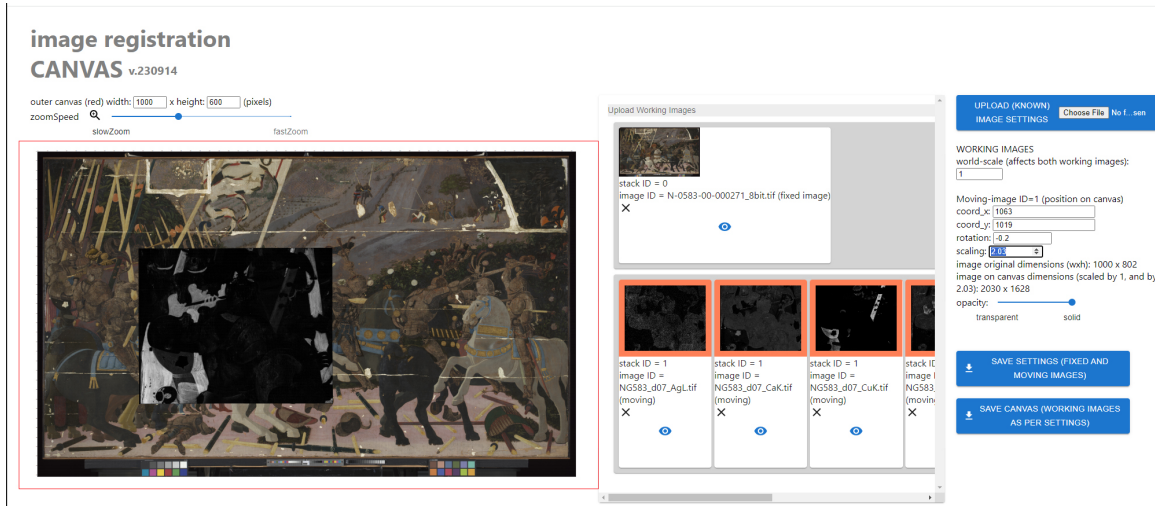


Figure A.5: Graphical user interface (GUI): The user modifies the parameters of the images that compose the moving image, here shown edged in orange. These are saved to the *settings.json* file.

Fig. A.6: The user can also upload a “settings.json” file that has been saved previously, and modify the parameters assigned to it so as to save a new version of this file - this can be done by clicking on the blue button ³ in the top right corner: “upload (known) image settings”.

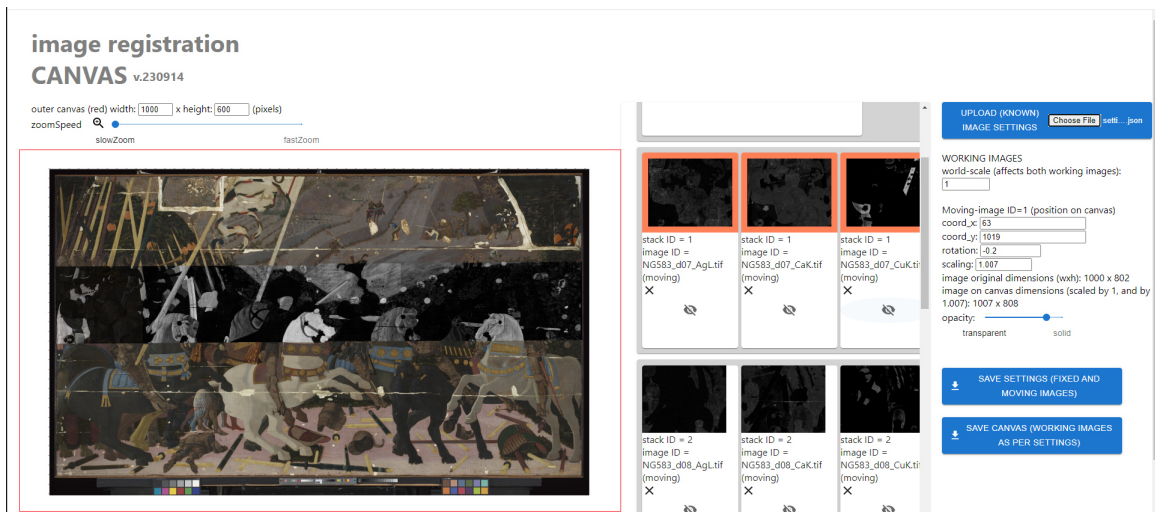


Figure A.6: Graphical user interface (GUI): The user can upload and modify a previously saved “settings.json” file, so as to continue or modify a previous configuration of initially aligned images.

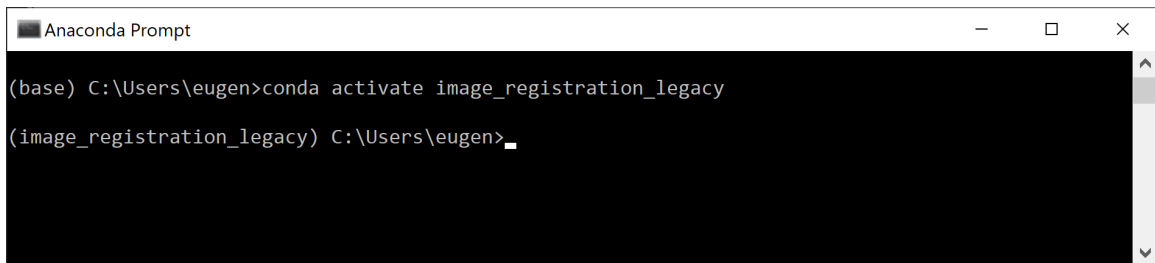
³ Note that in the current version of the GUI (v.230914), the blue buttons will display only after uploading a first image - this can be any image, as it will be removed by restoring the configuration of the “settings.json” file that will be uploaded next.

A.2 Implementing image registration with initial parameters (i.e. stored in *settings* file)

The second part of this user guide looks into implementing our area-based method, using as input the *settings.json* file with the relevant information. In particular, this file will contain the images to align and the parameters designated by the user (see A.1). For this purpose, this part of the user guide explains how to run the python files containing the algorithm supporting our area-based method, how to navigate the different folders where results will be saved, and how to access the resulting images obtained from the registration once complete.

Fig. A.7: Firstly, the user needs to set up the virtual environment with the corresponding libraries that support the functionalities behind the python code of our area-based method. For this purpose, from the toolbar of Windows OS, the user can open Anaconda Prompt. Once Anaconda opens, it will be set at the “base” virtual environment. The user needs to activate the “*image_registration_legacy*” environment, by typing:

“`conda activate image\_registration\_legacy`” into the command line. The user can check that this environment is activated as it shows in the command line in parentheses to the left, “`(image\_registration\_legacy)`”. This environment has all the libraries that enable the python code for registration to be ran.

A screenshot of the Anaconda Prompt window. The title bar reads "Anaconda Prompt". The command prompt shows the user typing "conda activate image_registration_legacy" at the "(base) C:\Users\eugen>" prompt. The next line shows the prompt has changed to "(image_registration_legacy) C:\Users\eugen>".

```

Anaconda Prompt
(base) C:\Users\eugen>conda activate image_registration_legacy
(image_registration_legacy) C:\Users\eugen>
```

Figure A.7: Anaconda Prompt: setting up the virtual environment with the corresponding libraries that support the functionalities behind the python code of our area-based method.

Fig. A.8: The user needs to create a folder where to save the image registration results. The structure of this folder is suggested as follows: in location “C:\Users\user\ImageRegistration”, create a folder named following artwork number and datacubes to register (i.e. NG583_d07). This is the project folder, which shall contain the python files to be run, plus a subfolder named “results_all” (this last one is the destination folder, where all results will be saved).

In the command line, the user needs to indicate the location of the folder where to run the registration (destination folder, i.e. “results_all”). Note that this can be done by typing “cd” in the command line, plus dragging the destination folder into the Anaconda Prompt. The address will be then copied into the command line, and the user can press “Enter”.

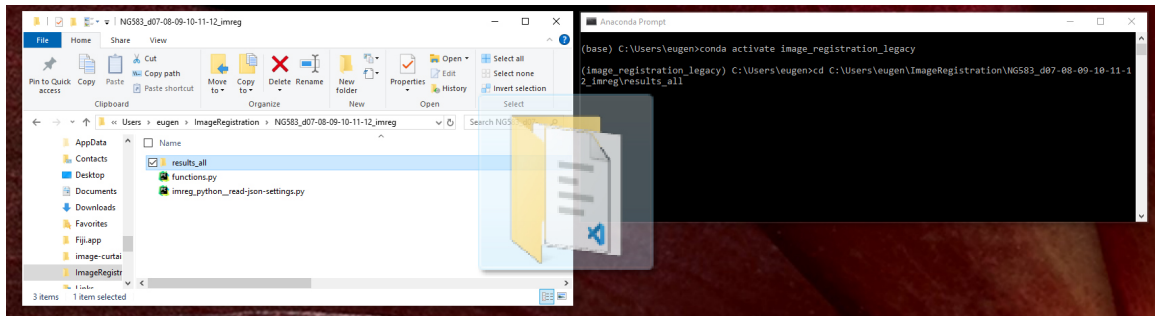


Figure A.8: Anaconda Prompt: indicating the location of the folder where to run the registration. In this image, the user has typed “cd” in the command line, plus dragged destination folder into the Anaconda Prompt. This action copies the location of the selected (and dragged) folder, which will be used for storing results of our area-based method.

Fig. A.9: The destination folder (i.e. “results_all”) will be referenced next to the virtual environment.

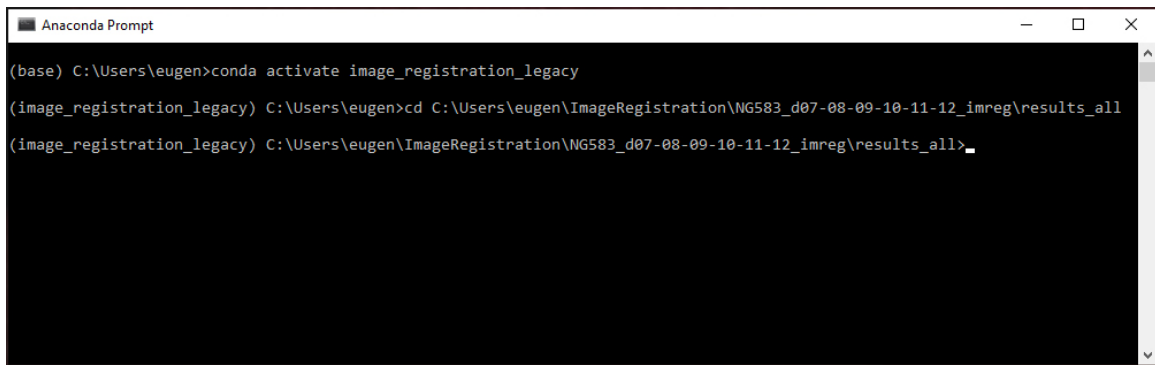


Figure A.9: Anaconda Prompt: indicating the location of the folder which will be used for storing results of our area-based method.

Fig. A.10: As mentioned previously in this user guide, the destination folder (i.e. “results_all”) is inside the project folder (i.e. “C:\Users\user>cd C:\Users\user\ImageRegistration\NG583_d07-08-09-10-11-12_”). The project folder holds the python files, namely, the file “imreg_python_read-json-settings.py” (the one to be ran in Anaconda) and the file “functions.py” (this one runs on the back, supporting the first one by containing all functions called throughout the automated area-based method).

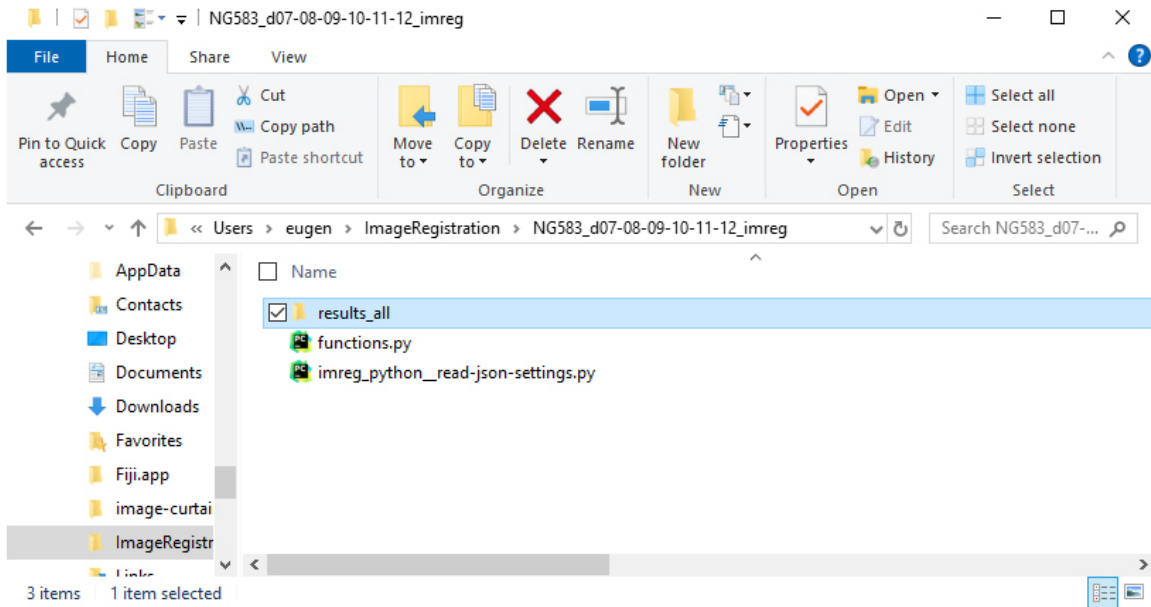


Figure A.10: Windows OS: folder which will be used for storing results of our area-based method.

Fig. A.11: The destination folder (i.e. “results_all”) has to host the *settings.json* file. This is the file created by the GUI, with all images and initial locations. It has to be named like this (“*settings.json*”), as the python code will look for this name file to get all the registration data.

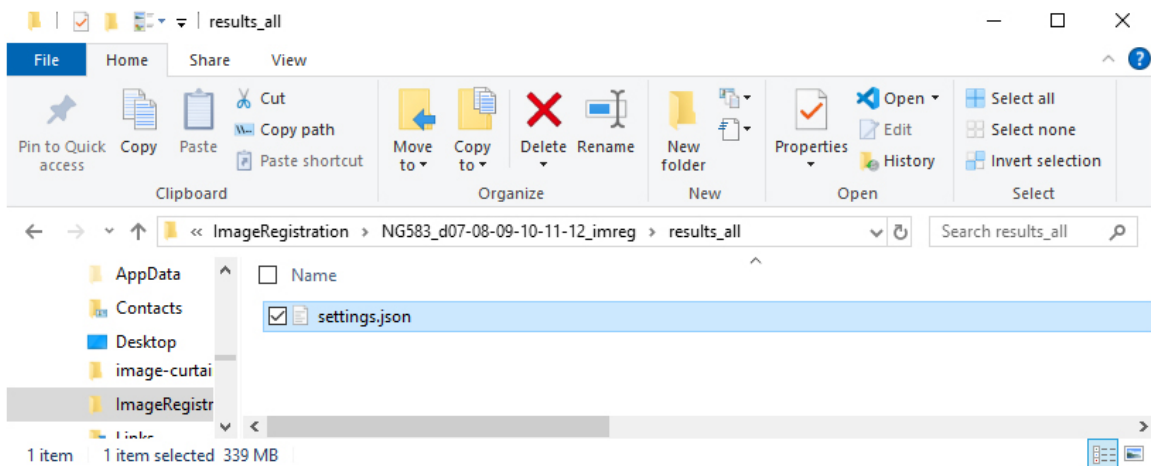


Figure A.11: Windows OS: folder for storing results contains the “*settings.json*” file.

Fig. A.12 and Fig. A.13: Once Anaconda Prompt is set to the correct folder (i.e. “C:\Users\scienceuser\ImageRegistration\NG583_d07-08-09-10-11-12_\results_all”), the user can instruct to run the python code, by typing “python” into the command line, plus typing the address of the file to run (this can be done by dragging the python file into the command line):

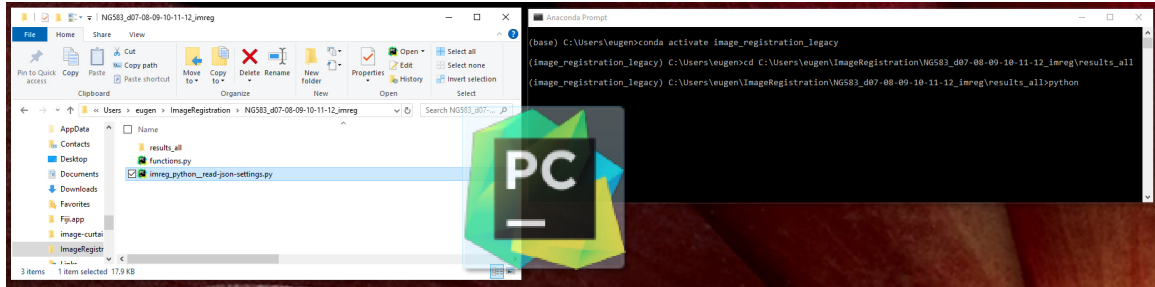


Figure A.12: Anaconda Prompt: to run the area-based method, the user needs to indicate the location of the corresponding python file. This can be typed into the command line, or the file can also be dragged into it.

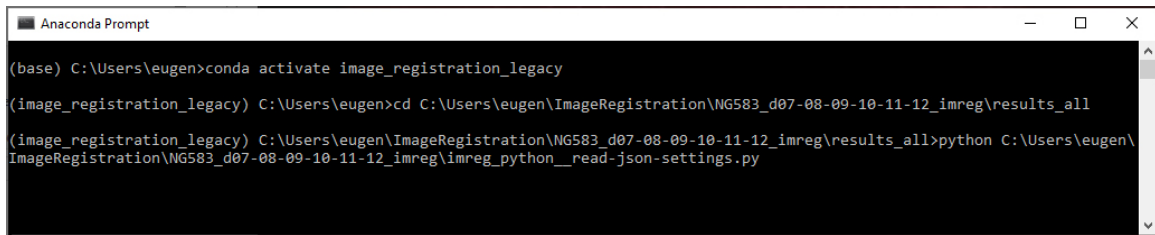


Figure A.13: Anaconda Prompt: indicating the location of the python file that implements our area-based image registration method.

Fig. A.14: Once this is done (i.e. the name of the python file to be run appears in the command line, followed by the “python” command), the user can press “enter” and the registration will commence.

```
(base) C:\Users\eugen>conda activate image_registration_legacy
(image_registration_legacy) C:\Users\eugen>cd C:\Users\eugen\ImageRegistration\NG583_d07-08-09-10-11-12_imreg\results_all
(image_registration_legacy) C:\Users\eugen\ImageRegistration\NG583_d07-08-09-10-11-12_imreg\results_all>python C:\Users\eugen\
ImageRegistration\NG583_d07-08-09-10-11-12_imreg\imreg_python_read-json-settings.py
configuration data:
['C:\\Users\\eugen\\ImageRegistration\\NG583_d07-08-09-10-11-12_imreg\\imreg_python_read-json-settings.py']
shape moving_image[0] = (802, 1000)
shape moving_image[1] = (802, 1000)
shape moving_image[2] = (802, 1000)
shape moving_image[3] = (802, 1000)
shape moving_image[4] = (802, 1000)
shape moving_image[5] = (802, 1000)
rot to check
0.493
ini_pos_d = (-63, -1019)
ideal_final_tr = {'a11': 1.0, 'a12': 0.0, 'a21': 0.0, 'a22': 1.0, 'tx': -63, 'ty': -1019, 'mi_average': -999.0}
shape fixed_image= (6923, 11467)
shape fixed_image COLOR = (6923, 11467, 3)
fixed img from json = (6923, 11467)
```

Figure A.14: Anaconda Prompt: commencement of registration using our area-based method.

Fig. A.15: All intermediate results produced throughout the application of our area-based method -while this is taking place- will be saved in the destination folder (i.e. “results_all”) in dedicated subfolders, one for each datacube that is present in the “settings.json” file.

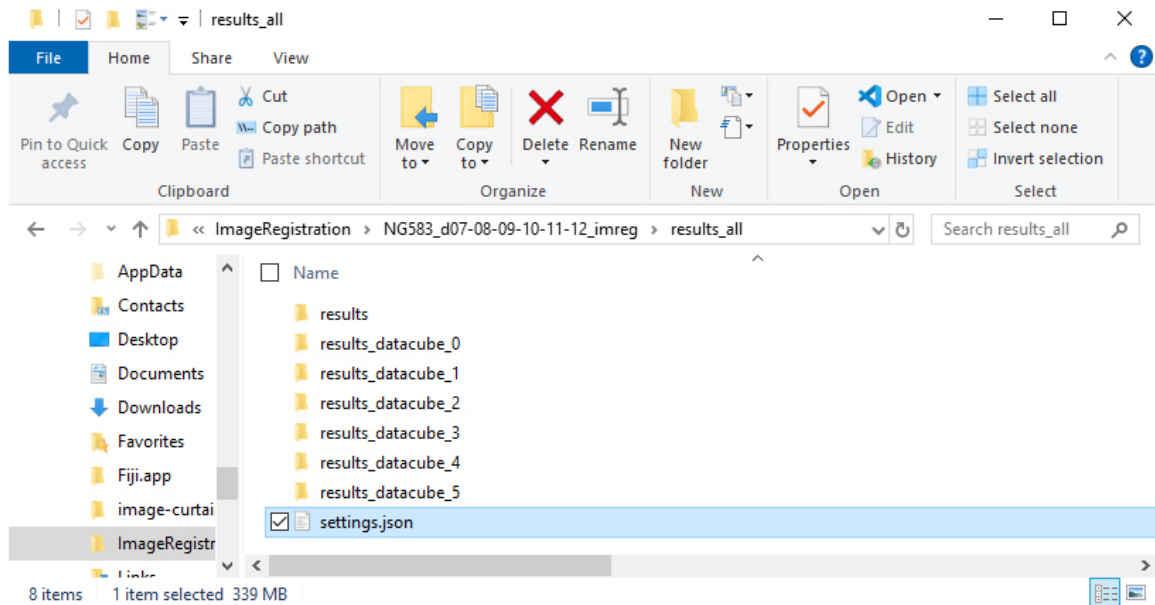


Figure A.15: Windows OS: subfolders for storing intermediate results that are produced while the registration method takes place.

Fig. A.16: The final results are the transformation that is applied to the datacube for its registration to the target fixed image, and the "transformed" images - which are each image of the image stack with the above-mentioned transformation applied. In particular, the images are the PNG layers for creating the mosaicked final image, the final transformations applied to each datacube, and the fixed image, as 8-bit image, grey and color versions, scaled following the settings assigned at the GUI. These will be saved inside the folder "results_all/results".

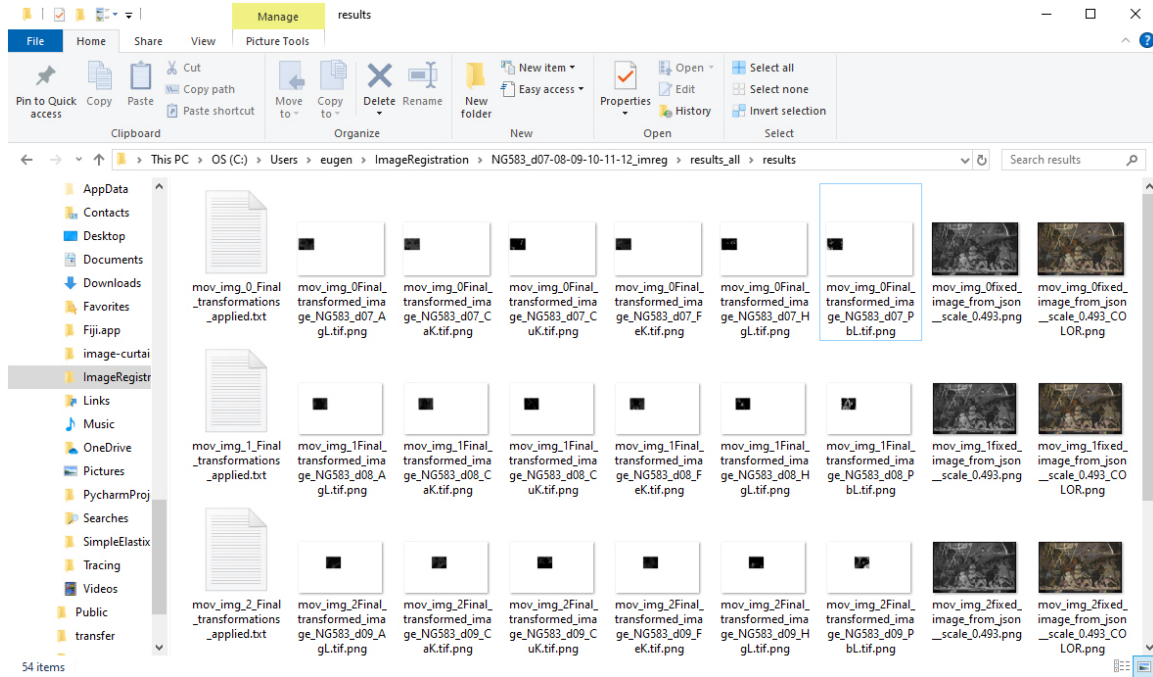


Figure A.16: Windows OS: folder with final results. Here are shown: the images (PNG format), which work as layers for creating the mosaicked final image, the final transformations applied to each datacube, and the fixed image, as 8-bit image, grey and color versions.

Once the registration has run and the results are obtained, the user can proceed to combine the resulting images (PNG format) in order to produce the mosaicked final image. In the manner of layers, these images can be opened in the same GUI, or in an image processing software such as Photoshop. The user can then decide the order of the layering of these images, since neighbouring images may overlap and their layering will entail one image may be placed below another one, or viceversa.

Note that PNG images and the visible image have all the same size, and opening them together, the images will be shown at their corresponding position relative to the visible image.

Fig. A.17: Opening PNG layers in Photoshop. The user needs to navigate to **File > Scripts > Load Files into stack**

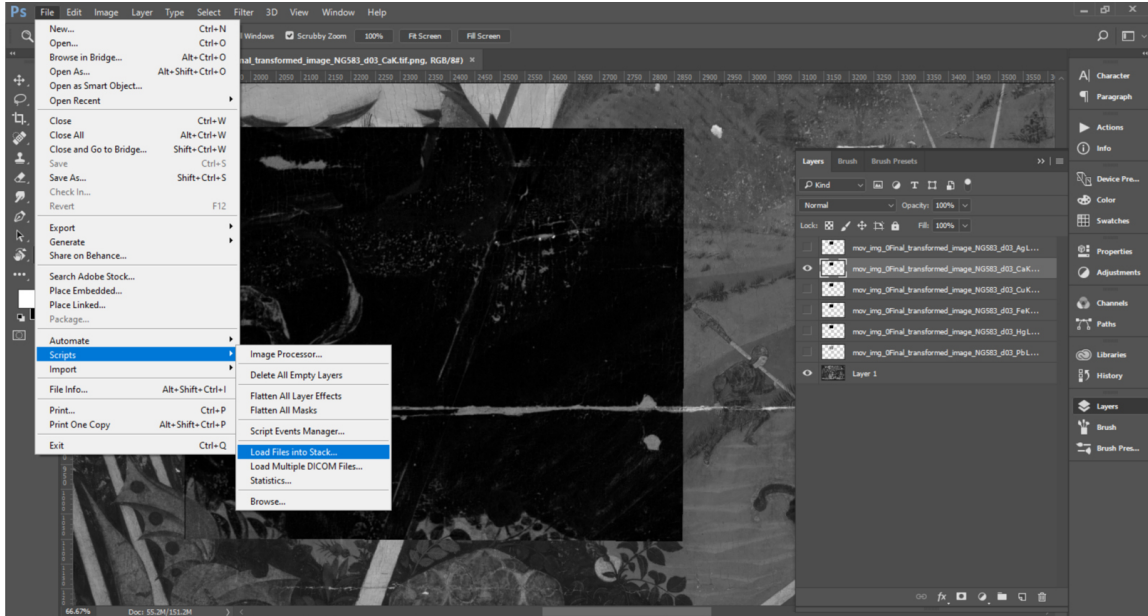


Figure A.17: Photoshop: opening the PNG images resulting from our area-based method.

Fig. A.18: By clicking on **Load Files into stack**, the **Load Layers** window will pop up. Here, the user may select the images to open as layers for visual inspection (i.e. PNG images together with the fixed visible image). Images will appear in the dialogue window. Press OK to load in Photoshop. Fig. A.19 shows the position of these layers relative to the target fixed image.

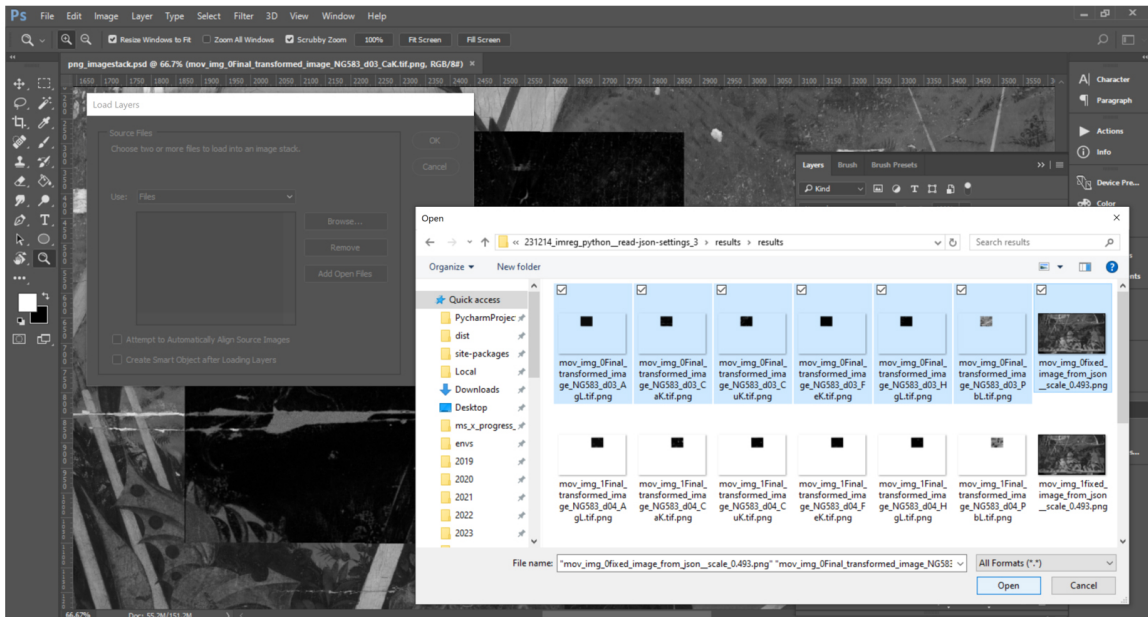


Figure A.18: Photoshop: selecting the images to open as layers for the final mosaicked image.

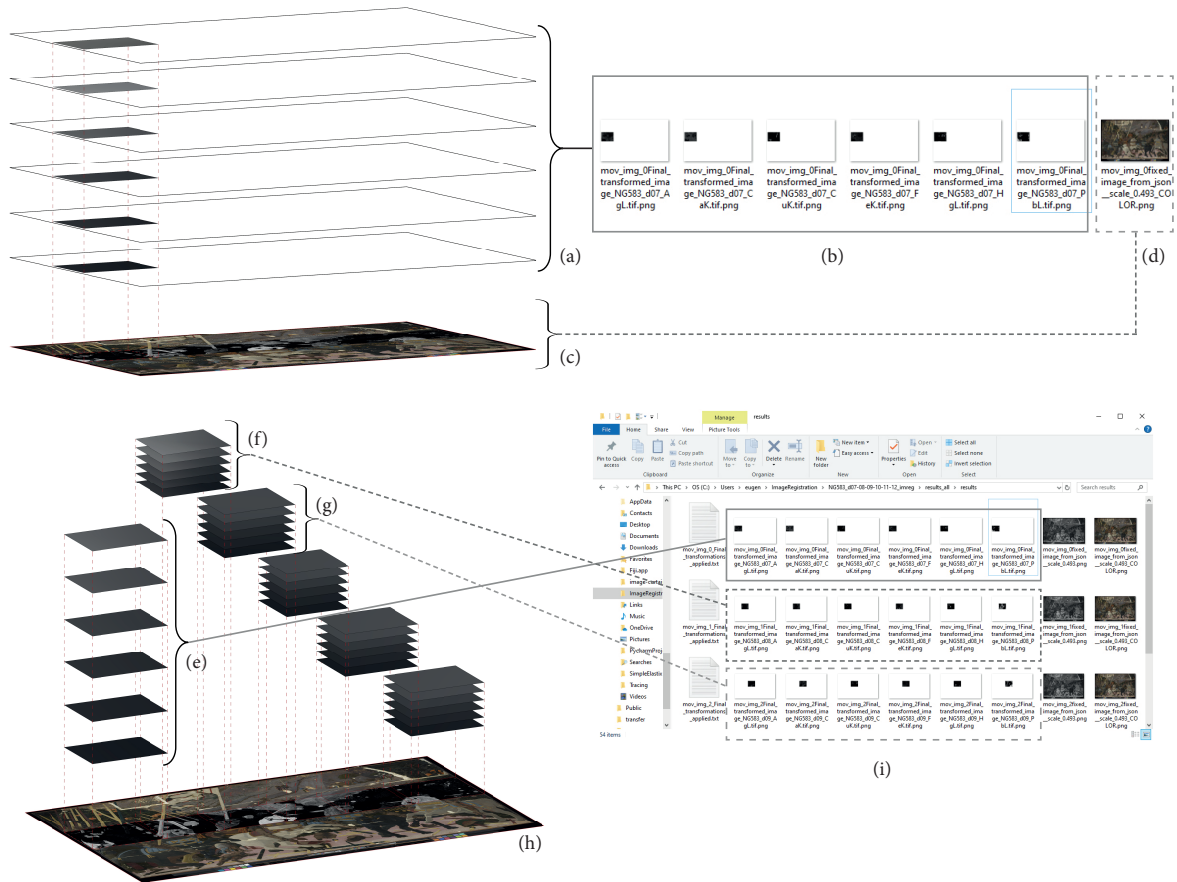


Figure A.19: Final mosaicked image: this sequence shows the PNG images (Fig. a) that constitute the layers of the first datacube, as obtained in the corresponding "results" folder (Fig. b), once registered to the target fixed image (Figs. c, d). It is noted that the PNG image layers are of the same size of the fixed image so that they can be conveniently overlayed. Next, the same first datacube (Fig. e) is shown in relation to the neighbouring datacubes (Figs. f, g) and fixed image (Fig. h), as obtained in the corresponding "results" folder (Fig. i).

Bibliography

- [1] A. Ardeshir Goshtasby. *2-D and 3-D Image Registration: For Medical, Remote Sensing, and Industrial Applications*. John Wiley & Sons, 2004. ISBN:9780471649540.
- [2] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21:977–1000, 2003.
- [3] A. Ardeshir Goshtasby. *Image Registration Principles, Tools and Methods*. Advances in Computer Vision and Pattern Recognition. Springer London, London, 1st ed. 2012. edition, 2012.
- [4] Nikos. Paragios, Yunmei. Chen, and Olivier D. Faugeras. *Handbook of Mathematical Models in Computer Vision*. Springer US, New York, NY, 1st edition, 2006.
- [5] Wilhelm Burger and Mark J. Burge. *Digital Image Processing, An Algorithmic Introduction Using Java, Texts in Computer Science*. Springer London, 2016.
- [6] MATLAB MathWorks. MATLAB: Approaches to registering images. <https://nl.mathworks.com/help/images/approaches-to-registering-images.html>, 2023. [Online; accessed 28-November-2023].
- [7] Marika Spring, Rachel Billinge, Letizia Treves, Nelly Von Aderkas, Catherine Higgitt, Annelies Van Loon, and Joris Dik. Goya’s Portraits in the National Gallery: their technique, materials and development. *National Gallery Technical Bulletin*, 37:78–104, 2016.
- [8] Barak Sober, Spike Bucklow, Nathan Daly, Ingrid Daubechies, Pier Luigi Dragotti, Catherine Higgitt, Jun-Jie Huang, Aleksandra Pižurica, Wei Pu, Suzanne Reynolds, Miguel Rodrigues, Carola-Bibiane Schönlieb, and Su Yan. Revealing and reconstructing hidden or lost features in art investigation. *IEEE BITS the Information Theory Magazine*, 2(1):4–19, 2022.
- [9] Richard A. Robb. Visualization. In Isaac N. Bankman, editor, *Handbook of Medical Image Processing and Analysis (Second Edition)*, pages 725–727. Academic Press, Burlington, second edition edition, 2009.
- [10] Lisa Gottesfeld Brown. A survey of image registration techniques. *ACM Comput. Surv.*, 24(4):325–376, dec 1992.

- [11] D. Conover, J.K. Delaney, and M.H. Loew. Automatic registration and mosaicking of technical images of Old Master paintings. *Applied Physics A*, 119:1567–1575, 2015.
- [12] Prasanna Sridhar, Abhishek Dutta, and Andrew Zisserman. Image Compare. <https://www.robots.ox.ac.uk/~vgg/software/image-compare/>, 2022. [Online; accessed 30-May-2023].
- [13] Aline Sindel, Andreas K. Maier, and Vincent Christlein. A multi-modal registration and visualization software tool for artworks using craquelurenet. *ArXiv*, abs/2208.08836, 2022.
- [14] Adobe Inc. Adobe photoshop CC 2019. <https://www.adobe.com/products/photoshop.html>, 2019.
- [15] The GIMP Development Team. Gimp v.02.10-2019. <https://www.gimp.org>, 2019.
- [16] John Cupitt, Joe Padfield, Hans Breuer, Rich Lott, and Leo Davidson. nip2 — a user interface for libvips. <https://github.com/libvips/nip2>, 2007. [Online; accessed 16-Jun-2023].
- [17] M. Picollo and J.K.Delaney. Jaic special issue on “reflectance hyperspectral imaging to support documentation and conservation of 2d artworks”. *JOURNAL OF THE AMERICAN INSTITUTE FOR CONSERVATION*, 58:1–2, 4 2019.
- [18] Marta Melchiorre Di Crescenzo, Catherine Higgitt, Rachel Billinge, and Marika Spring. *Integration of macro X-ray fluorescence scanning and reflectance hyperspectral imaging into research and conservation activities at the National Gallery, London, in 'Tales of the Unexpected' in Paintings Conservation*, pages 22–35. 10 2020.
- [19] Catherine Higgitt Marika Spring, Marta Melchiorre Di Crescenzo and Rachel Billinge. *Leonardo da Vinci's London Virgin of the Rocks re-examined, Techne: Hors-Série Léonard de Vinci, l'expérience de l'art*, pages 100–115. 2024.
- [20] David Saunders and John Cupitt. Image processing at the National Gallery: The VASARI project. *National Gallery Technical Bulletin*, 14:72–85, 1993.
- [21] David Saunders and John Cupitt. Elucidating reflectograms by superimposing infra-red and colour images. *National Gallery Technical Bulletin*, 16:61–65, 1995.
- [22] Haida Liang, David Saunders, and J. Cupitt. A new multispectral imaging system for examining paintings. *Journal of Imaging Science and Technology*, 49:551–562, 12 2005.
- [23] D. Mattes, D. R. Haynor, H. Vesselle, T. K. Lewellen, and W. Eubank. PET-CT image registration in the chest using free-form deformations. *IEEE Transactions on Medical Imaging*, 22(1):120–128, 2003.
- [24] J.P. Pluim, J.B. Maintz, and M.A. Viergever. Mutual-information-based registration of medical images: A survey. *IEEE Transactions on Medical Imaging*, 22(8):986–1004, 2003.

- [25] S. Klein, M. Staring, K. Murphy, M.A. Viergever, and J.P.W. Pluim. elastix: a toolbox for intensity based medical image registration. *IEEE Transactions on Medical Imaging*, 29, no1:196–205, 2010.
- [26] S. Yan, J. Huang, N. Daly, C. Higgitt, and P. L. Dragotti. When de Prony met Leonardo: An automatic algorithm for chemical element extraction from macro x-ray fluorescence data. *IEEE Transactions on Computational Imaging*, 7:908–924, 2021.
- [27] Su Yan, Jun-Jie Huang, Herman Verinaz-Jadan, Nathan Daly, Catherine Higgitt, and Pier Luigi Dragotti. A fast automatic method for deconvoluting macro x-ray fluorescence data collected from easel paintings. *IEEE Transactions on Computational Imaging*, 9:649–664, 2023.
- [28] Maria Eugenia Villafane, Nathan Daly, Christine Kimbriel, Catherine Higgitt, and Pier Luigi Dragotti. Multimodal image registration and mosaicking of artworks: an approach based on mutual information. In Haida Liang and Roger Groves, editors, *Optics for Arts, Architecture, and Archaeology (O3A) IX*, volume 12620, pages 12620–8. International Society for Optics and Photonics, SPIE, 2023.
- [29] D. Capel and A. Zisserman. Automated mosaicing with super-resolution zoom. In *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231)*, pages 885–891, 1998.
- [30] Art Certification Experts. Infrared photography - vidicon vs. digitized composite image, 2024. [Online; accessed 21-June-2024].
- [31] Stefan Klein and Marius Staring. Elastix, the manual. 10 October 2019. <https://elastix.lumc.nl/download/elastix-5.0.0-manual.pdf>, 2019. [Online; accessed 25-May-2023].
- [32] Chinmay Belthangady and Loic A. Royer. Applications, promises, and pitfalls of deep learning for fluorescence image reconstruction. *Nature Methods*, 16:1215–11225, 2019.
- [33] Hamid Reza Boveiri, Raouf Khayami, Reza Javidan, and Alireza Mehdizadeh. Medical image registration using deep neural networks: A comprehensive review. *Computers & Electrical Engineering*, 87:106767, 2020.
- [34] Jure Žbontar and Yann LeCun. Computing the stereo matching cost with a convolutional neural network. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1592–1599, 2015.
- [35] Marc-Michel Rohé, Manasi Datar, Tobias Heimann, Maxime Sermesant, and Xavier Pennec. Svf-net: Learning deformable image registration using shape matching. In *Medical Image Computing and Computer Assisted Intervention, MICCAI 2017*, pages 266–274. Springer International Publishing, 2017.

- [36] G. Balakrishnan, A. Zhao, M. R. Sabuncu, J. Guttag, and A. V. Dalca. Voxelmorph: A learning framework for deformable medical image registration. *IEEE Transactions on Medical Imaging*, 38(8):1788–1800, 2019.
- [37] Malte Hoffmann, Benjamin Billot, Douglas N Greve, Juan Eugenio Iglesias, Bruce Fischl, and Adrian V Dalca. Synthmorph: learning contrast-invariant registration without acquired images. *IEEE Transactions on Medical Imaging*, 41(3):543–558, 2022.
- [38] Ty Nguyen, Steven W. Chen, Shreyas S. Shivakumar, Camillo Jose Taylor, and Vijay Kumar. Unsupervised deep homography: A fast and robust homography estimation model. *IEEE Robotics and Automation Letters*, 3(3):2346–2353, 2018.
- [39] Damon M. Conover, John K. Delaney, Paola Ricciardi, and Murray H. Loew. Towards automatic registration of technical images of works of art. In David G. Stork, Jim Coddington, and Anna Bentkowska-Kafel, editors, *Computer Vision and Image Analysis of Art II*, volume 7869, page 78690C. International Society for Optics and Photonics, SPIE, 2011.
- [40] Aleksandra Pizurica, Ljiljana Platisa, Tijana Ruzic, Bruno Cornelis, Ann Doots, Maximilian Martens, Helene Dubois, Bart Devolder, Marc De Mey, and Ingrid Daubechies. Digital image processing of the ghent altarpiece: Supporting the painting’s study and conservation treatment. *IEEE Signal Processing Magazine*, 32(4):112–122, 2015.
- [41] B. Cornelis, T. Ružić, E. Gezels, A. Doots, A. Pizurica, L. Platiša, J. Cornelis, M. Martens, M. De Mey, and I. Daubechies. Crack detection and inpainting for virtual restoration of paintings: The case of the ghent altarpiece. *Signal Processing*, 93(3):605–619, 2013. Image Processing for Digital Art Work.
- [42] Matthias Alfeld and José A.C. Broekaert. Mobile depth profiling and sub-surface imaging techniques for historical paintings—a review. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 88:211–230, 2013.
- [43] Francesco Paolo Romano and Koen Janssens. Issue information. *X-Ray Spectrometry*, 48(4):247–318, 2019.
- [44] Henry Chopp, Alicia McGeachy, Matthias Alfeld, Oliver Cossairt, Marc Walton, and Aggelos Katsaggelos. Image processing perspectives of x-ray fluorescence data in cultural heritage sciences. *IEEE BITS the Information Theory Magazine*, 2(1):20–35, 2022.
- [45] Thompson et al. X-ray data booklet. *Lawrence Berkeley National Laboratory, University of California Berkeley, CA, USA*, 8(4), 2001.
- [46] R E Van Grieken and A A Markowicz. Handbook of x-ray spectrometry: Methods and techniques. 2001.

- [47] T. Arai, Norbert Langhoff, Alexandre Simionovici, Vladimir Arkadiev, W. Knüpper, T. Cechák, J. Leonhardt, J. Chavanne, A. Erko, A. Bjeoumikhov, Antonov Alexander, Burkhard Beckhoff, I. Grigorieva, Birgit Kanngießer, B. Vidal, F. Scholze, A. Longoni, C. Fiorini, L. Strüder, and W. Malzer. *Handbook of Practical X-Ray Fluorescence Analysis*. 05 2006.
- [48] P. van Espen, H. Nullens, and F. Adams. A computer analysis of x-ray fluorescence spectra. *Nuclear Instruments and Methods*, 142(1):243–250, 1977.
- [49] B. Vekemans, K. Janssens, L. Vincze, F. Adams, and P. Van Espen. Comparison of several background compensation methods useful for evaluation of energy-dispersive x-ray fluorescence spectra. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 50(2):149–169, 1995.
- [50] Vekemans, Bart and Janssens, K and Vincze, Laszlo and Adams, F and Vanespen, P. Analysis of x-ray-spectra by iterative least-squares (AXIL) : New developments. *X-ray Spectrometry*, 23(6):278–285, 1994.
- [51] V.A. Solé, E. Papillon, M. Cotte, Ph. Walter, and J. Susini. A multiplatform code for the analysis of energy-dispersive x-ray fluorescence spectra. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 62(1):63–68, 2007.
- [52] Matthias Alfeld and Koen Janssens. Strategies for processing mega-pixel x-ray fluorescence hyperspectral data: A case study on a version of Caravaggio’s painting Supper at Emmaus. *J. Anal. At. Spectrom.*, 30:777–789, 02 2015.
- [53] C.G. Ryan and D.N. Jamieson. Dynamic analysis: on-line quantitative pixe microanalysis and its use in overlap-resolved elemental mapping. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 77(1):203–214, 1993.
- [54] C.G. Ryan, D.N. Jamieson, C.L. Churms, and J.V. Pilcher. A new method for on-line true-elemental imaging using pixe and the proton microprobe. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 104(1):157–165, 1995. Nuclear Microprobe Technology and Applications.
- [55] Matthias Alfeld, Joana Pedroso, Margriet van Eikema Hommes, Geert Van der Snickt, Gwen Tauber, Jorik Blaas, Michael Haschke, Klaus Erler, Joris Dik, and Koen Janssens. A mobile instrument for in situ scanning macro-xrf investigation of historical paintings. *Journal of Analytical Atomic Spectrometry*, 28:760–767, 03 2013.
- [56] Damon M. Conover. *Fusion of Reflectance and X-ray Fluorescence Imaging Spectroscopy Data for the Improved Identification of Artists’ Materials*. PhD thesis, The George Washington University, 2015.

- [57] M. Vetterli, P. Marziliano, and T. Blu. Sampling signals with finite rate of innovation. *IEEE Transactions on Signal Processing*, 50(6):1417–1428, 2002.
- [58] Jose Antonio Urigüen, Thierry Blu, and Pier Luigi Dragotti. FRI sampling with arbitrary kernels. *IEEE Transactions on Signal Processing*, 61(21):5310–5323, 2013.
- [59] C. Prony, F.M. Gaspard, and R. De Baron. Essai expérimental et analytique: Sur les lois de la dilatabilité de fluides élastique et sur celles de la force expansive de la vapeur de l’alkool, a différentes températures. *Journal de l’école Polytechnique de Paris*, 1(22):24–76, 1795.
- [60] R. Morrison L. Keith, A. Roy and P. Schade. Leonardo da Vinci’s ‘Virgin of the Rocks’: Treatment, Technique and Display. <https://www.nationalgallery.org.uk/research/research-resources/technical-bulletin/technical-bulletin-volume-32>, 2011. [Online; accessed 16-February-2024].
- [61] Edward Rosten, Reid Porter, and Tom Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 32:105–119, 2010.
- [62] Edward Rosten and Tom Drummond. Fusing points and lines for high performance tracking. In *IEEE International Conference on Computer Vision*, volume 2, pages 1508–1511, October 2005.
- [63] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, volume 1, pages 430–443, May 2006.
- [64] Wikipedia. Features from accelerated segment test — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Features_from_accelerated_segment_test, 2023. [Online; accessed 10-October-2023].
- [65] Wikipedia. Difference of gaussians — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Difference_of_Gaussians, 2023. [Online; accessed 10-October-2023].
- [66] Michael W. Davidson and The Florida State University. Optical microscopy digital imaging, 2023. [Online; accessed 09-October-2023; <https://micro.magnet.fsu.edu/primer/java/digitalimaging/processing/diffgaussians/index.html>].
- [67] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *Computer Vision – ECCV 2010*, pages 778–792. Springer Berlin Heidelberg, 2010.
- [68] Wikipedia. Norm (mathematics) — Wikipedia, the free encyclopedia. [https://en.wikipedia.org/wiki/Norm_\(mathematics\)#p-norm](https://en.wikipedia.org/wiki/Norm_(mathematics)#p-norm), 2023. [Online; accessed 04-November-2023].

- [69] Deepanshu Tyagi. Introduction to orb (oriented fast and rotated brief). 2019. [Online; accessed 10-October-2023].
- [70] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [71] Peter Rousseeuw. Least median of squares regression. *Journal of the American statistical association*, 79:871–880, 12 1984.
- [72] Charles V. Stewart. Robust parameter estimation in computer vision. *SIAM Review*, 41(3):513–537, 1999.
- [73] Cerys Jones. *An Optimised Method for Multispectral Imaging of Documentary Material*. PhD thesis, University College London, 2020.
- [74] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. IEEE, 1999.
- [75] David G Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 60, pages 91–110, 2004.
- [76] Matthias Alfeld, Silvia Pedetti, Philippe Martinez, and Philippe Walter. Joint data treatment for vis–nir reflectance imaging spectroscopy and xrf imaging acquired in the theban necropolis in egypt by data fusion and t-sne. *Comptes Rendus Physique*, 19(7):625–635, 2018. Physics and arts / Physique et arts.
- [77] Matthias Alfeld. Datahandlerp wiki - fusion and analysis for hyperspectral data, 2019. [Online; accessed 28-December-2023].
- [78] Prasanna Sridhar, Abhishek Dutta, and Andrew Zisserman. Image Compare - source code. https://gitlab.com/vgg/image-compare-lib/-/blob/main/src/transform.cpp?ref_type=heads#L280, 2022. [Online; accessed 27-Oct-2023].
- [79] Northwestern University and Marc Walton. NU-ACCESS - Registration of MA-XRF and HSI cubes. <https://github.com/NU-ACCESS/Registration-MA-XRF-HSI-cubes/blob/main/README.md>, 2022. [Online; accessed 27-Oct-2023].
- [80] Marc Walton, Lindsay Oakley, Stephanie Zaleski, Billie Males, and Oliver Cossairt. Improved spectral imaging microscopy for cultural heritage through oblique illumination. *Heritage Science*, 8, 2020.

- [81] Piotr Foryś, Robert Sitnik, Jakub Markiewicz, and Eryk Bunsch. Fast adaptive multimodal feature registration (famfr): an effective high-resolution point clouds registration workflow for cultural heritage interiors. *Heritage Science*, 11, 2023.
- [82] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571, 2011.
- [83] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In Aleš Leonardis, Horst Bischof, and Axel Pinz, editors, *Computer Vision – ECCV 2006*, pages 404–417, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
- [84] Christopher G. Harris and M. J. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, 1988.
- [85] Paul L. Rosin. Measuring corner properties. *Computer Vision and Image Understanding*, 73(2):291–307, 1999.
- [86] Rafael C. Gonzalez and Richard E. Woods. *Digital image processing*. Prentice Hall, Upper Saddle River, N.J., 2008.
- [87] G.P. Penney, J. Weese, J.A. Little, P. Desmedt, D.L.G. Hill, and D.J. Hawkes. A comparison of similarity measures for use in 2-d-3-d medical image registration. *IEEE Transactions on Medical Imaging*, 17(4):586–595, 1998.
- [88] Wikipedia. Limited-memory bfgs — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Limited-memory_BFGS, 2023. [Online; accessed 11-June-2023].
- [89] Utpal Kumar. Numerical optimization based on the l-bfgs method. 2022. [Online; accessed 02-November-2023].
- [90] D. Rueckert, L.I. Sonoda, C. Hayes, D.L.G. Hill, M.O. Leach, and D.J. Hawkes. Nonrigid registration using free-form deformations: application to breast mr images. *IEEE Transactions on Medical Imaging*, 18(8):712–721, 1999.
- [91] Jean Duchon. Interpolation des fonctions de deux variables suivant le principe de la flexion des plaques minces. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, 10(R3):5–12, 1976.
- [92] M. Unser. Splines: A perfect fit for signal and image processing. *IEEE signal processing magazine*, 16(6):22–38, 1999.
- [93] Philippe Thévenaz, Thierry Blu, and Michael Unser. *Image Interpolation and Resampling*, page 393–420. Academic Press, Inc., USA, 2000.

- [94] Aristeidis Sotiras, Christos Davatzikos, and Nikos Paragios. Deformable medical image registration: A survey. *IEEE Transactions on Medical Imaging*, 32(7):1153–1190, 2013.
- [95] P. Viola and W.M. Wells. Alignment by maximization of mutual information. In *Proceedings of IEEE International Conference on Computer Vision*, pages 16–23, 1995.
- [96] William M. Wells, Paul Viola, Hideki Atsumi, Shin Nakajima, and Ron Kikinis. Multi-modal volume registration by maximization of mutual information. *Medical Image Analysis*, 1(1):35–51, 1996.
- [97] André M. F. Collignon, Frederik Maes, D. Delaere, Dirk Vandermeulen, Paul Suetens, and Guy Marchal. Automated multi-modality image registration based on information theory. 1995.
- [98] Frederik Maes, André Collignon, Dirk Vandermeulen, Guy Marchal, and Paul Suetens. Multimodality image registration by maximization of mutual information. volume 16, pages 14–22, 04 1997.
- [99] Wikipedia. Hessian matrix — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Hessian_matrix, 2023. [Online; accessed 01-November-2023].
- [100] Wikipedia. Rosenbrock function — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Rosenbrock_function, 2023. [Online; accessed 01-November-2023].
- [101] Max-Heinrich Laves, Tobias Ortmaier, and Sontje Ihler. Deformable medical image registration using a randomly-initialized CNN as regularization prior. In *Medical Imaging with Deep Learning (MIDL) 2019*, 06 2019.
- [102] Hessam Sokooti, B. D. Vos, Floris F. Berendsen, Boudewijn P. F. Lelieveldt, Ivana Isgum, and Marius Staring. Nonrigid image registration using multi-scale 3d convolutional neural networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2017.
- [103] Xiao Yang, Roland Kwitt, Martin Styner, and Marc Niethammer. Quicksilver: Fast predictive image registration – a deep learning approach. *NeuroImage*, 158:378–396, 2017.
- [104] Wikipedia. Sørensen–dice coefficient — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/S%C3%B8rensen%E2%80%93Dice_coefficient, 2023. [Online; accessed 02-November-2023].
- [105] Guha Balakrishnan, Amy Zhao, Mert Sabuncu, Adrian Dalca, and John Guttag. An unsupervised learning model for deformable medical image registration. pages 9252–9260, 06 2018.

- [106] John Ashburner. A fast diffeomorphic image registration algorithm. *NeuroImage*, 38(1):95–113, 2007.
- [107] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571, 2016.
- [108] Malte Hoffmann, Benjamin Billot, Douglas N Greve, Juan Eugenio Iglesias, Bruce Fischl, and Adrian V Dalca. synthmorph.ipynb, 2022. [Online; accessed 20-April-2023].
- [109] Malte Hoffmann, Benjamin Billot, Douglas N Greve, Juan Eugenio Iglesias, Bruce Fischl, and Adrian V Dalca. Synthmorph: learning image registration without images, 2022. [Online; accessed 20-April-2023].
- [110] Ignacio Arganda-Carreras, Carlos O. S. Sorzano, Roberto Marabini, José María Carazo, Carlos Ortiz-de Solorzano, and Jan Kybic. Consistent and elastic registration of histological sections using vector-spline regularization. In Reinhard R. Beichel and Milan Sonka, editors, *Computer Vision Approaches to Medical Image Analysis*, pages 85–95, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
- [111] Francesca Gabrieli, John K. Delaney, Robert G. Erdmann, Victor Gonzalez, Annelies van Loon, Patrick Smulders, Roy Berkeveld, Robert van Langh, and Katrien Keune. Reflectance imaging spectroscopy (RIS) for Operation Night Watch: Challenges and achievements of imaging Rembrandt’s masterpiece in the glass chamber at the Rijksmuseum. *Sensors*, 21(20), 2021.
- [112] Hamilton Kerr Institute, University Of Cambridge. Unlocking the English Portrait Miniature. <https://unlocking-miniatures.fitzmuseum.cam.ac.uk/>, 2023. [Online; accessed 12-June-2023].
- [113] P. Thevenaz and M. Unser. Optimization of mutual information for multiresolution image registration. *IEEE Transactions on Image Processing*, 9(12):2083–2099, 2000.
- [114] Stefan Klein, Marius Staring, and Josien P. W. Pluim. Evaluation of optimization methods for nonrigid medical image registration using mutual information and b-splines. *IEEE Transactions on Image Processing*, 16(12):2879–2890, 2007.
- [115] J. Dunkerton and M. Spring. Titian after 1540: Technique and style in his later works. <https://www.nationalgallery.org.uk/research/research-resources/technical-bulletin/technical-bulletin-volume-36>, 2015. [Online; accessed 22-November-2023].
- [116] Wikipedia. Client (computing) — Wikipedia, the free encyclopedia. [https://en.wikipedia.org/wiki/Client_\(computing\)](https://en.wikipedia.org/wiki/Client_(computing)), 2023. [Online; accessed 16-November-2023].

- [117] Paolo Uccello. The Battle of San Romano. <https://www.nationalgallery.org.uk/paintings/paolo-uccello-the-battle-of-san-romano>, about 1397 - 1475. [Online; accessed 18-November-2023].
- [118] Sony George and Jon Yngve Hardeberg. Estimation and correction of geometric distortion in pushbroom hyperspectral system for imaging art paintings. In *International Symposium on Electronic Imaging Science and Technology*, 2016.
- [119] Wikipedia. Principal component analysis — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Principal_component_analysis, 2024. [Online; accessed 14-February-2024].
- [120] G. Luo, L. Tian G. Chen, K. Qin, and S. Qian. Minimum noise fraction versus principal component analysis as a preprocessing step for hyperspectral imagery denoising. *Canadian Journal of Remote Sensing*, 42(2):106–116, 2016.
- [121] Wikipedia. False color — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/False_color, 2024. [Online; accessed 14-February-2024].
- [122] J. Padfield, D. Saunders, J. Cupitt, and R. Atkinson. Improvements in the acquisition and processing of x-ray images of paintings. *National Gallery Technical Bulletin*, 23:62–75, 2002.
- [123] Matteo di Giovanni. The Assumption of The Virgin, NG1155, The National Gallery, London UK. <https://www.nationalgallery.org.uk/paintings/matteo-di-giovanni-the-assumption-of-the-virgin>, 2024. [Online; accessed 16-February-2024].
- [124] International Platform for Art Research and Conservation Ltd (IPARC). Multispectral imaging consultation made easy with custom curtain viewer by iparc. https://www.iparc.eu/curtainviewer/bouts_ma_xrf2.html, 2020. [Online; accessed 21-February-2024].
- [125] The National Gallery London, Joe Padfield. Curtain Viewer - A Dominican, with the Attributes of Saint Peter Martyr. <https://jpadfield.github.io/simple-site/curtain-viewer.html>, 2019. [Online; accessed 21-February-2024].