**Indian Statistical Institute**

Doctoral Dissertation


# Statistically Consistent and Novel Image Registration Methods for Various Transformations in Presence of Zooming

*A thesis submitted to the Indian Statistical Institute in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Statistics*

*Author:* **Sujay Das**

*Supervisor:* **Dr. Partha Sarathi Mukherjee**

Interdisciplinary Statistical Research Unit
Statistical Sciences Division

# Declaration of Authorship

I, Sujay Das, declare that this dissertation titled "Statistically Consistent and Novel Image Registration Methods for Various Transformations in Presence of Zooming" and the works presented in it are my own. I confirm the following statements.

- The work was done while in candidature for a research degree at Indian Statistical Institute, Kolkata.
- No part of this dissertation has been submitted for a degree or any other qualification at this institute or elsewhere.
- I have clearly cited all the previously published works that I have consulted for writing this dissertation.
- I have properly acknowledged all main sources of data used in this dissertation and the methods on which I have built my work.
- I have properly acknowledged all my co-authors who helped in developing different parts of this dissertation.

---

**Sujay Das**
Senior Research Fellow in Statistics
Interdisciplinary Statistical Research Unit
Indian Statistical Institute, Kolkata

# Certificate from Supervisor

This is to certify that this doctoral dissertation entitled "Statistically Consistent and Novel Image Registration Methods for Various Transformations in Presence of Zooming," submitted by Sujay Das to Indian Statistical Institute is a record of bonafide research work under my guidance and supervision. The work contained in this dissertation is original and has not been submitted elsewhere for any degree.

---

**Dr. Partha Sarathi Mukherjee**
Associate Professor
Interdisciplinary Statistical Research Unit
Indian Statistical Institute, Kolkata

# List of Publications

## Published/Accepted Research Articles

1. **Das, S.**, Roy, A. and Mukherjee, P.S. "Image registration for zooming: A statistically consistent local feature mapping approach", *Stat*, 2024, 13(1), e664. DOI: 10.1002/sta4.664. [Link]

2. **Das, S.** and Mukherjee, P.S. "Image Registration for Zooming Using Similarity Matching", *Journal of Data Science*, 2024, 22(4), 558–574. DOI: 10.6339/24-JDS1139. [Link]

## Submitted Research Articles

1. "A Statistically Consistent Image Registration Method for Translation, Rotation, and Zooming" with Mukherjee, P.S.

# Abstract

In image processing literature, *image registration* is the process of spatially matching two or more images of the same scene or object obtained at various times, from different perspectives, or by separate sensors in order to simplify comparison, fusion, or analysis.

Its purpose is to identify the spatial transformation that minimizes the differences between related characteristics or intensities in the reference and moved images. Image registration techniques find a wide range of applications in several imaging fields, namely, medical imaging, satellite imaging, fingerprint matching, remote sensing, and image comparison, to name a few. The growing popularity of this imaging field motivates numerous researchers to develop novel and improved methods of image registration. In this dissertation, we develop several image registration techniques under various image transformations, especially when zooming or scaling is involved.

In our first research-work, we develop an intensity-based image registration technique that registers two images efficiently, where one is a zoomed-in version of the other. In order to improve the performance and also to reduce the computational complexity of the method at the same time, we opt for a feature-based approach for the same problem in the second research-work of this dissertation. Motivated by the success of the feature-based approach, we use similar approach to image registration for a more complex image transformation where translation, rotation, and zooming are all involved in our final research-work.

All our proposed methods are backed by extensive numerical studies and comparisons with several state-of-the-art methods, which show our methods' superiority over them. We also provide theoretical justifications when the image resolutions are high.

# Acknowledgements

This doctoral dissertation could never have been completed without the support of several people. I sincerely express my heartfelt gratitude to all of them.

To begin with, I would like to extend my sincere gratitude toward my dissertation supervisor, Dr. Partha Sarathi Mukherjee. I am really indebted to him for being there through thick and thin during this roller coaster ride of my doctoral studies. His continuous support and motivation helped me cope with the frustrations I felt from time to time, majorly because of some non-fruitful days in the field of research or some other reason as well. He first introduced me to the field of image processing and trained me continuously since then to excel in this field. Several valuable inputs from his side have greatly improved the quality of this dissertation. I also wish to thank my co-author of one of the chapters of this dissertation, Mr. Anik Roy for his fruitful insights during the course of discussions with him.

Furthermore, I would like to thank the Director of Indian Statistical Institute, the previous and current Deans of Studies, and all previous and current members of the Ph.D. & D.Sc. Committee of Statistics for having faith in me and giving me financial support in the form of a research fellowship that helped me immensely throughout this journey. I also fondly remember the Research Fellow Advisory Committee of the Statistical Sciences Division for evaluating my progress in a continuous manner, which helped in enhancing the quality of this dissertation and also ensured a productive doctoral studies at Indian Statistical Institute. I also express my gratefulness to the previous and present unit heads of the Interdisciplinary Statistical Research Unit (ISRU) and Statistical Sciences Division for providing me with a great research environment. In the beginning of the Ph.D. tenure, I had to go through a brief course work, the knowledge gained from which helped me overcome many difficulties in the path of the Ph.D. journey. I would like to thank all those course instructors and other distinguished faculty members for being patient with me and allowing me to share a portion of their vast knowledge. I also wish to convey my regards to the staff members of the library, the dean's office, the Computer & Statistical Service Centre, and my unit, ISRU.

I would also like to thank the research scholars of the Interdisciplinary Statistical Research Unit and the Statistical Sciences Division, without whom it would be very difficult for me to finish this journey. I fondly remember my daily tea session with them at the end of the day and lots of pleasant discussions during that session that helped me cope with the frustration and boredom from time to time.

Finally, this acknowledgement would not be complete without giving my mother the credit she deserves for the sacrifices she made for me. She has been the one beacon of light in all the darkness and has always supported and motivated me throughout this endeavor.

Sujay Das
Interdisciplinary Statistical Research Unit
Indian Statistical Institute, Kolkata

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

In the age of social media and digital marketing, images play a crucial role in communications. Understanding how images influence perception, behavior, and communication dynamics online is essential for businesses and marketers. As humans are highly visual beings, knowledge of how the brain processes and interprets visual data is crucial to a number of scientific disciplines, including psychology, neurology, and human-computer interaction. Research on images helps uncover insights into perception, cognition, and decision-making processes. With advancements in artificial intelligence and computer vision, research on images is vital for developing algorithms and technologies capable of understanding, analyzing, and generating visual content. Applications range from autonomous vehicles and medical imaging to surveillance and augmented reality. In the medical field, imaging methods, including MRIs, CT scans, and X-rays, are crucial for diagnosis, treatment design, and monitoring. In this context, research on images aims to develop new imaging modalities, improve diagnosis accuracy, and improve image quality. Remote sensing technologies use images captured by satellites and other platforms to monitor environmental changes, such as deforestation, urbanization, and climate change. Research on image analysis techniques contributes to environmental monitoring efforts and helps in policy-making.

In the context of image processing literature, an "image" typically refers to a two-dimensional array of pixel values, where each pixel represents a small area of an object or scene being captured. Images are often categorized into different types based on their characteristics, attributes, and applications. Some common types of images include binary images, grayscale images, color images, and tensor images, to name a few. Binary images are digital images where each pixel can only take one of two possible values, typically representing the foreground, i.e., the image object(s), and the background. Grayscale images are digital images where each pixel represents the intensity of light at that point,

usually ranging from black (minimum intensity) to white (maximum intensity). In color images, each pixel contains values for multiple color channels, such as red, green, and blue (RGB) or hue, saturation, and value (HSV). A tensor image is a type of digital image represented as a multi-dimensional array, or tensor, where each element of the tensor corresponds to a pixel in the image. Unlike traditional two-dimensional images, which consist of rows and columns of pixels, tensor images can have multiple dimensions, allowing them to represent more complex visual data.

An image must be digitized in terms of both brightness measurement and spatial location in order for computer software to process it. A digitizer is a component found in many image acquisition equipment, such as scanners and cameras, that transforms the captured images into digital formats. All images used in this dissertation are digital grayscale images, unless mentioned otherwise. A bivariate function $f(x, y)$ can be used to express a grayscale image. The function value $f(x, y)$ is proportional to the image's brightness at $(x, y)$ where $(x, y)$ indicates the image's spatial location. In literature, $f(x, y)$ is frequently referred to as the image intensity function evaluated at $(x, y)$.

A matrix $\{f(x, y), i = 1, \ldots, n_1; j = 1, \ldots, n_2\}$ can be used to represent a typical 2D digital image, where $i$ and $j$ are the row and column indices, respectively, as displayed below.

$$\begin{pmatrix} f(1,1) & f(1,2) & \ldots & f(1,n_2) \\ f(2,1) & f(2,2) & \ldots & f(2,n_2) \\ \vdots & \vdots & \ddots & \vdots \\ f(n_1,1) & f(n_1,2) & \ldots & f(n_1,n_2) \end{pmatrix}$$

Each component of the matrix is referred to as an image intensity at the respective pixel. The resolution of the image, $n_1 \times n_2$ depends on the values of $n_1$ and $n_2$, with large values corresponding to the high-resolution image and small values to the low-resolution image. The digitized value of $f(i, j)$ is typically believed to be an integer in the range $[0; L-1]$, with 0 signifying black and $L-1$ denoting white. The shade of the image at the position $(i, j)$ is determined by the value of the quantity $f(i, j)$. $f(i, j)$ is commonly referred to as the image's gray level at the $(i, j)$-th pixel. Figure 1.1 elaborates on how these intensity values, presented in the form of a matrix, get translated into an image. Refer to Gonzalez and Woods (2018), and Qiu (2005) for more detailed discussion on this topic.

From the perspective of image processing, images are considered raw data that can be manipulated, analyzed, and transformed using various algorithms and techniques. These techniques may include filtering, segmentation, feature extraction, feature mapping, and transformation, among others, aimed at tasks such as edge detection, denoising, registration, object recognition, image enhancement, comparison, monitoring, and so forth. We discuss a few of these image processing techniques in the next paragraph.

$$\begin{bmatrix}
0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\
0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\
0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\
0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\
0.5 & 0.5 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0.5 & 0.5 \\
0.5 & 0.5 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0.5 & 0.5 \\
0.5 & 0.5 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0.5 & 0.5 \\
0.5 & 0.5 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0.5 & 0.5 \\
0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\
0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\
0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\
0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5
\end{bmatrix}$$

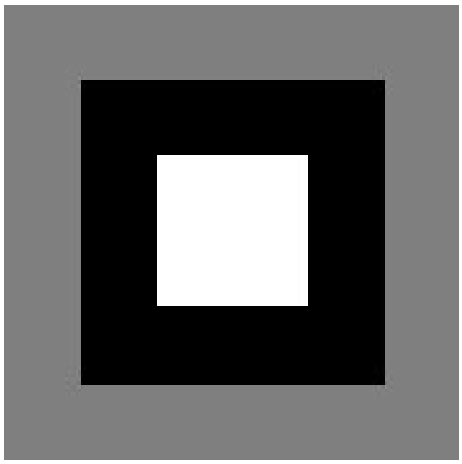


Figure 1.1: The digital grayscale image corresponding to the image intensity values shown by the matrix presented above.

Image denoising enhances the quality of the image by reducing or eliminating unwanted noise that arises from various sources, such as sensor limitations, transmission interference, or environmental factors, while preserving important image features and details. Edge detection is a fundamental image processing technique that identifies and highlights the boundaries or discontinuities within an image, which further helps in various tasks such as image segmentation, object recognition, and feature extraction. Image registration aligns multiple images of the same scene or object to a common coordinate system. Image enhancement improves the quality or visual appearance of digital images to make them more suitable for human perception or for further analysis by machines. Image comparison is the process of analyzing and evaluating the similarities or differences between two or more images. Finally, image monitoring involves the systematic and continuous observation, analysis, and management of images over time to track changes, identify patterns, or detect anomalies. It is commonly used in various fields, such as surveillance, environmental monitoring, remote sensing, and medical imaging.

In this dissertation, we focus on image registration, a very important image process-

ing technique. Image registration has numerous applications in several fields, including medical imaging, satellite imaging, fingerprint matching, remote sensing, and image comparison, to name a few. We can note one such important application in the case of image comparison. If we compare two images of the same scene where one is transformed, e.g., translated, rotated, etc., then the popular image comparison algorithms (e.g., Roy and Mukherjee (2024b)) mentioned in the literature often fail to identify them as the same images and rather detect changes in them. In this case, if we use image registration as a pre-processing tool to align the images and then apply the comparison algorithms, then we can find only the actual changes, if any, between the images. These applications make image registration an extremely important research topic, and more quality research would strengthen this field with more sophisticated methods for the same.

## 1.1 Image Registration

Image registration is a process used in digital image processing to align different images of the same scene or object, taken at different times, from different viewpoints, or by different sensors, so that they match spatially. The goal of image registration is to find a transformation that maps one image, say the moved image onto the other image, i.e., the reference or source image, by accounting for differences due to rotation, translation, scale/zooming, and other transformations present between the two images, which further facilitates comparison, fusion, and analysis. Figure 1.2 gives one such example where image registration has an important role to play in order to facilitate comparison between the displayed images. Image registration is an important stage in image analysis tasks such as image fusion, change detection, and multichannel image restoration, where the final information is derived by combining data from several sources. This technique finds a wide range of applications in varieties of fields like medical imaging (Klein et al., 2009), satellite imaging (Kim and Im, 2003; Chen et al., 2015), fingerprint matching (Liu et al., 2006), remote sensing (Yang et al., 2017; Ye et al., 2017, 2019), image mosaicing (Castanheira de Souza et al., 2012), and weather forecasting (Giri and Sharma, 2011), to name a few. In the next paragraph, we discuss applications of image registration in some of the aforesaid fields, viz., medical imaging, satellite imaging, etc.

Image registration is extensively used in medical imaging. Medical practitioners frequently need to combine data from numerous imaging modalities, such as MRI, CT, PET, and ultrasound, to get a complete picture of a patient's anatomy or disease. Image registration allows for the alignment and integration of these images, resulting in a more complete understanding of the patient's condition. We cite one more of many such instances where image registration is applied in medical imaging. During surgical pro-

Figure 1.2: Original, rotated, and zoomed version of 'cameraman' image. Left to right: Original image, rotated clockwise by $10^0$, scaled/zoomed version.

cedures, especially those involving complex anatomical structures or minimally invasive approaches, image-guided navigation systems use image registration to overlay preoperative imaging data onto the patient's anatomy in real-time. This provides surgeons with enhanced visualization and navigation capabilities, improving surgical accuracy and outcomes. Image registration improves the utility and efficacy of satellite images for a variety of purposes, including better decision-making, resource management, and environmental monitoring at the local, regional, and global levels. Satellite imaging is commonly used to track changes in land use, vegetation coverage, urban growth, and environmental conditions over time. Image registration enables the correct alignment of several satellite images taken at different periods, making it easier to discover and analyze differences between the images. This is critical for applications like urban planning, deforestation monitoring, and catastrophe management. Image registration techniques can be applied to fingerprint matching to improve the accuracy and reliability of biometric authentication systems. Fingerprint images can vary in orientation, scale, and distortion depending on factors such as the pressure used during fingerprint capture or the finger's position on the sensor. Image registration techniques can be used to precisely align fingerprint images, guaranteeing that associated features (such as minutiae points) are properly matched throughout the matching process. These widespread applications make image registration a pretty interesting topic for research.

There are numerous approaches to image registration in the literature. We can broadly classify them into three major categories: rigid registration (Ashburner and Friston, 2007; Debayle and Presles, 2016), affine registration (Song et al., 2013; Shen et al., 2019), and non-rigid registration (Crum et al., 2004; Li and Fan, 2017), depending on the nature of the transformation. *Rigid registration*, also known as rigid body transformation or rigid transformation, is an image registration approach that aligns images using just translation, rotation, and scaling transformations. To obtain the optimum alignment, rigid

registration adjusts just the spatial orientation and location of the objects in the pictures, leaving the shape and size unchanged. This transformation preserves the geometric properties of the objects in the images, such as distances between points, angles, and shapes. This means that the relative positions of corresponding features or landmarks in the images are maintained after the registration process. *Affine registration* is a type of image registration technique used to align and match two or more images by applying an affine transformation. An affine transformation consists of translation, rotation, scaling, and shearing transformations, allowing for a wider range of adjustments compared to rigid registration, which only includes translation, rotation, and scaling. *Non-rigid registration*, also known as deformable registration, is an image registration technique used to align and match two or more images by allowing for local deformations or warping of the images. Unlike rigid or affine registration, which only includes global transformations such as translation, rotation, scaling, and shearing, non-rigid registration enables more flexible adjustments to account for spatial variations and deformations within the images. Non-rigid registration is particularly useful in medical imaging, where images may exhibit complex anatomical deformations, such as in soft tissue deformation during organ motion or in the presence of tumors. It is also employed in computer vision, remote sensing, and other fields where images undergo non-linear transformations or deformations that cannot be adequately captured by rigid or affine transformations.

Image registration techniques can also be divided into two broad categories: intensity-based methods (Xing and Qiu, 2011; Qiu and Xing, 2013b) and feature-based methods (Qiu and Xing, 2013a; Yang et al., 2017), based on the approach used for aligning the images. *Intensity-based registration* is a technique for aligning and matching photographs based on pixel intensities. It entails determining the spatial transformation that best aligns the intensity values between the images. The purpose of intensity-based registration is to minimize disparities or maximize similarity between the intensity distributions of corresponding pixels or regions in images. A similarity metric, such as correlation coefficients, mutual information, normalized cross-correlation, or sum of squared differences (SSD), is defined to quantify the similarity or dissimilarity between the intensity values of corresponding pixels or regions in the images. An optimization algorithm is employed to find the transformation parameters that minimize the difference or maximize the similarity measure between the images. Once the optimal transformation parameters are determined, the images are finally transformed accordingly to achieve alignment. On the other hand, *feature-based registration* is a technique for aligning and matching two or more images using distinctive features or key points. In contrast to intensity-based registration, which is based on the similarity of pixel intensities, feature-based registration identifies and matches certain features, such as corners, edges, blobs, or other salient regions. In this method, the first step is to extract features from the images to be registered. After

extracting features from both images, the next step is to match the corresponding features between them. Once correspondences between features are established, the next step is to estimate the transformation that aligns the images. As we are discussing feature-based image registration, one question that naturally arises is, "What is an image feature?'. We discuss that briefly in the next paragraph.

An *image feature* is a distinct and identifiable characteristic or pattern inside an image that is important for tasks in computer vision and image processing. Features can indicate a variety of properties of an image, including intensity, texture, form, color, and local structures. These features are often derived from the raw pixel data of an image and used to characterize and analyze the content. They provide the foundation for tasks including object identification, recognition, classification, matching, and image retrieval. There are two major types of image features: point-based features and non-point-based features. Curves, edges, regions, and so on are examples of non-point features, whereas point features include corner points, landmark points, centroid points, and so on. We discuss a few of them in the next paragraph.

In image processing and computer vision, a *curve* is a continuous line or trajectory described by a series of connected image points. These points represent the contour or boundary of a distinct object or shape in the image. An *edge* is a fundamental image feature that is characterized by sharp transitions in pixel values and is frequently identified using edge detection techniques. These algorithms attempt to locate regions in an image where there are abrupt changes in intensity or color, generally by computing gradients or derivatives of the image. On the other hand, a *corner* is a point at which an image's brightness or intensity gradient changes significantly in numerous directions. Corners are characterized by the intersection of two or more edges, resulting in a distinct geometric structure that differs from the surrounding pixels. Corner detection algorithms aim to identify these points of interest in an image by analyzing local variations in intensity or color. We conclude by discussing landmark points as a point-based feature. A *landmark point*, as the name suggests, refers to a specific location within an image that is visually distinctive. Landmark points are often characterized by their uniqueness, repeatability across different images, and invariance to variations such as scale, rotation, and illumination changes. Landmark points are often found using feature detection algorithms that examine the image's local features, such as gradients, textures, or color distribution. Now, we have discussed almost all the prerequisites required for the topic of interest of this dissertation, i.e., image registration. In the next section, we shed some light on the relevant and popular research that has been done in this field till now.

## 1.2 Literature Review of Related Research

In the field of computer science, we find several papers focusing on the development of image registration techniques that work on the principle of characterizing specific features of the reference as well as sensed or moved images. Following that, using particular algorithms, the relationships between these features in the sensed and reference images are identified. The work by Moravec (1981) on stereo matching with a corner detector lays the groundwork for image matching utilizing a set of local interest points. Then, Harris et al. (1988) enhance the Moravec detector to ensure repeatability in minor image fluctuations and near edges. Harris demonstrates its effectiveness in motion tracking and 3D structure recovery. The Harris corner detector is now commonly utilized for image-matching applications. Harris also demonstrates its effectiveness in motion tracking and 3D structure recovery (Harris, 1993). The Harris corner detector is now commonly utilized for image-matching applications. These feature detectors, also known as corner detectors, choose any image location with large gradients in all directions at a predefined scale.

Though these methods were initially used for stereo and short-range motion tracking, they were later expanded to more complex issues. Zhang et al. (1995) demonstrate that by selecting plausible matches from a correlation window around each corner, Harris corners may be matched throughout a vast image range. Outliers are then removed by calculating a fundamental matrix representing the geometric constraints between the two perspectives of the rigid scene and excluding matches that do not correspond with the majority solution. Torr (1995) concurrently develops a similar approach for long-range motion matching, using geometric constraints to remove outliers from rigid objects in an image.

The groundbreaking work of Schmid and Mohr (1997) demonstrates that invariant local feature matching can be used to address general image recognition issues in which a feature is matched against a vast image database. They employ Harris corners to choose points of interest. But they use a rotationally invariant descriptor of the local image area instead of matching with a correlation window. This enables features to be matched over arbitrary orientation changes between the two images. They also show that multiple feature matches might achieve universal recognition under occlusion and clutter by recognizing consistent clusters of matched features.

The Harris corner detector is particularly sensitive to changes in image scale; hence, it does not provide a strong foundation for matching images of varied sizes. The work by Lowe (1999) expands the local feature technique to achieve scale invariance. This study introduces a new local descriptor that enhances distinguishing features while being less affected by local image distortions such as 3D viewpoint movement.

There is a large amount of previous research on identifying representations that remain stable as scale changes. Crowley and Parker (1984) accomplish some of the early work in this field, developing a representation that detected peaks and ridges in scale space and related them to a tree structure. The tree structure can be matched across photos with varying scales. Shokoufandeh et al. (1999) work on graph-based matching provides more unique feature descriptors based on wavelet coefficients. Lindeberg (1993) conducts extensive research on the topic of determining an adequate and consistent scale for feature detection.

There has been a significant amount of work on extending local features to be invariant to entire affine transformations (Baumberg, 2000; Mikolajczyk and Schmid, 2002; Schaffalitzky and Zisserman, 2002). This enables invariant matching of planar surface features to changes in orthographic 3D projection, which is often accomplished by resampling the picture in a local affine frame. Although these approaches are not entirely affine-invariant because they begin with initial feature sizes and positions chosen in a non-affine-invariant manner, due to the high cost of exploring the entire affine domain.

There are other types of features that have been proposed for use in recognition. One type of feature is one that makes use of image contours or area boundaries, which should make them less susceptible to disruption by cluttered backdrops near object boundaries. Matas et al. (2004) demonstrate that their maximum stable extremal regions can yield a significant number of matching features while maintaining stability. Mikolajczyk et al. (2003) create a new descriptor that leverages local edges while ignoring irrelevant neighboring edges, allowing them to detect stable features even near the borders of small shapes superimposed on background clutter. Nelson and Selinger (1998) demonstrate successful local feature extraction using picture contour groups. Pope and Lowe (2000) find that hierarchical grouping of image contours is effective for identifying objects with low texture detail.

The history of visual recognition research includes work on a wide range of different image qualities that can be used to quantify features. Carneiro and Jepson (2002) propose phase-based local features that indicate the phase, not the magnitude, of local spatial frequencies, potentially improving illumination invariance. Schiele and Crowley (2000) propose the use of multidimensional histograms to summarize the distribution of measurements inside image regions. This feature may be very useful for recognizing textured objects with changeable forms. Basri and Jacobs (1997) illustrate the usefulness of extracting local area boundaries for recognition. Color, motion, figure-ground discrimination, area form descriptors, and stereo depth cues are all useful qualities to consider. The local feature strategy may easily absorb novel feature types since extra features contribute to robustness when they give correct matches but do little harm other than computational

cost.

There are an extensive number of research articles in the literature that develop novel and improved algorithms to detect and describe image features. Lowe (2004) describes a method, viz., Scale Invariant Feature Transform (SIFT), for extracting distinguishable scale and rotation invariant features from images, which may then be utilized to do accurate matching between various views of an object or scene. In order to detect the SIFT features of an image, they first identify repeatable points in a pyramid of scaled images and then detect maxima and minima in the difference-of-Gaussian(DOG) pyramid. These features are proven to give strong matching across a wide range of affine distortion, 3D perspective changes, noise addition, and illumination changes. Bay et al. (2008) demonstrate a fast and efficient scale and rotation invariant interest point detector and descriptor named Speeded-Up Robust Features (SURF). The significant speed increase in the method is due to the use of integral pictures, which substantially reduce the number of operations for simple box convolutions, regardless of scale. Then the research in this field progressed in the direction of increasing the efficiency of the methods and reducing their run-time at the same time.

Then, a high-speed feature detector, viz., the Features from Accelerated Segment Test (FAST) corner detector, is suggested by Rosten and Drummond (2006). Calonder et al. (2011) propose a method to compute a binary descriptor, Binary Robust Independent Elementary Features, or BRIEF, on the basis of an intensity difference test. As binary descriptors are very popular to compare image features and perform their task very quickly while taking up less memory space, BRIEF is extremely fast and efficient to construct and match. Rublee et al. (2011) first proposed the ORB algorithm in 2011 as a FAST and BRIEF-based technique. The FAST algorithm extracts interest points from images, while the BRIEF approach extracts descriptors. Leutenegger et al. (2011) then propose an adaptive and high-performance algorithm, viz., Binary Robust Invariant Scalable Keypoints (BRISK), to detect keypoints in an image that is less computationally complex at the same time. The key to speed is the use of a novel scale-space FAST-based detector in conjunction with the building of a bit-string description from intensity comparisons obtained by dedicated sampling of each keypoint neighborhood. Then, Alcantarilla et al. (2012) in their paper introduce a multi-scale 2D feature detection and description technique based on non-linear scale space, contrary to the previous approaches that used Gaussian scale space. Alcantarilla and Solutions (2011) then improve their previous work by using a numerical scheme called Fast Explicit Diffusion (FED) (Weickert et al., 2016) which decreases the run-time of feature detection in non-linear scale spaces.

We trace some contributions from the Statistics community in the field of image registration, although they are very few in number. We start by citing the article by Qiu and

Xing (2010). They propose an intensity-based image registration technique that does not assume any parametric form for the geometrical transformation to be used for registration. This nonparametric mapping transformation allows discontinuity at certain places in the design space. Then, Xing and Qiu (2011) describe a unique intensity-based picture registration approach that employs nonparametric local smoothing. This method estimates the mapping transformation for a particular pixel locally in a neighborhood after taking into account certain image properties. Because of the flexibility of local smoothing, this technique eliminates the need for a parametric mapping transformation and also allows for a discontinuity in the transformation. Qiu and Xing (2013b) try to resolve the ill-posed nature of the image registration problem related to geometrical transformation not being well defined. Several notions, such as 2-D degenerate pixels, 2-D partial degenerate pixels, 1-D degenerate pixels, and 1-D partial degenerate pixels, have been defined and applied to describe the local features of the transformation $T$. Based on these ideas, methods for recognizing 2-D and 1-D non-degenerate pixels and calculating $T$ have been presented. Qiu and Xing (2013a) explore the problem of image registration through a feature-based approach. They study a new type of feature that has two properties: (i) the detected image features should provide us with enough information to accurately approximate the geometric matching transformations, and (ii) they should be easily identified by a computer system, allowing the full feature extraction method to be automated. Then, they use a thin plate spline (TPS) geometric transformation model to map these features. We also find some 3-D image registration techniques in the literature.

Song and Qiu (2017a) suggest a 3-D image registration technique using local smoothing statistical methods. This method does not assume any parametric form or other global regularity conditions for the transformation. Song and Qiu (2017b) solve the 3-D image registration problem using an affine transformation for the geometric transformation involved in the image registration problem. We conclude our discussion of the current section by citing a paper by Song and Qiu (2018). Here, they investigate a fast-computing environment for 3-D image registration using distributed parallel computing. The proposed 3-D image registration approach is based on Taylor's expansion and 3D local kernel smoothing.

Next, we discuss image registration techniques for diffeomorphic image registration. Diffeomorphic registration is achieved by modeling the flow of one image, i.e., the moving image, into the fixed image using a pair of vector fields. The first field moves the moving image towards the midpoint, whereas the second field moves from the midpoint to the fixed image. The vector fields are defined at a discrete set of points in a grid across the field-of-view, and the fineness of the grid determines the level of anatomical detail that can be achieved in the registration. Beg et al. (2005) propose a path-breaking solution for large deformation diffeomorphic metric mapping (LDDMM). In this paper, they study the

Euler-Lagrange equations for the solution of the large deformation diffeomorphic metric mapping problem. In this problem there are two images $I_0$ and $I_1$ are connected via the diffeomorphic change of coordinates $I_0 \circ \varphi^{-1} = I_1$ given and connected via the diffeomorphic change of coordinates where $\varphi = \phi_1$ is the end point at $t = 1$ of curve $\phi_t$, $t \in [0, 1]$ satisfying $\dot{\phi}_t = \nu_t(\phi_t), t \in [0, 1]$ with $\phi_0 = id$. The variational problem takes the form

$$\underset{\nu:\dot{\phi}_t=\nu_t(\phi_t)}{\operatorname{argmin}} \left( \int_0^1 ||\nu_t||_V^2 + ||I_0 + \phi_1^{-1} - I_1||_{L^2}^2 \right),$$

where $||\nu_t||_V$ is an appropriate Sobolev norm on the velocity field $\nu_t(.)$, and the second term enforces matching of the images with $||.||_{L_2}$ representing the squared-error norm. In this paper, they derive the Euler-Lagrange equation for the solution of the variational problem stated before in the space of smooth velocity vector fields. They express the implementation details of a gradient algorithm using the Euler-Lagrange equation for numerical computation of the solution to the aforementioned problem, and in particular, a semi-Lagrangian method of particle flows to integrate the vector fields, being a nondissipative method as opposed to traditional Eulerian methods, which are highly dissipative. The optimizer of the problem provides a geodesic shortest length path in the space of the group of transformations. Computational anatomy has many existing codes organized around diffeomorphic registration, such as DARTEL (Ashburner, 2007), Fast LDDMM (Zhang et al., 2017), and so forth.

In medical image registration, non-linear image transformation is common, and this problem is commonly viewed as an optimization problem. Interesting research problems in this field include the choice of image metric, strategies for efficient and global optimization, theoretical properties of the obtained solutions, and so forth. Song et al. (2017) provide a review on medical image registration as an optimization problem. Deep-learning-based diffeomorphic image registration method, such as (Mok and Chung, 2020), optical flow-based image registration method, which is a duality-based algorithm for total variation, TV-L1 (Pock et al., 2007) are designed to perform mon-linear image registration for various applications.

Some of the aforementioned articles motivate us in the work of this dissertation, and we use a few of them to compare the performance of the methods proposed in this dissertation. We discuss them in a little elaborate manner in the next section.

## 1.3 Description of Popular State-of-the-art Methods

First, we discuss a few techniques from a class of methods that we use as primary competitors of our proposed methods in this dissertation.

### Scale-Invariant Feature Transform (SIFT)

Lowe (1999) proposes the Scale-Invariant Feature Transform (SIFT) as a new way of creating image features. It is a vision-based algorithm that detects and describes scale-invariant and rotation-invariant region-based characteristics in an image. Finding repeated points in a pyramid of scaled images and locating maxima and minima in the difference-of-Gaussian (DOG) pyramid are two methods used to create the SIFT characteristics of an image (Catarious Jr et al., 2006), as displayed in Figure 1.3.

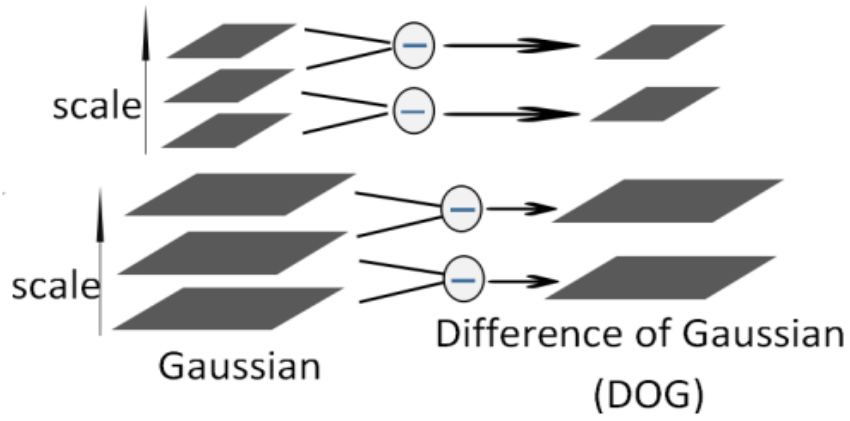



Figure 1.3: Gaussian pyramid and DOG pyramid. This image has been taken from Figure 2 of Li (2017).

Koenderink (1984) and Lindeberg (1993) have demonstrated that the Gaussian function is the sole feasible scale-space kernel under a number of plausible assumptions. Thus, a function, $L(x, y, \sigma)$, obtained by convolving a variable-scale Gaussian, $G(x, y, \sigma)$, with an input image, $I(x, y)$, defines the scale space of an image:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y),$$

where $*$ is the convolution operation in $x$ and $y$, and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}.$$

Lowe suggests the use of scale-space extrema in the difference-of-Gaussian function convolved with the image, $D(x, y, \sigma)$, which can be computed from the difference of two nearby scales separated by a constant multiplicative factor k, to efficiently detect stable key-point locations in scale space:

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma)) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned}$$

The original picture is represented by $I(x, y)$, the Gaussian filter at scale $\sigma$ is represented by $G(x, y, \sigma)$, and the convolution of image $I$ with the Gaussian blur is represented by $L(x, y, \sigma)$.

By fitting a three-dimensional quadratic function and eliminating the low-contrast candidate points and edge response points, SIFT descriptors are produced by appending orientation parameters $m$ and $\theta$ to each feature point with position $(x, y)$.

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - (L(x,y-1))^2},$$
$$\theta(x,y) = a\,tan\,2\left[\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right].$$

As SIFT is distinct, relatively easy to extract, robust to changes in lighting, noise, and slight changes in viewpoint, it is frequently utilized in applications such as object detection, robot localization and mapping, panorama, and ego-motion estimation (Wensley et al., 1978; Zhou et al., 2009).

**Speeded Up Robust Features (SURF)**

Bay et al. (2006) first introduced Speeded Up Robust Features (SURF), a robust local feature detector that is somewhat inspired by SIFT but operates differently in certain ways. At first, a Hessian matrix is computed for each pixel location, and non-maximal suppression is then applied to extract SURF features:

$$H(x,\sigma) = \begin{pmatrix} Ł_{xx}(x,\sigma) & Ł_{xy}(x,\sigma) \\ Ł_{xy}(x,\sigma) & Ł_{yy}(x,\sigma) \end{pmatrix}.$$

In this case, $L$ is defined in a similar way as in the case of SIFT, and the second partial derivatives of $L$ are $L_{yy}$, $L_{xy}$, and $L_{xx}$. The Hessian blob detector's determinant is approximated using an integer, and feature descriptors are determined by adding the Haar wavelet response surrounding the interest point rather than using the gradient histogram. In computer vision tasks, the feature-detection algorithm SURF outperforms SIFT in terms of speed and is robust to various image transformations (Juan and Gwun, 2009).

**The Binary Robust Independent Elementary Features (BRIEF)**

Calonder et al. (2011) suggest the BRIEF algorithm as a quick binary descriptor. Interest points can be extracted using FAST, HARRIS, SIFT, or SURF in the BRIEF algorithm. Following the Gaussian filtering denoising process, the descriptor is obtained by picking N pairs ($N = 125$) of Gauss-distributed feature points from the vicinity of an interest point and obtaining binary values by comparing the two intensities in each feature pair $< P1, P2 >$. An N-dimensional binary encoding is created by adding together the N binary values.

BRIEF can match features and build descriptors quickly. Rather than using a conventional gradient histogram to construct descriptors, the approach uses random response detection and computes the Hamming distance between features. BRIEF's low rotation invariance and restriction to modest rotation conditions are its drawbacks, though.

### Oriented Fast and Rotated Brief (ORB)

Rublee et al. (2011) first introduced the ORB algorithm in 2011 as a technique built on top of FAST (Rosten and Drummond, 2006) and BRIEF (Calonder et al., 2011). The FAST technique is used to extract interest points from images, and the BRIEF approach is used to obtain descriptors.

Figure 1.4 illustrates how the ORB algorithm, an enhanced version based on FAST and BRIEF, established a two-dimensional system with the key point as its center and the line linking the key point and the centroid of the connection region as its $x$ axis, achieving better rotation invariance.

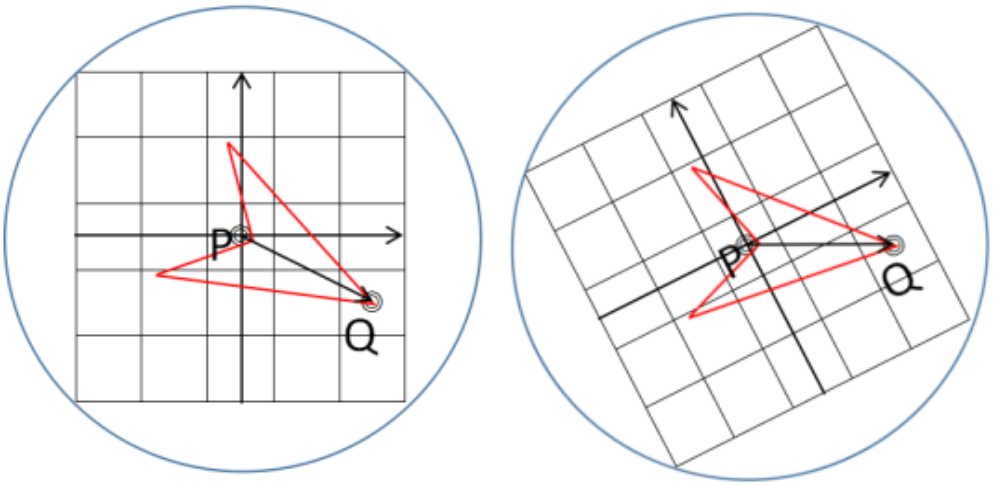



Figure 1.4: The 2-dimensional system in ORB to achieve rotation invariance. This image is taken from Figure 3 of Li (2017).

This method makes use of the intensity centroid (Rosin, 1999), a straightforward yet useful indicator of corner orientation. It is possible to impute an orientation using the intensity centroid vector, which makes the assumption that a corner's intensity is offset from its center. Rosin characterizes a patch's moments as:

$$M_{pq} = \sum_{x,y} x^p y^q I(x, y),$$

and then the centroid can be found as:

$$C = \left( \frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right).$$

From the centroid, C, to the corner's center, O, a vector can be constructed. Thus, the patch's orientation is as follows:

$$\theta = \text{atan2}(M_{01}, M_{10}),$$

where atan2 is arctan's quadrant-aware version.

While the ORB algorithm takes advantage of the FAST approach to speed up feature detection, it is unable to address the issue of poor scale invariance. Studies demonstrate that ORB's speed is more than 100 times faster than SIFT's and more than 10 times faster than SURF's. Furthermore, techniques like the use of image pyramids are employed in real image and video processing to enhance scale invariance.

**Binary Robust Invariant Scalable Keypoints (BRISK)**

BRISK is a binary descriptor (Leutenegger et al., 2011) that enhances the ideas of BRIEF. The keypoint detection methodology is based on the work by Mair et al. (2010) for identifying regions of interest in the image, with an emphasis on computational efficiency. In order to attain invariance to scale, which is essential for high-quality keypoints, they go one step ahead and use the FAST scores as a saliency metric to look for maxima both in the image plane and in scale-space. The true size of each keypoint in the continuous scale-space is estimated by the BRISK detector, even though it discretizes the scale axis at coarser intervals than other high-performance detectors.

Following the selection of keypoints, a sample pattern is implemented in the keypoint's surrounding area. Short-distance and long-distance pairs are the two subsets of pairs of pixels surrounding the keypoint. The direction of the feature point is ascertained, and local intensity gradients from long-distance pairings are computed. This orientation is used to rotate short-distance pairs. The results of pairwise brightness comparison tests at short distances are concatenated to provide a BRISK descriptor. This descriptor is expressed as a 64-dimensional feature vector and is made up of a bit string with a length of 512 (64 bytes).

Next, we discuss a few of those methods that motivate us to develop the proposed methods in this dissertation. The common setup for the following research works is the same.

They try to register two images, R and M. R(x, y) and M (x, y) represent their genuine

image intensity functions. It is believed that they are related in the following ways:

$$M(T_1 x, y), T_2(x, y)) = R(x, y), \text{ for } (x, y) \in \Omega,$$

where $T(x, y) = (T_1(x, y), T_2(x, y))$ is an unknown geometrical transformation to estimate, and $\Omega$ is the design space of the image $R$.

**Intensity-based Image Registration by Xing and Qiu (2011)**

In this paper, they first define the concepts of local degenerate pixels and local non-degenerate pixels. They show that only in the vicinity of local non-degenerate pixels or locations where the reference image R's image intensity function is not smooth can the geometrical transformation $T(x, y)$ be accurately approximated. They suggest an intensity-based image registration (IBIR) procedure with four main phases, which are outlined below, based on that outcome. According to them, at a given point $(x, y) \in \Omega$,

$$\begin{pmatrix} T_1(x, y) \\ T_2(x, y) \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} b(x, y) \\ c(x, y) \end{pmatrix}.$$

Estimating $T(x, y)$ is therefore equal to estimating $(b(x, y), c(x, y))$.

By using Taylor's expansion, if $M$ contains the first-order partial derivatives at $(x, y)$ and both $b(x, y)$ and $c(x, y)$ are small,

$$M(T_1(x, y), T_2(x, y)) = M(x, y) + M'_x(x, y)b(x, y) + M'_y(x, y)c(x, y) + o(||T(x, y) - (x, y)||).$$

Next, they solve the following minimization problem to estimate $b(x, y)$ and $c(x, y)$:

$$\min_{b(x,y),c(x,y)} \sum_{i,j} [Z_M(x_i, y_j) - Z_R(x_i, y_j) + \widehat{M}'_x(x_i, y_j)b(x, y) + \widehat{M}'_y(x_i, y_j)c(x, y)]^2 K_{h_n},$$

where $Z_M(x, y)$ and $Z_R(x, y)$ are the intensity values at the pixel location $(x, y)$ for the images $M$ and $R$, respectively, and $K_{h_n} = K(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n})$, with $K$ being the bivariate density kernel function with unit circular support. It is then shown that the above minimization problem has the following solution:

$$\begin{pmatrix} \widehat{b}(x, y) \\ \widehat{c}(x, y) \end{pmatrix} = \frac{\begin{pmatrix} K_{22} & -K12 \\ -K12 & K_{11} \end{pmatrix} \begin{pmatrix} K_1^* \\ K_2^* \end{pmatrix}}{K_{11}K_{22} - K_{12}^2}. \tag{1.3.1}$$

The estimator defined in Eqn. (1.3.1) is implied to be poorly defined in locations where the following equation holds:

$$K_{11}K_{22} - K_{12}^2 = 0. \tag{1.3.2}$$

On the basis of the above set-up, they suggest the four-step registration process:

1. At first, they use an edge detector to find the edge pixels for the observed reference image. Then they decide which pixels in $R$ are continuity pixels.

2. Then, they decide on the continuity pixels of $R$ on the basis of the number of edge pixels contained by a circular neighborhood around those pixels. Then they demarcate between those continuity pixels on the basis of Eqn. (1.3.2) as non-degenerate pixels and degenerate pixels. Pixels that abide by the Eqn. (1.3.2) are termed as degenerate pixels and called non-degenerate pixels, otherwise.

3. Next, for all non-degenerate pixels of $R$, they consider a neighborhood around them and then find a similar neighborhood in $M$ on the basis of a similarity metric, mean squared deviation (MSD). And then map the central point of the neighborhood in $M$ to the corresponding non-degenerate pixel in $R$.

4. For a local degenerate pixel $(x, y)$, they propose to find the closest non-degenerate pixel, say, $(x_1, y_1)$, and then define:

$$\widehat{T}_1(x, y) = (x, y) + (\widehat{b}(x_1, y_1), \widehat{c}(x_1, y_1)).$$

Finally, they define the estimator $T(x, y)$ in the following way:

$$\begin{aligned} \widehat{T}(x, y) &= \widehat{T}_1(x, y) \quad \text{if } \; MSD(\widehat{T}_1(x, y); (x, y)) \leq MSD((x, y); (x, y)), \\ &= (x, y) \qquad \text{Otherwise.} \end{aligned}$$

**Feature-based Image Registration by Qiu and Xing (2013a)**

In this paper, they propose an image registration technique consisting of three steps.

1. To begin with, they extract the non-degenerate feature points in the images $R$ and $M$, respectively. Here, they define non-degenerate feature points in a similar way as in Xing and Qiu (2011).

2. Next, they match the feature points of $R$ and $M$ using a similar concept stated in point 3 of the previously discussed four-step intensity-based image registration technique by Xing and Qiu (2011). Here, they use the mean squared deviation (MSD) as the similarity metric.

3. Finally, they find the relationship between these matched feature points of $R$ and $M$, respectively. Here, they use the Thin Plate Spline (TPS) (Debayle and Presles, 2016; Bookstein, 1989) interpolation to find the geometric transformation defined in the entire design space.

In the next section, we briefly discuss the research work that we have accomplished in this dissertation.

## 1.4 Brief Description of This Dissertation

In the first part of this dissertation, we consider the problem of registering two images of the same resolution where one is a zoomed-in version of the other. Here, we propose an intensity-based approach to solve this problem. While numerous feature-based techniques have been proposed in the literature to solve zooming transformations, direct intensity-based image registration approaches are hard to find. The central idea of this procedure is that if we select a pixel in the zoomed-in image and consider a neighborhood around it, then we might find a similar neighborhood in the reference image. But that neighborhood would be in a distant place compared to the zoomed-in image. Therefore, we construct our method in such a way that, corresponding to a certain neighborhood of a pixel in the zoomed-in image, it can find a similar neighborhood and corresponding pixel in the reference image. We make use of certain similarity metrics, viz., mean squared deviation (MSD), mean absolute deviation (MAD), and correlation, to check the similarity between the neighborhood of the pixels in the reference image and the zoomed image. This method's performance gets better when the image resolution increases. Our intensity-based approach surpasses many state-of-the-art feature-based registration algorithms in images with a low number of features, as it does not rely on the image features for the task of registration. The proposed method's simple construction and lack of requirement for explicit feature recognition in images are its main advantages. As this method is simple in its construction, it is easy to understand and implement.

The next part of our dissertation focuses on solving the problem of registering two images of the same scene, where one is a zoomed-in version of the other, but from a different approach. Intensity-based approaches are generally computationally complex, as the intensities of all pixel locations of the reference and the moved images are examined to check for similarity between them. So, here we opt for a feature-based approach to reduce the run-time of the process significantly. Feature-based approaches generally revolve around the principles of feature selection and feature mapping. In this research, we propose a new and efficient image registration technique based on non-point features in the presence of a zooming transformation. The primary feature of our suggested approach is a non-point feature, namely the edges. It is to be noted that, as long as the image is not trivial, this feature is always present in an image. First, we identify the edge points in the zoomed-in image as well as the reference image. We employ a feature mapping approach to map the edge pixels of the reference and the zoomed image since the number

of edge points in them is not the same. Furthermore, it is unknown which edge pixel of the reference image has moved to which edge pixel of the zoomed image. The feature mapping of our approach works on the principle of finding the most similar neighborhood and respective edge point in the reference image, corresponding to a selected edge point and its neighborhood in the zoomed image. The similarity is measured through two similarity metrics, namely mean squared difference (MSD) and mean absolute difference (MAD). After the feature mapping step, we explain the relationship between the mapped features of the reference and the zoomed image using a mathematical model. Lastly, we use the previously mentioned relationship to generate the registered image. The performance of this method improves significantly as the image resolution increases. The computational complexity of this approach also decreases substantially in comparison to the intensity-based approach discussed in the previous paragraph. One noticeable advantage of the proposed method is that it works even when the point features in the images are small in number, something most state-of-the-art methods fail to do. In addition to that, we also estimate the zooming center and the zooming factor.

Finally, we engage in the pursuit of a more complex registration problem where we register two images of the same resolution, and the moved image includes translation, rotation, and zooming of the image objects. Our solution is also a feature-based approach, with edges being the only non-point feature. We choose this feature for the same reason as earlier. Similar to other feature-based registration techniques, this approach also involves two major steps, viz., feature extraction and feature mapping. We approach the image registration problem under the aforementioned transformations in the following manner. First, we detect edge points in both the moved/sensed and reference images. We use a feature mapping strategy as earlier to map the edge pixels of the reference and moved images because the numbers of edge points in the two images are possibly different, and it is unknown which edge pixel from the reference image has moved to which edge pixel of the moved image. After the feature mapping step, we end up with two sets, having the same cardinality, of feature points from the reference image and the moved image. Then, we move on to the step of explaining the relationship between these mapped features through a mathematical model. But this model, consisting of too many parameters, makes the job of uniquely estimating model parameters troublesome. To get rid of this problem, we assume that the image center is also the zooming center and the rotation center. We first make use of a strategy to estimate the rotation angle from the mapped feature points themselves. This strategy also takes care of the discrepancy in the estimation that might creep in due to translation. Next, we plug in the estimate of the rotation angle in the said mathematical model, and use the model to explain the relationship between the mapped feature points of the reference image and the moved image. Our proposed approach performs well in a variety of scenarios. In case there are only a few point features in

the images, this method outperforms some of the most advanced state-of-the-art image registration algorithms.

This dissertation focuses only on rigid-body registration and global scaling, which are commonly applicable in certain types of images, such as satellite images, scenery images, and images used in manufacturing industries. In these applications, rigid-body transformation and global scaling occur due to the image framing and camera positioning. Non-linear image transformations are rather uncommon in these contexts. Non-linear and diffeomorphic image registrations are used in other types of applications, such as registration of brain MRI with an existing template, registration of brain MRIs of different subjects, and so forth, where the image objects are indeed of unequal shapes. This dissertation provides jump regression-based solutions for the first type of scenarios involving only translation, rotation, and scaling, which can be used as a stepping stone to develop more sophisticated solutions for the more general non-linear image registrations under the jump regression framework.

## 1.5 Contributions and Novelty of This Dissertation

Primary contributions and novelty of this dissertation are as follows:

- We consider the problem of various types of *image registration*. We first note that the state-of-the-art methods can work well only on certain types of images and image transformations. We provide our solutions that perform well on a much wider range of situations.
- This dissertation aims to make the first stride towards widening the scope of rigid-body image registration to include zooming or scaling under the framework of Jump Regression Analysis (JRA). It is probably the first step to include more general non-linear registrations under JRA.
- In Chapter 2, we propose an intensity-based image registration approach to register two images where one is a zoomed-in version of the other. The proposed method works on nearly all types of images that we encounter in practice, whereas the popular feature-based registration methods fail to work if the number of prominent image features is low. The proposed idea of local similarity matching is simple to understand and implement.
- Furthermore, we propose a feature-based approach in Chapter 3 to solve the same problem as above. While being computationally less complex than the previous solution, this approach surpasses several state-of-the-art feature-based methods if

the number of point-features in the images is scarce. In our solution, we use the *edges* as the image features, which are available in all images we encounter in practice. Hence, our method works well on nearly all types of practical images.

- Finally, in Chapter 4, we provide our solution to the image registration problem that considers image transformations such as translation, rotation, and zooming/scaling. In this case also, our solution is feature-based, using the *edges* of the images. This proposed approach works well in a wide range of cases and outperforms even the most powerful feature-based algorithms if the number of point-features in the images is small. Our solution works even if the rotation angle is large, in which case many rigid-body image registration methods fail to work.

- Our proposed solutions to various *image registration* problems should be useful in various applications, such as image comparisons, image monitoring, and so forth. These areas of research are becoming popular with the use of images as the primary data in many industries. The use of *artificial intelligence (AI)* in medical science, manufacturing industries, agriculture, defense, etc., also demands efficient *image registration* so that the problems of geometric misalignment of the images are resolved.

- Not only are our proposed methods of *image registration* useful in many practical applications, the adapted versions of the central ideas of our solutions can be used to develop many other techniques in data science, such as non-linear image registration.

## 1.6 Organization of This Dissertation

The rest of the dissertation is organized as follows. In Chapter 2, we propose an intensity-based image registration for two images of the same resolution where one is a zoomed-in version of the other. We also provide several numerical examples and comparisons with some state-of-the-art methods. Chapter 3 introduces our proposed feature-based approach for the same problem as earlier and also provides a few numerical examples in support of its superiority over some cutting-edge algorithms in the literature. Finally, in Chapter 4, we present a feature-based image registration technique under a complex transformation of the image objects that includes translation, rotation, and scaling/zooming. Numerical comparisons with existing approaches are also provided. We conclude by discussing the major limitations of our approaches and a number of future directions of our research in Chapter 5. We provide the details of the proofs of the Propositions in Appendix A.

# Chapter 2

# Intensity-based Image Registration for Zooming Transformation[1]

## 2.1 Introduction

In the image processing literature, various methods of image registration are discussed, both intensity-based and feature-based. However, most methods in the literature do not take into account the possibility that the two images may not be zoomed to the same extent. It may happen sometimes that two images are zoomed unequally due to the varying distance of the camera lens from the same image object, or artificial zooming of the image object in one image to get a closer look at the image object. Therefore, if we have two images where one is a zoomed-in version of the other, then it poses before us the problem of image registration. A good image registration method will enable us to compare the images, if needed. Figure 2.1 illustrates the problem visually.

Although there are many feature-based methods developed by computer scientists for addressing the issue of zooming transformations, direct intensity-based image registration approaches for addressing zooming are not easy to find. In fact, the contribution of the statistics community in this regard is scarce. Therefore, we consider this as the first problem of this dissertation from a statistician's perspective.

In this chapter, we consider two images of equal resolution where one is a zoomed version of the other, and propose an intensity based method to register them. The central idea of this procedure is that, if we select a pixel in the zoomed-in image and consider a neighborhood around it, then we might find a similar neighborhood in the reference

[1]This chapter is based on the publication Das and Mukherjee (2024): **Das, S.**, and Mukherjee, P.S. "Image Registration for Zooming Using Similarity Matching", *Journal of Data Science*, 2024, 22(4), 558–574. DOI: 10.6339/24-JDS1139.

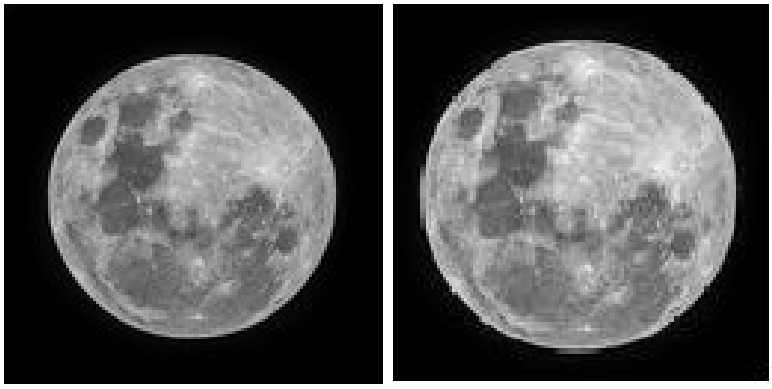


Figure 2.1: From left to right: The reference and the zoomed image.

image. But that neighborhood would be at a distant place compared to the zoomed-in image. Therefore, we construct our method in such a way that, corresponding to a certain neighborhood of a pixel in the zoomed-in image, it can find a similar neighborhood and corresponding pixel in the reference image. One major advantage of the proposed method is that it is very simple in its construction, and it does not require explicit detection of image features.

In this chapter, we describe our proposed registration method in Section 2.2, and provide its theoretical justification in Section 2.3. Section 2.4 presents our numerical studies. In section 2.5, we compare our method with some state-of-the-art methods. We conclude by providing a few remarks on this method in Section 2.6.

## 2.2 Proposed Image Registration Procedure

### 2.2.1 Assumptions

In this chapter, we consider zooming to be the only transformation of the image object. Translation and rotation of the image object are not considered in this chapter. Moreover, we assume that the true image intensity function of the background of the image is constant. The resolutions of both the images, i.e., reference and zoomed images, are the same, although the last assumption is not critical.

### 2.2.2 Description of the Problem

We have two images of the same object or scene. One is the actual image, which we call the reference image, and the other is the zoomed version of the reference image. Our goal is to register these two images, i.e., we want to find a one-to-one mapping between these

two images, or, in other words, we want to derive a mathematical function such that once we put the function on one image, it will return the other image.

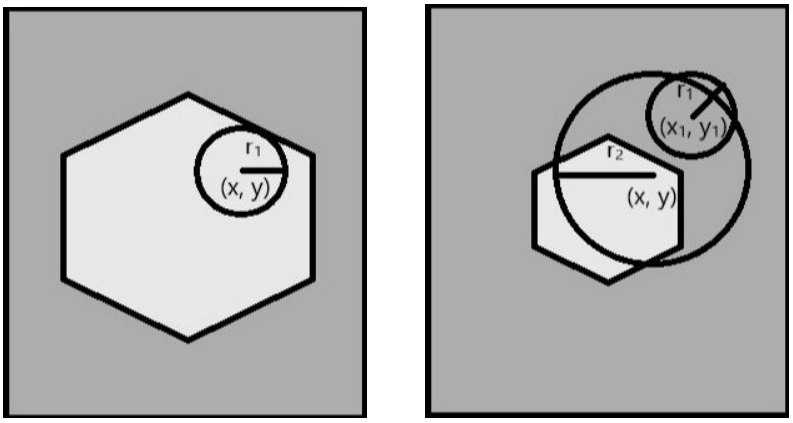
First, we briefly describe the zooming process. A specific region (e.g., the center of the image) of the image is being selected as the target region for zooming. Then that region is zoomed while keeping the resolution the same as that of the reference image.

### 2.2.3 The Proposed Image Registration Procedure

We propose the following steps for performing the registration procedure for zooming:

**Step 1:** For a given pixel $(x, y)$ in the zoomed image, we consider a circular neighborhood of radius $r_1$ around that pixel, denoted as $O(x, y, r_1)$.

**Step 2:** Then, in the reference image, around the same coordinates as that given pixel, we consider a circular neighborhood of radius $r_2$, denoted as $O(x, y, r_2)$, where $r_2 \geq r_1$. For any pixel $(x_1, y_1)$ in that neighborhood $O(x, y, r_2)$, we consider a circular neighborhood of radius $r_1$ around $(x_1, y_1)$, denoted as $O(x_1, y_1, r_1)$.




(a) (b)

Figure 2.2: (a) Zoomed image, (b) Reference image.

**Step 3:** Next, we compare the two neighborhoods, i.e., $O(x, y, r_1)$ of the zoomed image, and all possible choices of $O(x_1, y_1, r_1)$ corresponding to the reference image on the basis of three measures: mean square difference (MSD), mean absolute difference (MAD), and correlation (CC) and choose the neighborhood $O(x_1, y_1, r_1)$ of the reference image that is most similar to the neighborhood $O(x, y, r_1)$ of the zoomed image with respect to these three measures. Subsequently, we map the pixel $(x, y)$ of the zoomed image to that $(x_1, y_1)$ of the reference image. These three measures are defined below:

$$\text{MSD}((x_1, y_1); (x, y)) = \frac{1}{N} \sum (I_R(x_1 + s, y_1 + t) - I_Z(x + s, y + t))^2,$$

$$\text{MAD}((x_1, y_1); (x, y)) = \frac{1}{N} \sum |I_R(x_1 + s, y_1 + t) - I_Z(x + s, y + t)|,$$

$$\text{CC}((x_1, y_1); (x, y)) = \frac{\sum (I_R - \bar{I}_R)(I_Z - \bar{I}_Z)}{\sqrt{\sum (I_R - \bar{I}_R)^2} \sqrt{\sum (I_Z - \bar{I}_Z)^2}},$$

where all the above summations are taken over the set $\{(s, t) : (x_1+s, y_1+t) \in O(x_1, y_1, r_1)\}$, $N$ is the number of pixels in $O(x_1, y_1, r_1)$, $I_{(.)}(a, b)$ denotes the intensity of the image pixel (a,b), $R$ denotes the reference image, and $Z$ denotes the zoomed image. Therefore, the transformed or mapped pixels are as follows:

$$\arg\min_{(x_1,y_1) \in O(x,y,r_2)} \text{MSD}((x_1, y_1); (x, y)),$$

$$\arg\min_{(x_1,y_1) \in O(x,y,r_2)} \text{MAD}((x_1, y_1); (x, y)),$$

$$\arg\max_{(x_1,y_1) \in O(x,y,r_2)} \text{CC}((x_1, y_1); (x, y)),$$

corresponding to the measures MSD, MAD, and CC, respectively. There are two tuning parameters in this procedure, which are the radii $r_1$ and $r_2$ of the neighborhoods. The performance of this procedure depends highly on the selection of these parameters. In the next section, we address the selection process for these parameters.

### 2.2.4 Selection of the Procedure Parameters

In the procedure described above, $r_1$ is the radius of the neighborhood $O(x, y, r_1)$ corresponding to some pixel $(x, y)$ of the zoomed image, and $r_2$ is the radius of the neighborhood $O(x, y, r_2)$ corresponding to the same pixel of the reference image. We use the trial-and-error method to select $r_1$ and $r_2$. We find the mapping for some possible choices of $r_1$ and $r_2$ and create the registered version of the zoomed image. After that, we compare the similarity between the reference image and the registered version of the zoomed image on the basis of MSD. Then we select those values of $r_1$ and $r_2$ for which $\text{MSD}(Z, R, r_1, r_2)$ is minimized, i.e.,

$$(r_1, r_2) = \arg\min_{\widetilde{r}_1, \widetilde{r}_2} \sum_{(x,y)} (I_G(x, y, \widetilde{r}_1, \widetilde{r}_2) - I_R(x, y))^2,$$

where $I_G(x, y, \widetilde{r}_1, \widetilde{r}_2)$ is the intensity of the pixel $(x, y)$ of the registered version of the zoomed image, and $\widetilde{r}_1$ and $\widetilde{r}_2$ are the parameters used for the registration procedure.

## 2.3 Theoretical Justification

**Proposition 2.3.1.** *Let $(x, y)$ be an edge pixel of an image of resolution $n \times n$, and there is a zoomed version of the same image. In the zoomed image, the pixel $(x, y)$ has moved to $(x_1, y_1)$, where $x_1 = x + c_1$ and $y_1 = y + c_2$, $(c_1, c_2) \in \mathbb{R}^2$. When we apply our mapping procedure with window widths $r_1$ and $r_2$ to register these two images, then $(x_1, y_1)$ in the zoomed image is being mapped to $(x_2, y_2)$ in the registered image. If $n \to \infty$ and $r_2 \to \infty$ such that $r_2/n \to 0$, then*

$$\frac{1}{n^2}[(x_2 - x)^2 + (y_2 - y)^2] \xrightarrow{P} 0,$$

*where $\xrightarrow{P}$ denotes convergence in probability.*

**Proof:** We provide a proof of the proposition in Appendix A.1.1.

## 2.4 Numerical Studies

**Numerical studies on the 'simulated' image of resolution $64 \times 64$:**

In order to apply this method, we simulate an image of resolution $64 \times 64$ and the zoomed version of the same image, where the resolutions of the image and zoomed version of the image are equal. In the simulated reference image, the central square region is of resolution $42 \times 42$, and the intensities of its pixels are all 1, and the intensities of all other image pixels are 0. Then, in order to complete the construction of the simulated image, we add a Gaussian noise with a mean value 0 and a standard deviation of 0.01 to the intensities of each pixel. In the zoomed image, the resolution of the central region is $52 \times 52$. The resolutions of the pixels in the region covering the central region are all 0 as in the simulated reference image. Finally, we add a Gaussian noise of mean value 0 and standard deviation 0.01 to each pixel of the zoomed image.

Then, for each combination of the values of $r_1$ and $r_2$, we run the proposed registration procedure 10 times and find out the sample mean and sample standard deviation of the mean squared error (MSE) values corresponding to the similarity measures MSD, MAD, and CC. Since mean squared error (MSE) is a commonly used measurement of pixelwise difference of two images, we use it on the registered and reference images to assess the performance of the concerned registration methods. If the registration performance is good, MSE would be small, and if the registration performance is poor, MSE would be large. In order to choose optimal values of $r_1$ and $r_2$, we run this procedure for 7 choices of $r_2$, viz., 3 to 9 while keeping $r_1$ at 3. Here we see that, for these choices of $r_1$ and $r_2$,

the minimum value of MSE appears at $(3, 5)$ for MSD, MAD, and CC measures. So, we anticipate that the optimal choice of $r_2$ is 5. Then, in order to find the optimal choice of $r_1$, we run this procedure for 4 choices of $r_1$, viz., 2 to 5, when $r_2$ is at 5. Finally, we choose that combination of $r_1$ and $r_2$ for which the calculated MSE is the smallest. We present those MSE values in Table 2.1.

| Mean (standard deviation) | | | |
|---|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm | Correlation |
| (3,3) | 0.3327 (0.0013) | 0.3362 (0.0012) | 0.4724 (0.0035) |
| (3,4) | 0.2337 (0.0015) | 0.2339 (0.0011) | 0.4026 (0.0046) |
| (3,5) | 0.1016 (0.0042) | 0.1004 (0.0029) | 0.314 (0.0073) |
| (3,6) | 0.1111 (0.0046) | 0.1126 (0.0038) | 0.3598 (0.0031) |
| (3,7) | 0.1192 (0.002) | 0.1186 (0.0034) | 0.3673 (0.0043) |
| (3,8) | 0.1198 (0.0036) | 0.122 (0.0035) | 0.3714 (0.0121) |
| (3,9) | 0.1255 (0.0017) | 0.1305 (0.0019) | 0.4081 (0.0053) |
| (3,10) | 0.1332 (0.006) | 0.1387 (0.005) | 0.4136 (0.0046) |
| (1,5) | 0.105 (0.0035) | 0.1087 (0.0032) | 0.4069 (0.0046) |
| (2,5) | **0.0827 (0.0024)** | **0.0809 (0.0021)** | 0.3691 (0.0056) |
| (4,5) | 0.1223 (0.0013) | 0.1201 (0.0018) | 0.2391 (0.0035) |
| (5,5) | 0.1408 (0.0023) | 0.1385 (0.0012) | **0.1636 (0.0029)** |

Table 2.1: Table for comparing the MSD values for the simulated image. The rationale behind presenting only this set of $(r_1, r_2)$ values is provided in the second paragraph of Section 2.4.

We see from Table 2.1 that the optimal choices of $r_1$ and $r_2$ are 2 and 5 for the registration procedure using MSD and MAD measures. And the optimal choices for $r_1$ and $r_2$ for the CC method are 5 and 5, respectively. Therefore, using these optimal parameters for the specified methods, we run the algorithm and create the registered images. These registered images thus created contain some disjoint pixels (very small in number) where there is no mapping found according to this registration procedure, and as a result, the intensity of these pixels remains unavailable. Then we take a neighborhood of width 2 around these pixels and perform local smoothing. Finally, we get the registered

images corresponding to different measures. We follow the same process for the rest of the examples mentioned in this chapter. These images are shown in the second row of Figure 2.3.

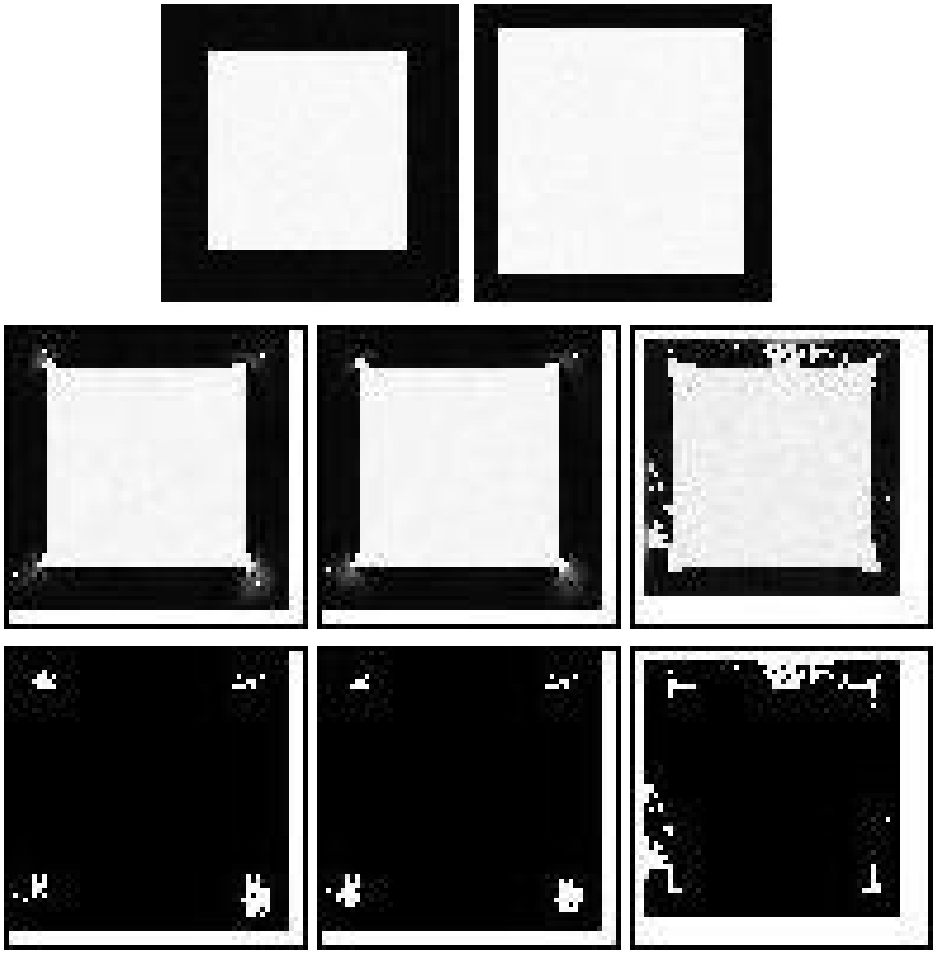


Figure 2.3: These figures are regarding the simulated image. Row 1 (from left to right): the reference image and the zoomed image; Row 2 (from left to right): registered images under $L_1$-norm, $L_2$-norm, and CC method, respectively; Row 3 (from left to right): anomaly images under $L_1$-norm, $L_2$-norm, and CC method, respectively.

In order to check the usefulness of these methods, we compare registered images under all three methods with the reference image and create three matrices named 'anomaly' under the respective methods. We compare each pixel of the registered image with the corresponding pixel of the original (reference) image, and on the basis of that comparison, we assign some value to the respective location of anomaly matrices. Here, the inputs of the anomaly matrices are obtained as below. If the absolute difference between the image intensities of the registered and the original image is less than 15 percent of the range of intensities of the zoomed image, then the anomaly matrix will get a value 0 at that location and a value of 1 elsewhere. Then, from these anomaly matrices, we create three anomaly images under these registration procedures. These images are presented in the third row of Figure 2.3.

**Numerical studies on the 'girl' image of resolution $128 \times 128$:**

For our next example, we consider a 'girl' image of resolution $128 \times 128$ and select its central region of resolution $116 \times 116$. We then zoom in on it while keeping the resolution the same as the original one. We show these two images in the first row of Figure 2.4.

Next, we add a Gaussian noise of mean value 0 and standard deviation 0.01 to each pixel of the two images before applying the method. We add this noise because, while applying this method, we found out that in some neighborhoods we get standard deviation values of 0 and, as a result, the correlation value is becoming indefinite. Then, for each combination of the values of $r_1$ and $r_2$, we run the proposed registration procedure 10 times and find out the sample mean and sample standard deviation of the mean squared error (MSE) values corresponding to the similarity measures MSD, MAD, and CC. In order to choose optimal values of $r_1$ and $r_2$, we run this procedure for 8 choices of $r_2$, viz., 3 to 10, while keeping $r_1$ at 3, and find out the sample mean and sample standard deviation of MSE values corresponding to the measures MSD, MAD, and CC. Here we see that, for these choices of $r_1$ and $r_2$, the minimum value of MSE appears at $(3, 6)$ for MSD and MAD measures and at $(3, 5)$ for CC measures. So, we anticipate that the optimal choice of $r_2$ is either 5 or 6. Then, in order to find the optimal choice of $r_1$, we run this procedure for 5 choices of $r_1$, viz., 2 to 6 when $r_2$ is 6, and for 4 choices of $r_1$, viz., 2 to 5 when $r_2$ is 5. Finally, we choose the values of $r_1$ and $r_2$ for which the calculated MSE value is the smallest. We present those MSE values in Table 2.2.

In this example, we see from Table 2.2 that the optimal choices of $r_1$ and $r_2$ are 2 and 6 for the registration procedure using MSD and MAD measures, and the optimal choices for $r_1$ and $r_2$ for the registration procedure using the CC method are 5 and 6, respectively. Therefore, using these optimal parameters for the specified methods, we run the algorithm and create the registered images. Those images are placed side by side in the second row of Figure 2.4.

Next, we compare the original image with each of the registered images and create three anomaly matrices under three registration procedures. Here, the inputs of the anomaly matrices are obtained as below. If the absolute difference in image intensities between registered and reference images is less than 15 percent of the range of intensities of the zoomed image, then the anomaly matrix will get a value 0 at that location and a value of 1 elsewhere. From these anomaly matrices, we create three anomaly images under these registration procedures. The images are presented in the third row of Figure 2.4.

**Numerical studies on the 'peppers' image of resolution $256 \times 256$:**

For our next example, we consider the 'peppers' image of resolution $256 \times 256$, select

| Mean (standard deviation) | | | |
|---|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm | Correlation |
| (3,3) | 0.1172 (0.0026) | 0.1164 (0.0023) | 0.1256 (0.0023) |
| (3,4) | 0.1091 (0.0027) | 0.109 (0.0031) | 0.1244 (0.0031) |
| (3,5) | 0.0988 (0.0022) | 0.0995 (0.002) | 0.1221 (0.002) |
| (3,6) | 0.0972 (0.0022) | 0.0965 (0.0021) | 0.1238 (0.0021) |
| (3,7) | 0.1027 (0.002) | 0.102 (0.002) | 0.1249 (0.002) |
| (3,8) | 0.1064 (0.0021) | 0.1061 (0.0017) | 0.1311 (0.0017) |
| (3,9) | 0.1143 (0.002) | 0.1133 (0.0017) | 0.1385 (0.0017) |
| (3,10) | 0.1187 (0.0017) | 0.1175 (0.0013) | 0.1472 (0.0013) |
| (2,6) | **0.0742 (0.0029)** | **0.0753 (0.0031)** | 0.1189 (0.0043) |
| (3,6) | 0.0826 (0.0029) | 0.0835 (0.0025) | 0.1087 (0.0023) |
| (4,6) | 0.0921 (0.002) | 0.0924 (0.0019) | 0.1032 (0.0029) |
| (5,6) | 0.1009 (0.0026) | 0.1006 (0.0025) | **0.1025 (0.0026)** |
| (6,6) | 0.1073 (0.0021) | 0.1067 (0.0022) | 0.1071 (0.0022) |
| (2,5) | 0.084 (0.0029) | 0.0863 (0.0026) | 0.1256 (0.0034) |
| (4,5) | 0.1004 (0.002) | 0.1003 (0.0022) | 0.1094 (0.0019) |
| (5,5) | 0.1074 (0.0025) | 0.107 (0.0023) | 0.1119 (0.0028) |

Table 2.2: Table for comparing the MSD values for the 'girl' image. The rationale behind presenting only this set of $(r_1, r_2)$ values is provided in the second paragraph of the numerical studies on the 'girl' image.

its central region of resolution $246 \times 246$, and then zoom it while keeping the resolution the same as the original one. We show these two images in the first row of Figure 2.5.

Next, for each combination of the values of $r_1$ and $r_2$, we run the proposed registration procedure 10 times and find out the sample mean and sample standard deviation of the mean squared error (MSE) values corresponding to the similarity measures MSD, MAD, and CC. In order to choose optimal values of $r_1$ and $r_2$, we run this procedure for 7

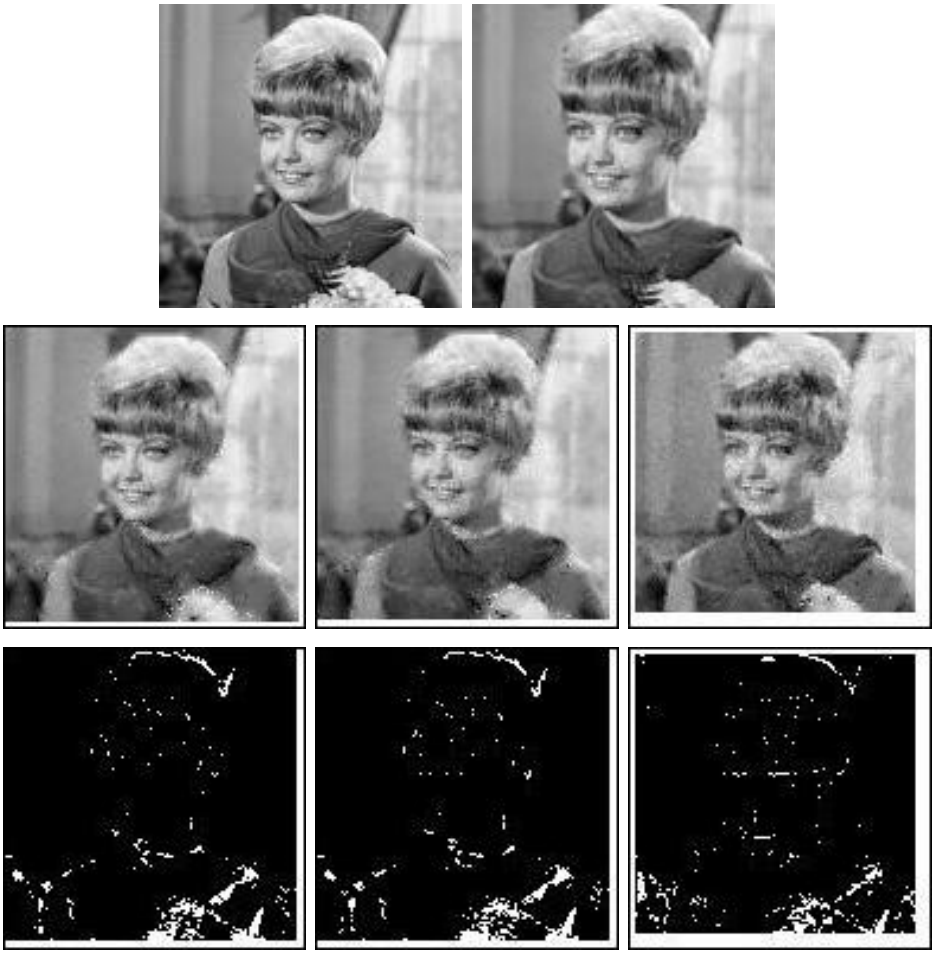


Figure 2.4: These figures are regarding the 'girl' image. Row 1 (from left to right): the reference image and the zoomed image; Row 2 (from left to right): registered images under $L_1$-norm, $L_2$-norm, and CC method, respectively; Row 3 (from left to right): anomaly images under $L_1$-norm, $L_2$-norm, and CC method, respectively.

choices of $r_2$, viz., 3 to 9, while keeping $r_1$ at 3, and find out the sample mean and sample standard deviation of MSE values corresponding to the measures MSD, MAD, and CC. Here we see that, for these choices of $r_1$ and $r_2$, the minimum value of MSE appears at $(3, 6)$ for MSD, MAD, and CC measures. So, we anticipate that the optimal choice of $r_2$ is 6. Then, in order to find the optimal choice of $r_1$, we run this procedure for 5 choices of $r_1$, viz., 2 to 6 when $r_2$ is 6. Finally, we choose the values of $r_1$ and $r_2$ for which the calculated MSE value is the smallest. We present those MSE values in Table 2.3.

In this example, we see from Table 2.3 that the optimal choices of $r_1$ and $r_2$ are 2 and 5 for the registration procedure using MSD and MAD measures, and the optimal choices for $r_1$ and $r_2$ for the CC method are 4 and 5, respectively. Therefore, using these optimal parameters for the specified methods, we run the algorithm and create the registered images. Those images are placed side by side in the second row of Figure 2.5.

Next, we compare the original image with each of the registered images and create three

| Mean (standard deviation) | | | |
|---|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm | Correlation |
| (3,3) | 0.104 (0.0023) | 0.1047 (0.0023) | 0.1211 (0.0019) |
| (3,4) | 0.0927 (0.0021) | 0.0932 (0.0021) | 0.1165 (0.0022) |
| (3,5) | 0.122 (0.0017) | 0.121 (0.0018) | 0.1453 (0.0015) |
| (3,6) | 0.0691 (0.003) | 0.0685 (0.0029) | 0.0952 (0.0031) |
| (3,7) | 0.0907 (0.002) | 0.0901 (0.0021) | 0.1202 (0.0016) |
| (3,8) | 0.0966 (0.002) | 0.0958 (0.002) | 0.1296 (0.002) |
| (3,9) | 0.1025 (0.002) | 0.1013 (0.0015) | 0.1378 (0.0017) |
| (2,6) | **0.0628 (0.0029)** | **0.0626 (0.0024)** | 0.1032 (0.0059) |
| (4,6) | 0.0774 (0.0029) | 0.0773 (0.0028) | **0.0939 (0.0026)** |
| (5,6) | 0.0873 (0.0025) | 0.0872 (0.0025) | 0.0965 (0.0025) |
| (6,6) | 0.0965 (0.0025) | 0.0964 (0.0025) | 0.1018 (0.0024) |

Table 2.3: Table for comparing the MSD values for the 'peppers' image. The rationale behind presenting only this set of $(r_1, r_2)$ values is provided in the second paragraph of the numerical studies on the 'peppers' image.

anomaly matrices under three registration procedures. Here, the inputs of the anomaly matrices are obtained as below. If the absolute difference between the image intensities of the registered and reference images is less than 15 percent of the range of intensities of the zoomed image, then the anomaly matrix will get a value 0 at that location and a value of 1 elsewhere. Then, from these anomaly matrices, we create three anomaly images under these registration procedures. These images are presented in the third row of Figure 2.5.

This method might not work satisfactorily in cases where images have too much uniformity in a neighborhood. Then this method might fail to distinguish between two similar pixels and might end up overlapping one with the other. This tile image and its zoomed version, presented in the first row of Figure 2.6, are one such example where this method might not work well.

In this example, we use the above method and find out that the optimal choices of $r_1$ and $c$ are 2 and 2, respectively, for the registration procedure using the MAD, MSD, and CC measures. Therefore, using these optimal parameters for these methods, we run the

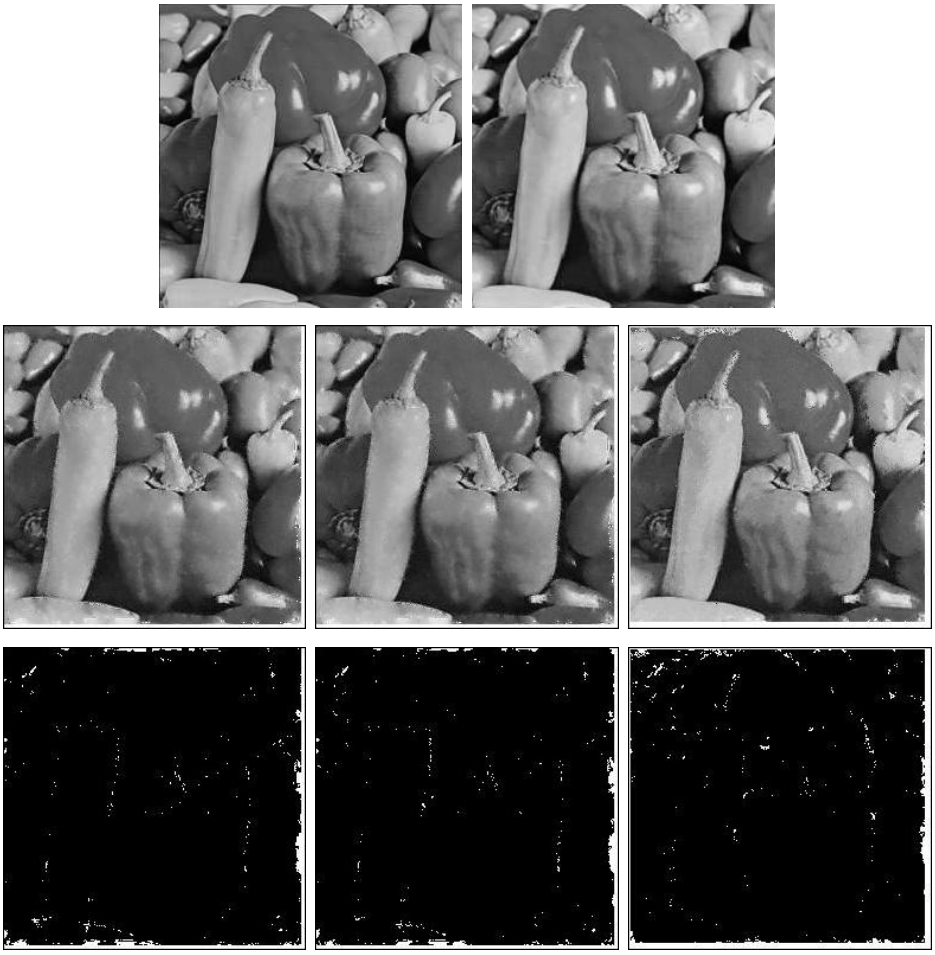


Figure 2.5: These figures are regarding the 'peppers' image. Row 1 (from left to right): the reference image and the zoomed image; Row 2 (from left to right): registered images under $L_1$-norm, $L_2$-norm, and CC method, respectively; Row 3 (from left to right): anomaly images under $L_1$-norm, $L_2$-norm, and CC method, respectively.

algorithm and create the registered images. Those images are placed side by side in the second row of Figure 2.6.

In this example, we also compare the original image with each of the registered images and create three anomaly matrices under three registration procedures. Here, the inputs of the anomaly matrices are obtained as below. If the absolute difference between the image intensities of the registered and reference images is less than 15 percent of the range of intensities of the zoomed image, then the anomaly matrix will get a value 0 at that location and a value of 1 elsewhere. Then, from these anomaly matrices, we create three anomaly images under these registration procedures. These images are presented in the third row of Figure 2.6.

**Discussion about symmetry of the proposed method:**

Next, we check whether the symmetry of the proposed solution is preserved or not. That means, if we interchange the role of reference image and zoomed image, or in other

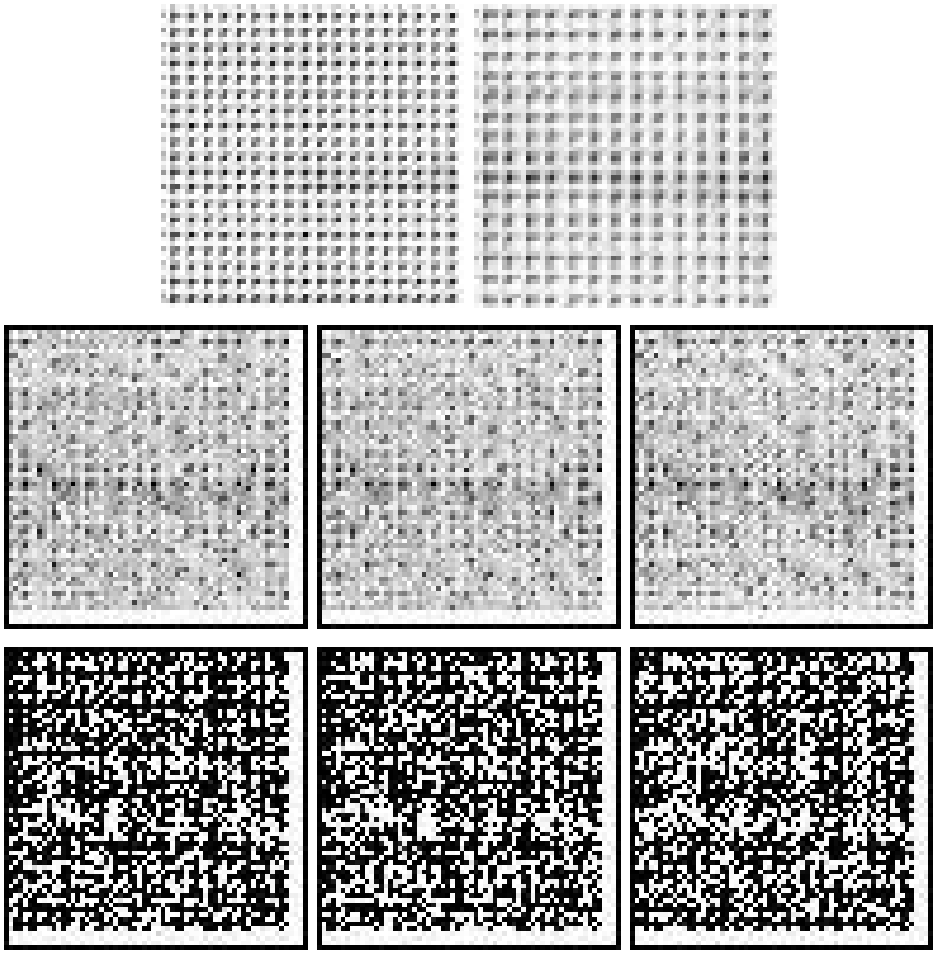


Figure 2.6: These figures are for the 'tile' image. Row 1 (from left to right): the reference image and the zoomed image; Row 2 (from left to right): registered images under $L_1$-norm, $L_2$-norm, and CC method, respectively; Row 3 (from left to right): anomaly images under $L_1$-norm, $L_2$-norm, and CC method, respectively.

words, if the moved image is zoomed-out rather than zoomed-in, whether our proposed solution is able to produce consistent results (See Christensen and Johnson (2001); Srivastava and Klassen (2016)). For that purpose, we consider the simulated image and its zoomed-in version shown in Figure 2.3 and interchange their roles in the registration algorithm, i.e., we treat the zoomed image as the reference image and the other image as the moved or zoomed-out image.

Next, we run the algorithm with $(r_1, r_2) = (2, 5)$ for $L_1$ and $L_2$ norms, and $(r_1, r_2) = (5, 5)$ for the CC-method, the optimal values in the first case, where we consider the zoomed-in image as the moved image. The MSD values for $L_1$-norm, $L_2$-norm, and the CC-method in this case are 0.1201, 0.1216, and 0.2030, respectively. We present all the images: zoomed (reference), target, registered images under $L_1$-norm, $L_2$-norm, and CC-method, and the anomaly images under these methods below:

From Figure 2.7, we see that the $L_1$-norm and the $L_2$-norm work satisfactorily except

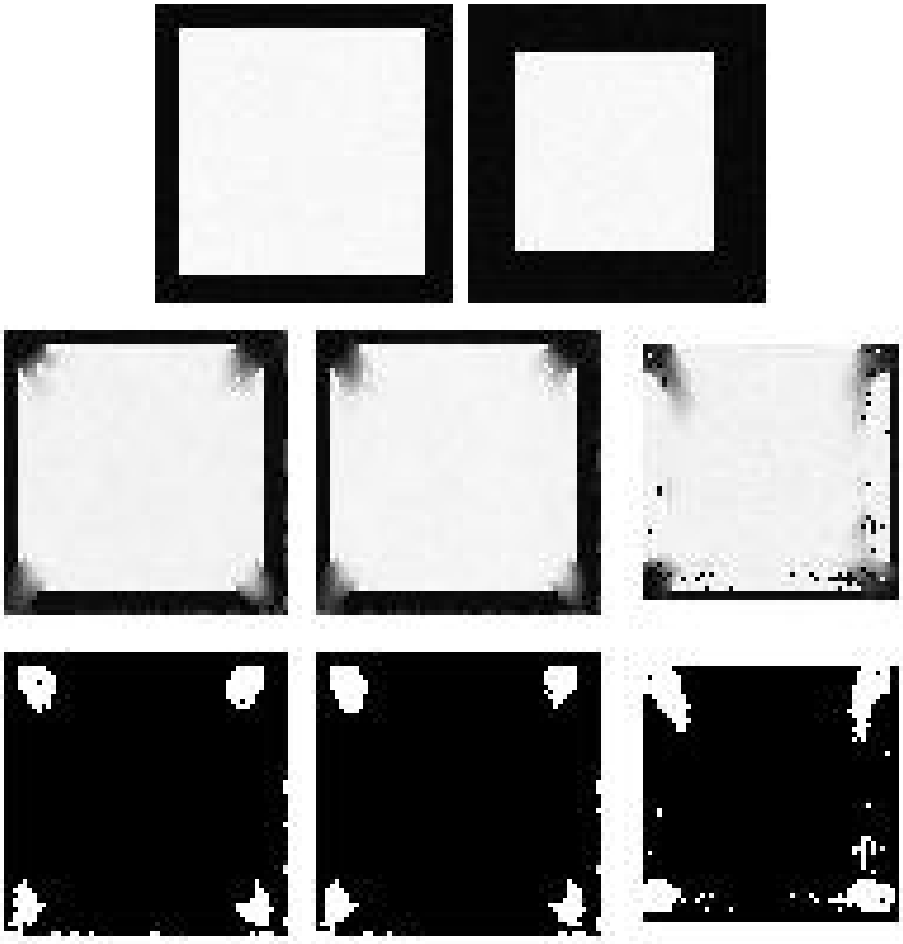


Figure 2.7: These figures are for the simulated image. Row 1 (from left to right): the reference image and the zoomed-out image; Row 2 (from left to right): registered images under $L_1$-norm, $L_2$-norm, and CC method, respectively; Row 3 (from left to right): anomaly images under $L_1$-norm, $L_2$-norm, and CC method, respectively.

for the corner regions. But the performance of the CC-method is rather poor.

## 2.5 Comparison with State-of-the-art Methods

In this section, we compare our method with a number of state-of-the-art image registration techniques. These methods are based on certain feature-matching algorithms. There are several popular feature matching algorithms in the literature, viz., Scale Invariant Feature Transform (SIFT) (Lowe, 2004), Speeded Up Robust Features (SURF) (Bay et al., 2008), Oriented Fast and Rotated Brief (ORB) (Rublee et al., 2011), Binary Robust Invariant Scalable Points (BRISK) (Leutenegger et al., 2011), KAZE (Alcantarilla et al., 2012), Accelerated-KAZE (AKAZE) (Alcantarilla and Solutions, 2011), to name a few. These methods work on the principle of detection and description of the features in the reference and transformed images. Then the relationship between these features from

reference and transformed images is found using some algorithm, e.g., RANSAC (Derpanis, 2010), and registered images are generated. In this chapter, we take ORB, AKAZE, and BRISK among these methods to compare with our best-performing algorithm, i.e., registration using $L_1$-norm. We first apply these methods to some images and create the registered images. Then we compute the MSE mutual information (MI) (Cover, 1999) between the reference image and the registered images obtained with respect to these methods and provide a table for comparison. Mean squared error (MSE), a similarity metric, is evaluated on the basis of the difference in gray level between two images. The expression for MSE is as follows:

$$\text{MSE}(I_R, \widehat{I}_M) = \sum_{(u,v)\in\Omega} (\widehat{I}_M(u,v) - I_R(u,v))^2,$$

where $\widehat{I}_M(u,v)$ and $I_R(u,v)$ represent the image intensity values at the $(u,v)$-th pixel coordinate for the registered and reference images, respectively. MI measures the mutual dependence between two images. Intuitively, it measures the reduced uncertainty of one image in the presence of the information of another image. Therefore, a larger value of MI is preferable for a greater degree of similarity. Mathematically, MI between reference and registered image can be expressed as

$$MI(I_R, \widehat{I}_M) = H(I_R) + H(\widehat{I}_M) - H(I_R, \widehat{I}_M),$$

where $H(I_R)$ and $H(\widehat{I}_M)$ are the information contained in the reference image and registered image, respectively. $H(I_R, \widehat{I}_M)$ is the joint entropy of $I_R$ and $\widehat{I}_R$. The normalized mutual information (NMI) is calculated as

$$NMI(I_R, \widehat{I}_M) = \frac{2MI(I_R, \widehat{I}_M)}{H(I_R) + H(\widehat{I}_M)}.$$

We also provide the reference image, the zoomed image, and the registered images side by side to facilitate comparison between these methods.

Since the proposed method does not involve learning from a large pool of data, comparing it with deep-learning methods, which are already pre-learned, is unfair. The performances of these deep-learning methods (Fu et al., 2020) depend very much on the set of images they were given to learn. For this reason, we do not consider any deep-learning based method for comparison purposes.

The registration of zooming transformation that we consider is a well-conditioned problem. However, the general non-linear diffeomorphic transformation is an ill-conditioned problem in the sense that there is a large number of non-linear transformations that

can map the moved image onto the reference image. The methods in the literature of diffeomorphic registration perform regularization, i.e., they impose a penalty on sharply varying deformation fields. The regularization steps of these methods, when applied to our problem, are not obvious from the context. The performance would depend very much on the choice of regularization. The proposed method does not require any such regularization, and comparison with these methods on a much simpler, well-conditioned problem would be rather out-of-context.

Here we consider four images, viz., a simulated image, a girl image, a pepper image, and a mountain image, for comparison purposes. The reason for taking the mountain image is that this image has very few features. As these methods (ORB, AKAZE, and BRISK) are all feature-based, we want to compare their performance with our intensity-based approach on the basis of this image as well. Next, we present the table containing MSE values computed for different methods.

| Mean Squared Error (MSE) | | | | |
|---|---|---|---|---|
| | $L_1$-norm | ORB | AKAZE | BRISK |
| simulated | **0.0827** | NA | NA | NA |
| girl | 0.0742 | 0.0087 | 0.0318 | **0.0049** |
| peppers | 0.0628 | 0.0053 | 0.0238 | **0.0037** |
| mountain | **0.0479** | 0.0925 | NA | NA |
| Normalized Mutual Information (NMI) | | | | |
| | $L_1$-norm | ORB | AKAZE | BRISK |
| simulated | **1.1335** | NA | NA | NA |
| girl | 1.1585 | 1.2348 | 1.0716 | **1.2633** |
| peppers | 1.1446 | 1.2935 | 1.1010 | **1.3282** |
| mountain | **1.2409** | 1.1201 | NA | NA |

Table 2.4: Mean squared error (MSE) and normalized mutual information (NMI) for different methods.

From MSE part of Table 2.4, we see that, in the case of a simulated image, the feature-based methods are failing miserably as the number of features in the simulated image is sparse. It is to be noted that the number of features detected by these competing methods is small. The feature-based methods (especially BRISK) outperform our proposed method in the case of the 'girl' and 'peppers' images because there are numerous features in these images. Again, because of the sparsity of the features according to their algorithms, the feature-based methods perform poorly in the case of the 'mountain' image. In this case, our proposed method is the clear winner. Note that the competing feature-based methods detect a small number of features according to their algorithms. From Figure 2.8, we see that, in the case of simulated images, the feature-based methods are unable to generate the registered images, whereas our proposed method performs reasonably. In the case of the 'girl' and 'peppers' images, the feature-based methods are clear winners. Finally, in the

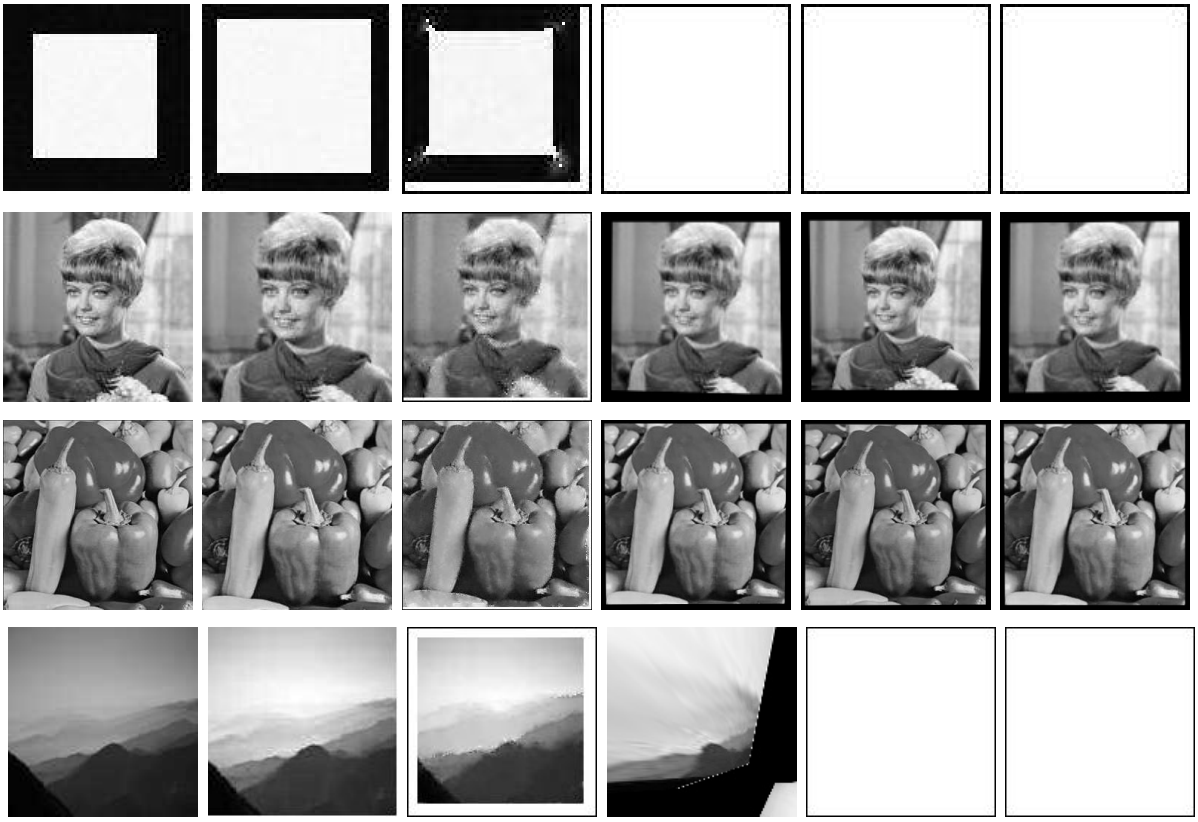


Figure 2.8: The reference image, the zoomed image, and the registered images under the proposed method, ORB, AKAZE and BRISK.

case of the 'mountain' image, AKAZE and BRISK are not able to generate the registered image due to a scarcity of features. On the other hand, the ORB method generates a distorted registered image that is not comparable to the reference image. In this case also, our proposed method also performs reasonably well and clearly outperforms some of the well-known feature-based methods. Similarly, from the NMI part of Table 2.4, we see mixed results. Our method clearly beats other methods in the case of the simulated image and the 'mountain' image. Though in the case of 'girl' image and 'pepper' image, the BRISK method beats our method significantly.

In addition to Table 2.4, we also provide the registered images generated by these methods in Figure 2.8. In each row of the following figure, we place the images in the following order: reference image, zoomed image, registered image for registration using $L_1$-norm, registered image for ORB, registered image for AKAZE, and registered image for BRISK. In the first row, we provide images corresponding to the simulated image, and in the following rows, we place the images for the girl, peppers, and mountain image, respectively.

## 2.6 Discussions

In this chapter, we propose an intensity-based method to register an image with its zoomed counterpart. This method works reasonably well in many of the cases described in this chapter. As the resolution of the image increases, the performance of this method also improves, as justified in Section 2.3. In Section 2.4, the numerical performance of the proposed method on three different images of various resolutions also supports the statement. There are, however, certain drawbacks to this method, which are discussed in the latter part of Section 2.4. In images where the number of features is small, our intensity-based method outperforms many state-of-the-art feature-based registration methods.

In the theoretical justification, the assumption $r_2/n \to 0$ states that the amount of zooming or the zooming factor is small. We prove the consistency of the registration procedure under this assumption. Dropping this assumption can certainly cover more general case of large zooming. In that scenario, stronger assumptions on the true image intensity function will be required. For example, if the true image intensity function is piecewise constant, then large $r_2$ can be used and the assumption $r_2/n \to 0$ will not be needed. The given proof can be easily extended to handle this situation. However, this new assumption is also restrictive in the sense that it does not hold true in many real life images. The given proof considers the situation when the zooming factor is small, but it works for a wide type of images.

Another shortcoming of this approach is that it is computationally heavy, and that is because this algorithm is being applied to each pixel location in the image. As a result of that, the run-time of the process is high, and if the zooming factor increases, then the neighborhood radius $r_2$ needs to be increased in order to find that point in the reference image with a similar neighborhood as the zoomed image. An increase in neighborhood radius increases the number of pixels in it, and neighborhood searching has to be done for each pixel location. This problem then makes this approach practically infeasible. One solution to this problem is to resort to feature-based approaches, where we run our algorithm on a few of the selected feature points and then, depending on the findings, frame a relationship that applies to the entire image.

# Chapter 3

# Feature-based Image Registration for Zooming Transformation[1]

## 3.1 Introduction

In the previous chapter, we propose an intensity-based image registration procedure, where one is a zoomed-in version of the other. As in the case of intensity-based methods, where we aim to map each pixel value to each other, the procedure becomes computationally expensive. In order to address this shortcoming of our proposed method, we aim to solve the same problem from a different point of view and resort to a feature-based approach. There are several feature-based approaches in the literature capable of handling the zooming transformation quite well. However, most of these methods perform miserably when the number of point features in the images is scarce. In real-life scenarios, there are lots of situations where point features are not readily available. See Figure 3.1 as an example. We also wish to address this problem in this chapter and select a feature that is readily available in almost all non-trivial images.

In this chapter, we propose a novel and effective non-point feature-based image registration method in the presence of a zooming transformation. We consider a non-point feature, viz., jump points or edge points, as the main feature of our proposed method. Note that this feature is always available as long as the image is non-trivial. One major advantage of the proposed method is that, along with the mapping, it is capable of estimating the zooming center and the scaling parameter. We first detect the edge points in both the reference and the zoomed image. As the numbers of edge points for the reference

[1]This chapter is based on the publication Das et al. (2024): **Das, S.**, Roy, A. and Mukherjee, P.S. "Image registration for zooming: A statistically consistent local feature mapping approach", *Stat*, 2024, 13(1), e664. DOI: 10.1002/sta4.664.

Figure 3.1: Mountain image: (a) reference (left), (b) zoomed-in (right).

and the zoomed images are not the same, we use a feature mapping algorithm to map them, and then we use a mathematical model to explain the relationship between the edge points of the reference and the zoomed image. Finally, we generate the registered image using the aforementioned relationship. This method works well in many situations and performs exceptionally well in comparison with the methods discussed in the previous paragraph when there are not enough point features in the image. We get a glimpse of that in Figure 3.2. Among the methods ORB, AKAZE, and BRISK used for comparison, ORB generates a distorted image, whereas the other two methods are not able to create a registered image.

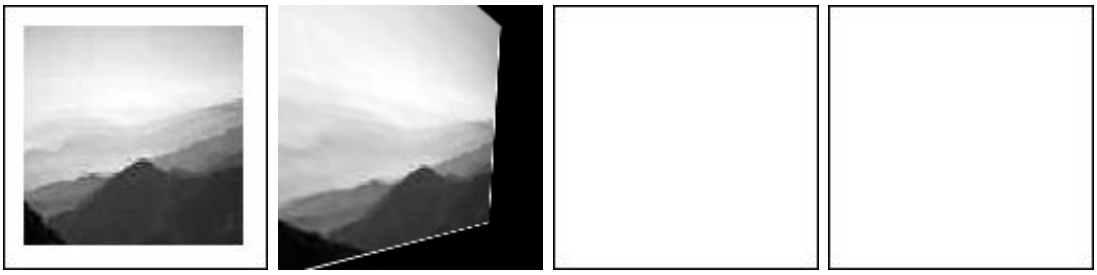


Figure 3.2: Registered images for the 'mountain' image. From left to right: (a) registered image under the
proposed method, (b) registered image under ORB, (c) registered image under AKAZE, (d) registered image under BRISK.

The rest of the chapter is organized as follows. In Section 3.2, we describe our proposed image registration procedure. It is then followed by some statistical properties of the method in Section 3.3. We present several numerical examples along with comparisons with some state-of-the-art methods in Section 3.4 and 3.5. We conclude this chapter by providing relevant discussions in Section 3.6.

## 3.2 Proposed Image Registration Procedure

Our proposed method is described in the following steps. To begin with, we describe the edge detection method that we consider in Section 3.2.1. Then, we state the underlying assumptions for our procedure in Section 3.2.2. Section 3.2.3 introduces the statistical model for describing the image registration procedure. Section 3.2.4 describes the first pivotal step for this imaging technique, which is feature (edge point) selection for both reference and zoomed images. Then, we describe the algorithm for mapping the edge pixels of the zoomed image to the edge pixels of the reference image in Section 3.2.5. Section 3.2.6 explains the process of parameter selection for the procedure. We conclude this section with a description of the estimation process for the zooming parameters in Section 3.2.7.

### 3.2.1 Edge Detection by Local Kernel Smoothing

As discussed in the previous section, a key step in our proposed image registration procedure is the detection of jump points (edges) using a reasonable edge detector. Theoretically speaking, any suitable edge detector can be used for this purpose. In this chapter, we use an edge detection method based on a local linear kernel estimator. Under the jump regression analysis (JRA) framework (Qiu, 2005), a 2-D gray-scale image can be expressed by the following regression model:

$$I_f(x_i, y_j) = f(x_i, y_j) + \varepsilon_f(x_i, y_j) \quad \text{for} \quad i, j = 1, 2, ....n, \tag{3.2.1}$$

where $\{(x_i, y_j) : i, j = 1, 2, ...., n\}$ are equally spaced pixel coordinates in the design space $\Omega = [0, 1] \times [0, 1]$, $f$ is the unknown image intensity function, $N = n^2$ is the sample size, and $\varepsilon_f$ are independent and identically distributed (i.i.d.) random errors with mean 0 and variance $\sigma^2 > 0$. Now, to detect the edge pixels, we consider a circular neighborhood at each pixel $(x, y)$ as $O(x, y; h_n) = \{(u, v) : \sqrt{(u - x)^2 + (v - y)^2} \leq h_n\}$, where $h_n > 0$ is a bandwidth parameter. Then, the LLK (local linear kernel) smoothing procedure is accomplished by minimizing the following objective function:

$$\min_{a,b,c} \sum_{(x_i,y_j)\in O(x,y;h_n)} \{I_f(x_i, y_j) - \alpha - \beta_1(x_i - x) - \beta_2(y_j - y)\}^2 K_{ij}, \tag{3.2.2}$$

where $K_{ij} = K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n}\right)$ is a radially symmetric bivariate density kernel with support $\{(x, y) : x^2 + y^2 \leq 1\}$. The estimate of $(\alpha, \beta_1, \beta_2)$ from the above minimization problem (3.2.2) is denoted as $\left(\widehat{\alpha}(x, y), \widehat{\beta}_1(x, y), \widehat{\beta}_2(x, y)\right)$. When there is no jump, $\widehat{\alpha}$ is the LLK

estimate of $f(x, y)$, and $(\widehat{\beta}_1, \widehat{\beta}_2)'$ is the LLK estimator of the gradient vector. Now, along the direction perpendicular to the estimated gradient vector, we divide $O(x, y; h_n)$ into two halves, denoted as $O_1(x, y)$ and $O_2(x, y)$. We calculate Nadaraya–Watson local constant kernel estimators by solving similar optimization problems on both regions $O_1$ and $O_2$ and denote them as $\widehat{f}_{LCK,O_1}$ and $\widehat{f}_{LLK,O_2}$, respectively. Then, a design point $(x, y)$ is flagged as a detected jump point if $|\widehat{f}_{LCK,O_1}(x, y) - \widehat{f}_{LCK,O_2}(x, y)|$ is large. Readers are referred to Kang and Qiu (2014); Qiu (2004) for more details about the jump detection procedure.

### 3.2.2 Assumptions

In this chapter, we consider a non-rigid transformation, viz., zooming of the image object. Translation and rotation of the image are not considered here. Next, we assume that the true intensity functions of the 2-D grayscale images are piecewise constants. Additionally, we assume that the image resolutions, i.e., the image dimensions of both the reference and the zoomed images, are the same. Furthermore, we choose the zooming factor and the center of zooming in such a manner that the region of interest of the reference image is inside the zoomed image.

### 3.2.3 Model Description

We have two images of the same object or scene to register. One is the actual image; we call it the reference image and denote it by R, and the other is the zoomed version of the reference image, which we denote by Z. We aim to register these two images, i.e., we wish to find a one-to-one mapping between these two images, or, in other words, we want to formulate a mathematical function such that once we apply the function to the zoomed image, it will return the reference image. This problem can be more precisely described by the following statistical model. Let R and Z be two images of resolution $n \times n$ to register. Their true image intensity functions are denoted as $R(x, y)$ and $Z(x, y)$. It is assumed that they have the following relationship:

$$Z(T_1(x, y), T_2(x, y)) = R(x, y), \text{ for } (x, y) \in \Omega, \tag{3.2.3}$$

where $\Omega = [0, 1] \times [0, 1]$ is the design space of image R, and $T(x, y) = (T_1(x, y), T_2(x, y))$ is the zooming transformation to estimate. This method tries to estimate $T(x, y)$ from

the observed image intensities of the two images defined below:

$$\begin{aligned} I_R(x_i, y_j) &= R(x_i, y_j) + \varepsilon_R(x_i, y_j) \\ I_Z(x_i, y_j) &= Z(x_i, y_j) + \varepsilon_Z(x_i, y_j) \\ &\text{for } \; i, j = 1, 2, ....n, \end{aligned} \tag{3.2.4}$$

where $\{(x_i, y_j) : i, j = 1, 2, ...., n\}$ are equally spaced pixel locations and $\varepsilon_R(x_i, y_j)$ and $\varepsilon_Z(x_i, y_j)$ are i.i.d. random errors in these images with mean 0 and unknown variances $\sigma_R^2$ and $\sigma_Z^2$, respectively.

### 3.2.4 Feature Selection

Next, we consider the most popular feature of an image, i.e., the edge, for registration purposes. So, we detect the edge pixels of reference and zoomed images according to the method described in Section 3.2.1. Let $\{(u_i, v_i) : i = 1, 2, ..., n_1\}$ be the detected edge pixels for the reference image and $\{(U_i, V_i) : i = 1, 2, ..., n_2\}$ be the detected edge pixels for the zoomed image, where $n_2 > n_1$. Now, we aim to find a relationship between these edge pixels of the reference and the zoomed image to create the registered version of the zoomed image. A suitable model that captures the relationship between the detected edge pixels of the reference and the zoomed image is stated below:

$$\begin{aligned} U_i &= s(u_i - \ell_x) + \ell_x, \\ V_i &= s(v_i - \ell_y) + \ell_y, \end{aligned} \tag{3.2.5}$$

where $\{(u_i, v_i) : i = 1, 2, \ldots, N\}$ are N edge pixels of the reference image that has been mapped to $\{(U_i, V_i) : i = 1, 2, ..., N\}$ of the zoomed image, $(\ell_x, \ell_y)$ is the zooming center (the focal point of the camera lens while zooming), and $s$ is the zooming factor (the ratio of the distance of a point from the zooming center in the zoomed image to the distance of the corresponding point from the zooming center in the reference image). However, we cannot directly use this model in our context because the numbers of edge pixels in the reference and the zoomed image are not the same. Moreover, we do not know which $(u_i, v_i)$ of the reference image has been shifted to which $(U_i, V_i)$ of the zoomed image. We solve this problem through feature mapping of the edge pixels, which we discuss in the next subsection.

### 3.2.5 Feature Mapping

We propose the following steps for performing feature mapping for zooming:

**Step 1:** For a given edge pixel $(U, V)$ in the zoomed image, we consider a circular neighborhood of radius $r_1$ around that pixel, denoted as $O(U, V; r_1)$.

**Step 2:** Then, in the reference image, around the same coordinates as that given pixel, we consider a circular neighborhood of radius $r_2$, denoted as $O(U, V; r_2)$, where $r_2 \geq r_1$. For any pixel $(u, v)$ in that neighborhood $O(U, V; r_2)$, we consider a circular neighborhood of radius $r_1$ around $(u, v)$, denoted as $O(u, v; r_1)$. See Figure 3.3 for a visual demonstration.

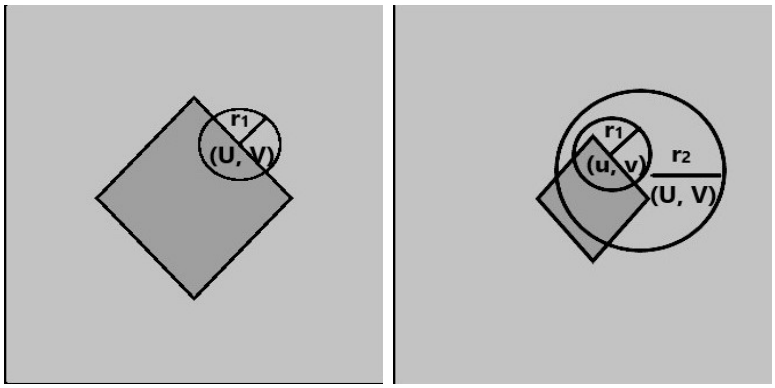



Figure 3.3: A demonstration of feature mapping algorithm.

**Step 3:** Next, we compare the two neighborhoods, i.e., $O(U, V; r_1)$ of the zoomed image, and all possible choices of $O(u, v; r_1)$ corresponding to the reference image on the basis of two measures: mean square difference (MSD) and mean absolute difference (MAD), and choose the neighborhood $O(u, v; r_1)$ of the reference image that is most similar to the neighborhood $O(U, V; r_1)$ of the zoomed image with respect to these two measures. Subsequently, we map the pixel $(U, V)$ of the zoomed image to that $(u, v)$ of the reference image. These two measures are defined below:

$$MSD((u, v); (U, V)) = \frac{1}{N} \sum_{s,t} (I_R(u + s, v + t) - I_Z(U + s, V + t))^2, \tag{3.2.6}$$

$$MAD((u, v); (U, V)) = \frac{1}{N} \sum_{s,t} |I_R(u + s, v + t) - I_Z(U + s, V + t)|, \tag{3.2.7}$$

where all the above summations are taken over the set $\{(s, t) : (u+s, v+t) \in O(u, v; r_1)\}$, N is the number of pixels in $O(u, v; r_1)$, $I_{(.)}(a, b)$ denotes the intensity of the image pixel (a,b), R denotes the reference image, and Z denotes the zoomed image. Therefore, the transformed or mapped pixels are as follows:

$$\underset{(u,v) \in O(U,V;r_2)}{arg\,min} \text{MSD}((u, v); (U, V)), \tag{3.2.8}$$

$$\underset{(u,V) \in O(U,V;r_2)}{arg\,min} \text{MAD}((u, v); (U, V)), \tag{3.2.9}$$

corresponding to the measures MSD and MAD, respectively. There are two tuning parameters in this procedure, which are the radii $r_1$ and $r_2$ of the neighborhoods. The performance of this procedure depends highly on the selection of these parameters. We address the selection process for these parameters in the next section.

### 3.2.6 Selection of Procedure Parameters

In the proposed image registration procedure, there are primarily three parameters to choose from: the bandwidth parameter $h_n$ for detecting edge pixels and the radii of the two circular neighborhoods $r_1$ and $r_2$ associated with the feature mapping step. In this subsection, we propose a practical guideline for selecting the procedure parameters.

**Choosing $h_n$:** In this chapter, we consider choosing $h_n$ in an ad-hoc manner, as the registration performance is not very sensitive to the choice of this parameter. Based on our numerical experience, we suggest choosing $h_n \in [\frac{2}{n}, \frac{3}{n}]$. In the simulation study in Section 3.4, we choose $h_n = \frac{2}{n}$.

**Choosing $r_1$ and $r_2$:** In the above-mentioned feature mapping method, $r_1$ is the radius of the neighborhood $O(U, V; r_1)$ corresponding to the edge pixel $(U, V)$ of the zoomed image, and $O(U, V; r_2)$ is the neighborhood of radius $r_2$ corresponding to the same pixel of the reference image. Here, we find the mapping between $(u, v)$ and $(U, V)$ for various possible choices of $r_1$ and $r_2$ and create the registered version of the reference image by following the steps described in Sections 2.5 and 2.6. We can measure the similarity between the reference image and its registered version through their mean square error (MSE). This can be expressed as,

$$MSE(\widetilde{r}_1, \widetilde{r}_2) = \sum_{(u,v)\in R} (\widehat{I}_R(u, v, \widetilde{r}_1, \widetilde{r}_2) - I_R(u, v))^2,$$

where $\widehat{I}_R(u, v, \widetilde{r}_1, \widetilde{r}_2)$ is the intensity of the pixel $(u, v)$ of registered version $(\widehat{I}_R)$ of the reference image R and $\widetilde{r}_1$ and $\widetilde{r}_2$ are the parameters for feature mapping. Then, we select those values of $r_1$ and $r_2$ for which $MSE(\widetilde{r}_1, \widetilde{r}_2)$ is minimum.

### 3.2.7 Estimation of the Zooming Parameters

In this step, we find a one-to-one mapping of the edge pixels of the reference and the zoomed image using the above-mentioned feature mapping method. In this chapter, we start with the edge pixels of the zoomed image and find their counterparts in the reference

image through feature mapping. Then, we fit the model mentioned in eqn.(3.2.5) to those data points (the edge pixels of the reference and the zoomed image).

Note that $(u_i, v_i)$, for $i = 1, 2, ..., n$, are the edge pixels of the reference image, and $(U_i, V_i)$, for $i = 1, 2, ..., N$, are the edge pixels of the zoomed image. Once we apply the feature mapping algorithm to $(U_i, V_i)$, we get $(u_i', v_i')$, for $i = 1, 2, ..., N$, a set of mapped edge pixels of the reference image. Next, we use classical linear regression to estimate the parameters of the model: for $i = 1, 2, \ldots N$,

$$U_i = su_i' + \ell_x^* + \zeta_{U_i}, \quad V_i = sv_i' + \ell_y^* + \zeta_{V_i}, \tag{3.2.10}$$

where $\ell_x^* = \ell_x(1-s)$ and $\ell_y^* = \ell_y(1-s)$. $\zeta_{U_i}$ and $\zeta_{V_i}$ are error terms associated with linear regression. Here, the estimates of $s$, $\ell_x^*$ and $\ell_y^*$ can be obtained by using the following relationship:

$$\begin{aligned} \widehat{s} &= \frac{\sum_{i=1}^{N}(u_i' - \bar{u'})(U_i - \bar{U})}{\sum_{i=1}^{N}(u_i' - \bar{u'})^2}, \\ \widehat{\ell_x^*} &= \bar{U} - \widehat{s}\bar{u'}. \end{aligned} \tag{3.2.11}$$

Similarly, from the second equation of (3.2.10), we can find the estimates $\widehat{s}$ and $\widehat{\ell_y^*}$ of $s$ and $\ell_y^*$, respectively. The estimates of $s$ found from both equations will be close. We choose any one of them. Once we find the estimates of $s$, $\ell_x^*$, and $\ell_y^*$, then using the relationship between $\ell_x^*$, $\ell_x$, and $s$, and the relationship between $\ell_y^*$, $\ell_y$, and $s$, we can find the estimates of $\ell_x$ and $\ell_y$. Once this step is complete, we use model (3.2.5) to create the registered version of the reference image.

### 3.2.8 Steps summarizing the proposed registration algorithm

**Step 1:** Detect the edge pixels, i.e., the features of the reference image and the zoomed image by following the method mentioned in Section 3.2.1. For each edge pixel, execute Steps 2 and 3.

**Step 2:** Follow the feature mapping algorithm as mentioned in Section 3.2.5 to find the closest match of an edge pixel of the zoomed image in the reference image.

**Step 3:** Next, estimate the zooming parameters by the process described in Section 3.2.7.

**Step 4:** Finally, create the registered image by using the estimated parameters and Equation 3.2.5.

## 3.3 Statistical Properties

In this section, we discuss some statistical properties regarding the proposed image registration procedure. The proposed method has two major steps, which we justify theoretically. We have the following two propositions.

**Proposition 3.3.1.** *Let $(u, v)$ be an edge pixel of an image of resolution $n \times n$, and there is a zoomed version of the same image. In the zoomed image, the pixel $(u, v)$ has moved to $(U, V)$, where $U = u + c_1$ and $V = v + c_2$, $(c_1, c_2) \in \mathbb{R}^2$. When we apply our feature mapping procedure with window widths $r_1$ and $r_2$ to register these two images, $(U, V)$ in the zoomed image is being mapped to $(u', v')$ in the registered image. If $n \to \infty$ and $r_2 \to \infty$ such that $\frac{r_2}{n} \to 0$,*

$$\frac{1}{n^2}[(u' - u)^2 + (v' - v)^2] \xrightarrow{P} 0.$$

**Proof of Proposition 3.3.1:** We provide a proof of this proposition in Appendix A.2.1.

Note that the above proposition establishes the consistency of the feature mapping procedure mentioned in Section 3.2.5. Our next proposition is based on zooming parameter estimation.

**Proposition 3.3.2.** *Consider the model (3.2.10), and under the assumption that $\zeta_{U_i}$ and $\zeta_{V_i}$ are from distributions with mean 0 and finite variances, $\widehat{s} \xrightarrow{P} s$, $\widehat{\ell}_x \xrightarrow{P} \ell_x$ and $\widehat{\ell}_y \xrightarrow{P} \ell_y$ as $N \to \infty$.*

**Proof of Proposition 3.3.2:** We provide a proof of this proposition in Appendix A.2.2.

The above proposition establishes the consistency of the estimates of the zooming factor and the zooming center.

## 3.4 Numerical Studies

In this section, we investigate the numerical performance of the proposed registration method. We apply our method to various applications, both simulated and real. We start with the zoomed image and create the registered version of the reference image by following the procedure described in Section 3.2. We consider a simulated image of resolution $64 \times 64$ to begin with, and subsequently, we apply our method to the 'lake' and 'brain' images of resolution $128 \times 128$.

At first, we simulate a reference image of resolution $64 \times 64$ and its zoomed version, which is also of resolution $64 \times 64$. In the reference image, all pixels except the central

circular region of radius 15/64 have intensity 1, and all other pixels have intensity 0, and in the zoomed image, the central circular region, having intensity 1 at all pixel locations, is of radius 20. All other pixels, except those in the central region, have an intensity 0. Then, to complete the construction of the simulated images, we add a Gaussian noise of mean 0 and standard deviation of 0.01 to each pixel of the reference and the zoomed image. These images are placed side by side in Figure 3.4.

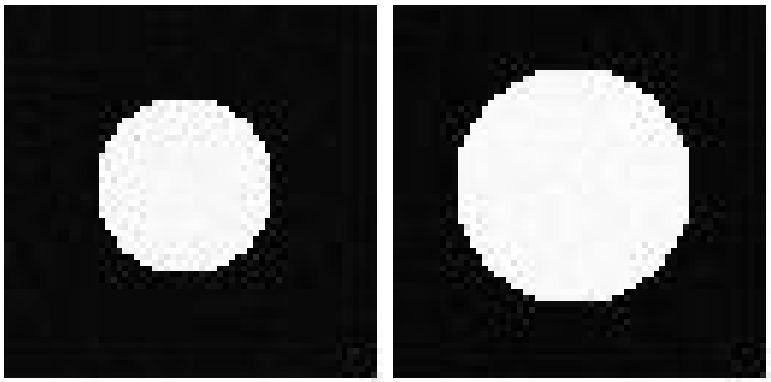


Figure 3.4: Simulated example: (a) reference image (left), (b) zoomed image (right).

Next, we detect the edge pixels of the reference and the zoomed image using the procedure mentioned in Section 3.2.1, and then follow the feature mapping procedure as discussed in Section 3.2.5. To select the parameters $r_1$ and $r_2$ for feature mapping, we implement the steps mentioned in Section 3.2.7. In this context, we consider 9 values of $r_2$, from 4 to 12, and for each choice of $r_2$, we consider $r_1$ to be from 1 to that value of $r_2$. For each such choice of $r_1$ and $r_2$, we run the procedure and create the registered version of the reference images. We repeat the procedure 10 times for 10 different sets of noisy images. We then compute the mean and standard deviation of the Mean Square Error (MSE) values (mentioned in Section 3.2.7) for those 10 replications corresponding to two similarity measures, MAD and MSD, and note them against each specific choice of $r_1$ and $r_2$. We provide a partial outcome of the above procedure in Table 3.1.

From Table 3.1, we see that the average MSE is minimum for the registration procedure using $L_1$-norm if $(r_1, r_2)$ is chosen to be $(10, 10)$ and the corresponding value of $(r_1, r_2)$ is $(7, 7)$ while using $L_2$-norm. Therefore, using these values of $r_1$ and $r_2$, we run the registration procedures and create the registered images. During the process of image registration, we also estimate the zooming center $(\ell_x, \ell_y)$ and zooming factor $s$. Those estimates are tabulated in Table 3.2.

Next, we create the residual images for the registration procedure involving $L_1$-norm and $L_2$-norm, respectively. Since the target image is the reference image, the intensity of any given pixel of the residual image will be the difference in intensities at the same image pixel (as that of the residual image) of the reference image and the registered version of

| Mean (standard deviation) | | |
|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm |
| (6 , 7) | 0.0142 (0.0047) | 0.0119 (0.0059) |
| (7 , 7) | 0.0124 (0.0078) | **0.0091 (0.0065)** |
| (1 , 8) | 0.0691 (0.0205) | 0.0666 (0.0141) |
| (2 , 8) | 0.0230 (0.0136) | 0.0250 (0.0151) |
| (3 , 8) | 0.0180 (0.0131) | 0.0157 (0.0116) |
| (4 , 8) | 0.0135 (0.0031) | 0.0150 (0.0063) |
| (5 , 8) | 0.0175 (0.0003) | 0.0175 (0.0003) |
| (6 , 8) | 0.0139 (0.0060) | 0.0152 (0.0071) |
| (7 , 8) | 0.0104 (0.0077) | 0.0133 (0.0078) |
| (8 , 8) | 0.0099 (0.0028) | 0.0110 (0.0034) |
| (1 , 9) | 0.0851 (0.0184) | 0.0847 (0.0245) |
| (2 , 9) | 0.0328 (0.0234) | 0.0366 (0.0265) |
| (3 , 9) | 0.0216 (0.0111) | 0.0212 (0.0106) |
| (4 , 9) | 0.0156 (0.0039) | 0.0191 (0.0095) |
| (5 , 9) | 0.0189 (0.0001) | 0.0189 (0.0001) |
| (6 , 9) | 0.0165 (0.0057) | 0.0125 (0.0072) |
| (7 , 9) | 0.0127 (0.0086) | 0.0157 (0.0098) |
| (8 , 9) | 0.0103 (0.0030) | 0.0118 (0.0028) |
| (9 , 9) | 0.0214 (0.0025) | 0.0209 (0.0050) |
| (1 , 10) | 0.1105 (0.0214) | 0.1155 (0.0252) |
| (2 , 10) | 0.0332 (0.0174) | 0.0341 (0.0172) |
| (3 , 10) | 0.0201 (0.0099) | 0.0237 (0.0130) |
| (4 , 10) | 0.0192 (0.0095) | 0.0173 (0.0109) |
| (5 , 10) | 0.0205 (0.0001) | 0.0205 (0.0001) |
| (6 , 10) | 0.0206 (0.0061) | 0.0176 (0.0081) |
| (7 , 10) | 0.0172 (0.0092) | 0.0130 (0.0087) |
| (8 , 10) | 0.0118 (0.0033) | 0.0112 (0.0027) |
| (9 , 10) | 0.0202 (0.0072) | 0.0186 (0.0073) |
| (10 , 10) | **0.0073 (0.0006)** | 0.0120 (0.0095) |
| (1 , 11) | 0.1238 (0.0221) | 0.1342 (0.0281) |

Table 3.1: Mean and standard deviation of MSE values for the simulated image.

| | $(\widehat{\ell_x}, \widehat{\ell_y})$ | $\widehat{s}$ | $(\ell_x, \ell_y)$ | $s$ |
|---|---|---|---|---|
| $L_1$-norm | $(31.4/64, 31/64)$ | 1.35 | $(0.5, 0.5)$ | 1.33 |
| $L_2$-norm | $(32.8/64, 32.8/64)$ | 1.33 | $(0.5, 0.5)$ | 1.33 |

Table 3.2: Estimates and the true values of the zooming center and the zooming factor for the simulated image.

the reference image. We show these images (registered and residual) side by side in Figure 3.5. The images are in the following order: registered image using $L_1$-norm, registered image using $L_2$-norm, residual image using $L_1$-norm, and residual image using $L_2$-norm.

In addition to the simulated image, we apply our procedure to the 'lake' image of resolution $128 \times 128$. Then we select its central region of resolution $112 \times 112$ and zoom

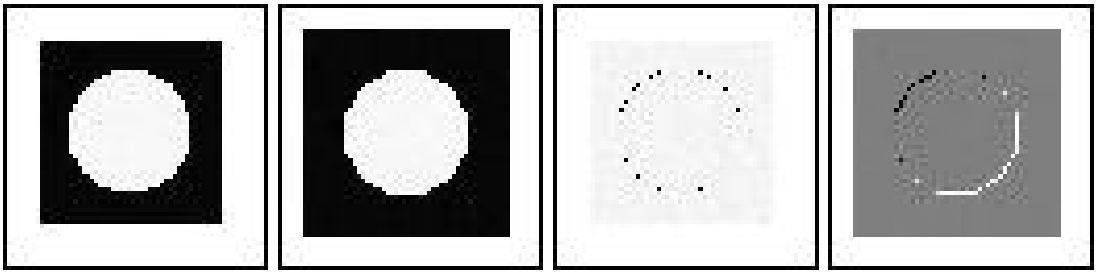


Figure 3.5: From left to right: (a) the registered image using $L_1$-norm, (b) the registered image using $L_2$-norm, (c) the residual image using $L_1$-norm, (d) the residual image using $L_2$-norm.

it while keeping the resolution the same as the original one. We present both images, reference and zoomed, in Figure 3.6.

Figure 3.6: The 'lake' image: (a) reference (left), (b) zoomed (right).

Next, we add Gaussian noise of a mean 0 and a standard deviation of 0.01 to each pixel of the reference and the zoomed image. Adding such noise is not required for our procedures to run, though. We then detect the edges of the reference and the zoomed image and perform the feature mapping. To select the parameters $r_1$ and $r_2$ for feature mapping, we implement the steps mentioned in Section 3.2.7. In this context, we consider 8 different values of $r_2$, from 5 to 12, and for each choice of $r_2$, we consider the values of $r_1$ to be from 1 to that value of $r_2$. We follow the same procedure as mentioned for the simulated image and compute the mean and standard deviation of the Mean Squared Error (MSE) values corresponding to the two similarity measures MAD and MSD, and note them against each specific choice of $r_1$ and $r_2$. We provide a partial outcome of the above procedure in Table 3.3.

We see from Table 3.3 that the average MSE is minimum for the registration procedure using $L_1$-norm if $(r_1, r_2)$ is chosen to be $(6, 11)$ and minimum is obtained for the registration procedure using $L_2$-norm if $(r_1, r_2)$ is chosen to be $(3, 12)$. So, using the values of $r_1$ and $r_2$ for the respective procedures, we run the registration procedure and create the registered version of the reference image for the respective procedures. We also obtain

| Mean (standard deviation) | | |
|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm |
| (5, 11) | 0.0086 (0.0008) | 0.0268 (0.0030) |
| (6, 11) | **0.0034 (0.0033)** | 0.0276 (0.0023) |
| (7, 11) | 0.0061 (0.0005) | 0.0236 (0.0007) |
| (8, 11) | 0.0072 (0.0003) | 0.0279 (0.0002) |
| (9, 11) | 0.0054 (0.0004) | 0.0294 (0.0003) |
| (10, 11) | 0.0061 (0.0003) | 0.0314 (0.0007) |
| (11, 11) | 0.0048 (0.0003) | 0.0324 (0.0003) |
| (1, 12) | 0.0246 (0.0002) | 0.0251 (0.0025) |
| (2, 12) | 0.0175 (0.0045) | 0.0204 (0.0033) |
| (3, 12) | 0.0174 (0.0008) | **0.0201 (0.0008)** |
| (4, 12) | 0.0110 (0.0024) | 0.0255 (0.0026) |

Table 3.3: Mean and standard deviation of MSE values for the 'lake' image.

the estimate of the zooming center $(\ell_x, \ell_y)$ and the zooming factor $s$ during this process of image registration. Those estimates are tabulated in Table 3.4.

| | $(\widehat{\ell_x}, \widehat{\ell_y})$ | $\widehat{s}$ | $(\ell_x, \ell_y)$ | $s$ |
|---|---|---|---|---|
| $L_1$-norm | $(66.5/128, 66/128)$ | 1.14 | $(0.5, 0.5)$ | 1.14 |
| $L_2$-norm | $(68.8/128, 83/128)$ | 1.10 | $(0.5, 0.5)$ | 1.14 |

Table 3.4: Estimates and the true values of the zooming center and the zooming factor for the 'lake' image.

Next, we create the residual images for the registration procedure using $L_1$-norm and $L_2$-norm, respectively. These images (registered and residual) are presented side by side in Figure 3.7. The images are in the following order: registered image using $L_1$-norm, registered image using $L_2$-norm, residual image using $L_1$-norm, and residual image using $L_2$-norm.

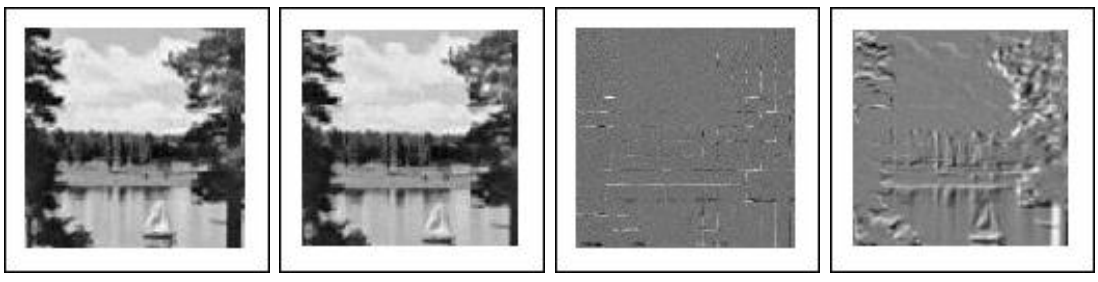


Figure 3.7: From left to right: (a) the registered image using $L_1$-norm, (b) registered image using $L_2$-norm, (c) residual image using $L_1$-norm, (d) residual image using $L_2$-norm.

Next, we apply our procedure to the 'brain' image of resolution $128 \times 128$. We select its central region of resolution $112 \times 112$ and zoom it while keeping the resolution the

same as the original one. We see those images, both original (reference) and zoomed in Figure 3.8.

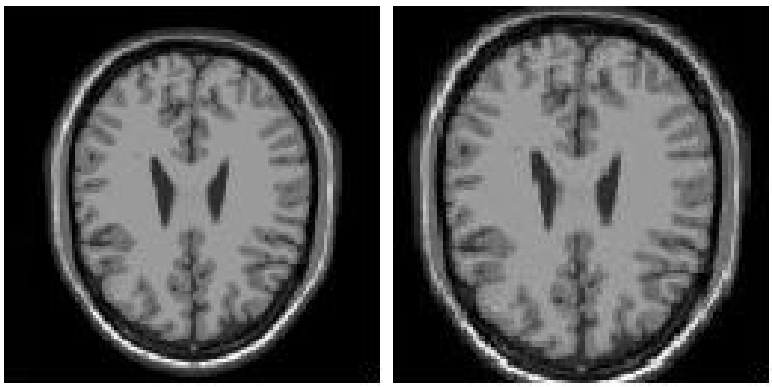


Figure 3.8: The 'brain' image: (a) reference (left), (b) zoomed (right).

As before, we add a Gaussian noise of mean 0 and standard deviation of 0.01 to each pixel of the reference and the zoomed image and detect the edges of the reference and the zoomed image. We then perform the feature mapping. We select the parameters $r_1$ and $r_2$ for the feature mapping by implementing the steps mentioned in Section 3.2.7. In this context, we consider 9 different values of $r_2$, from 4 to 12, and for each choice of $r_2$, we consider $r_1$ to be from 1 to that value of $r_2$. We then follow the same procedure as mentioned earlier (for the simulation and the 'lake' image) and compute the mean and standard deviation of the Mean Squared Error (MSE) values corresponding to two similarity measures, MAD and MSD, and note them against each specific choice of $r_1$ and $r_2$. We provide a partial outcome of the above procedure in Table 3.5.

From Table 3.5, we see that the average MSE is minimum for the registration procedure using $L_1$-norm if $(r_1, r_2)$ is chosen to be $(9, 11)$ and the minimum is obtained while using $L_2$-norm if $(r_1, r_2)$ is chosen to be $(8, 9)$. Therefore, using these values of $r_1$ and $r_2$ for the respective procedures, we run the registration method and create the registered images. Along with the registered images, we also obtain the estimates of the zooming center $(\ell_x, \ell_y)$ and the zooming factor $s$. Those estimates are provided in Table 3.6.

Following the same pattern as earlier, we create the residual images for the registration procedure involving $L_1$-norm and $L_2$-norm, respectively. These images (registered and residual) are placed side by side in Figure 3.9. The images are in the following order: registered image using $L_1$-norm, registered image using $L_2$-norm, residual image using $L_1$-norm, and residual image using $L_2$-norm.

Furthermore, we check the performance of the proposed method on two subsequent images, viz., a 'mountain' image and a satellite image of 'Aral Sea' of resolutions of $128 \times 128$ each. Similar to previous examples, we consider the central region of resolution

| Mean (standard deviation) | | |
|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm |
| (7 ,9) | 0.0144 (0.0034) | 0.0150 (0.0038) |
| (8 ,9) | 0.0131 (0.0002) | **0.0129 (0.0005)** |
| (9 ,9) | 0.0131 (0.0009) | 0.0147 (0.0036) |
| (1 ,10) | 0.0256 (0.0005) | 0.0256 (0.0004) |
| (2 ,10) | 0.0192 (0.0004) | 0.0203 (0.0034) |
| (3 ,10) | 0.0198 (0.0003) | 0.0195 (0.0004) |
| (4 ,10) | 0.0184 (0.0033) | 0.0202 (0.0005) |
| (5 ,10) | 0.0133 (0.0026) | 0.0174 (0.0040) |
| (6 ,10) | 0.0148 (0.0042) | 0.0200 (0.0027) |
| (7 ,10) | 0.0143 (0.0035) | 0.0168 (0.0044) |
| (8 ,10) | 0.0129 (0.0005) | 0.0134 (0.0027) |
| (9 ,10) | 0.0133 (0.0025) | 0.0150 (0.0038) |
| (10 ,10) | 0.0132 (0.0027) | 0.0191 (0.0036) |
| (1 ,11) | 0.0290 (0.0036) | 0.0288 (0.0037) |
| (2 ,11) | 0.0210 (0.0038) | 0.0242 (0.0036) |
| (3 ,11) | 0.0195 (0.0005) | 0.0192 (0.0004) |
| (4 ,11) | 0.0199 (0.0004) | 0.0201 (0.0005) |
| (5 ,11) | 0.0142 (0.0033) | 0.0174 (0.0040) |
| (6 ,11) | 0.0134 (0.0031) | 0.0183 (0.0041) |
| (7 ,11) | 0.0135 (0.0026) | 0.0176 (0.0043) |
| (8 ,11) | 0.0140 (0.0033) | 0.0151 (0.0040) |
| (9 ,11) | **0.0124 (0.0005)** | 0.0168 (0.0045) |
| (10 ,11) | 0.0149 (0.0039) | 0.0173 (0.0043) |

Table 3.5: Mean and standard deviation of the MSE values for the 'brain' image.

| | $(\widehat{\ell_x}, \widehat{\ell_y})$ | $\widehat{s}$ | $(\ell_x, \ell_y)$ | $s$ |
|---|---|---|---|---|
| $L_1$-norm | $(71/128, 68.2/128)$ | 1.12 | $(0.5, 0.5)$ | 1.14 |
| $L_2$-norm | $(70/128, 68.1/128)$ | 1.12 | $(0.5, 0.5)$ | 1.14 |

Table 3.6: Estimates and the true values of the zooming center and the zooming factor for the 'brain' image.

$112 \times 112$ of these images and zoom them to make those of size $128 \times 128$. We then follow the same procedure as described before and create the registered version of the reference images. The parameters $(r_1, r_2)$ selected for these images, i.e., the 'mountain' and 'Aral sea' are $(4, 11)$ and $(9, 10)$, respectively, for the registration procedure using the $L_1$-norm and $(12, 12)$ and $(8, 11)$, respectively, for the registration procedure using the $L_2$-norm. We present the evidence in support of that in Tables 3.7 and 3.8, respectively.

We present below the reference image, the zoomed image, and the registered images for the registration procedures using $L_1$-norm and $L_2$-norm, respectively. In Figure 3.10, we represent these images for the 'mountain' image and the 'Aral sea' image. In addition

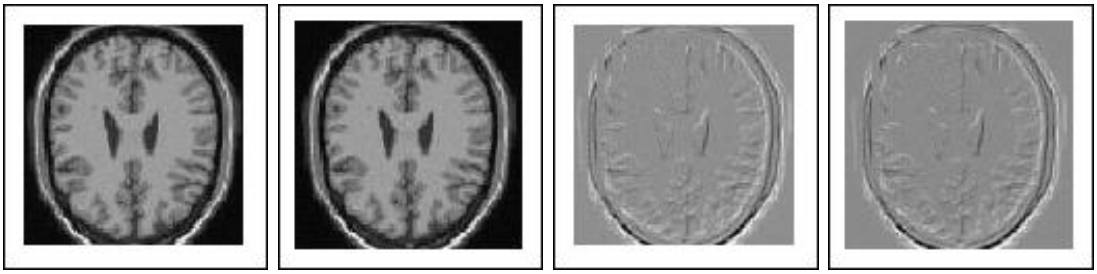


Figure 3.9: From left to right: (a) registered image using $L_1$-norm, (b) registered image using $L_2$-norm, (c) residual image using $L_1$-norm, (d) residual image using $L_2$-norm.

to that, we also provide the table for the parameter selection for the 'mountain' (see Table 3.7) and the 'Aral sea' images (see Table 3.8), respectively.

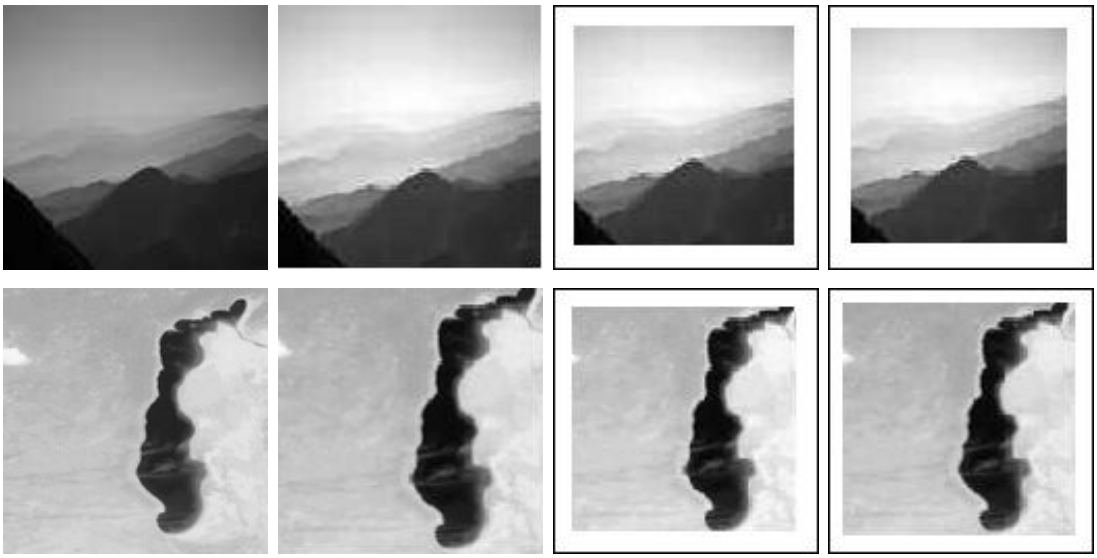


Figure 3.10: Top row: (from left to right) the reference image, the zoomed image, and the registered images under the registration procedure using $L_1$-norm and $L_2$-norm, respectively, for the 'mountain' image; Bottom row: same sequence of images for the 'Aral sea' image.

In the next section, we compare our method with some of the state-of-the-art methods on the basis of mean squared error (MSE) and normalized mutual information (NMI) between the reference image and the registered image. We also provide the registered images created by the different methods to facilitate comparison.

## 3.5 Comparison with State-of-the-art Methods

For comparisons, we consider a number of popular state-of-the-art methods that are all based on certain feature-matching algorithms. We find several popular feature matching algorithms in the literature, viz., Scale Invariant Feature Transform (SIFT) (Lowe, 2004),

| Mean (standard deviation) | | |
|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm |
| (3 ,11) | 0.0443 (0.0009) | 0.0474 (0.0010) |
| (4 ,11) | **0.0436 (0.0005)** | 0.0467 (0.0007) |
| (5 ,11) | 0.0441 (0.0005) | 0.0464 (0.0008) |
| (6 ,11) | 0.0439 (0.0006) | 0.0462 (0.0011) |
| (7 ,11) | 0.0439 (0.0004) | 0.0456 (0.0009) |
| (8 ,11) | 0.0440 (0.0003) | 0.0449 (0.0008) |
| (9 ,11) | 0.0443 (0.0005) | 0.0444 (0.0005) |
| (10 ,11) | 0.0444 (0.0004) | 0.0440 (0.0005) |
| (11 ,11) | 0.0444 (0.0006) | 0.0457 (0.0042) |
| (1 ,12) | 0.0452 (0.0011) | 0.0489 (0.0016) |
| (2 ,12) | 0.0454 (0.0009) | 0.0507 (0.0032) |
| (3 ,12) | 0.0441 (0.0005) | 0.0474 (0.0009) |
| (4 ,12) | 0.0440 (0.0006) | 0.0483 (0.0008) |
| (5 ,12) | 0.0437 (0.0004) | 0.0465 (0.0007) |
| (6 ,12) | 0.0441 (0.0006) | 0.0466 (0.0008) |
| (7 ,12) | 0.0438 (0.0006) | 0.0451 (0.0012) |
| (8 ,12) | 0.0440 (0.0005) | 0.0450 (0.0007) |
| (9 ,12) | 0.0442 (0.0005) | 0.0445 (0.0009) |
| (10 ,12) | 0.0443 (0.0005) | 0.0436 (0.0004) |
| (11 ,12) | 0.0443 (0.0005) | 0.0431 (0.0007) |
| (12 ,12) | 0.0444 (0.0004) | **0.0429 (0.0005)** |

Table 3.7: Mean and standard deviation of the MSE values for the 'mountain' image.

Speeded Up Robust Features (SURF) (Bay et al., 2008), Oriented Fast and Rotated Brief (ORB) ((Rublee et al., 2011)), Binary Robust Invariant Scalable Points (BRISK) (Leutenegger et al., 2011), KAZE (Alcantarilla et al., 2012), Accelerated-KAZE (AKAZE) (Alcantarilla and Solutions, 2011), to name a few. The primary working principles of these methods are the detection and description of the features in the reference and transformed images. Then the relationships between these features in the reference and transformed images are found using some algorithm, e.g., RANSAC (Derpanis, 2010), and registered images are generated. In this chapter, we choose ORB, AKAZE, and BRISK among these methods to compare with our best-performing method, i.e., registration using $L_1$-norm. These three methods are comparable in the sense that all of them are feature-based and invariant under scaling. To evaluate the performance of the algorithm in comparison with the existing methods, we consider MSE and normalized mutual information (NMI) (Cover, 1999) as the performance evaluation measures. These measures are widely used in the image registration literature and are helpful in assessing registration performance by quantifying the degree of similarity between two images. MI measures the mutual dependence between two images. Intuitively, it measures the reduced uncertainty of one image in the presence of the information of another image. Therefore, a larger value of

| Mean (standard deviation) | | |
|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm |
| (7 ,10) | 0.0042 (0.001) | 0.0035 (0.0007) |
| (8 ,10) | 0.0041 (0.0008) | 0.0025 (0.0002) |
| (9 ,10) | **0.0022 (0.0001)** | 0.0024 (0.0001) |
| (10 ,10) | 0.0023 (0.0002) | 0.0025 (0.0001) |
| (1 ,11) | 0.0055 (0.0023) | 0.0058 (0.0014) |
| (2 ,11) | 0.0029 (0.0002) | 0.0031 (0.0002) |
| (3 ,11) | 0.0034 (0.0011) | 0.0034 (0.0003) |
| (4 ,11) | 0.0025 (0.0007) | 0.003 (0.001) |
| (5 ,11) | 0.0033 (0.0005) | 0.0031 (0.0007) |
| (6 ,11) | 0.0043 (0.0012) | 0.0037 (0.0007) |
| (7 ,11) | 0.0044 (0.0007) | 0.0038 (0.0002) |
| (8 ,11) | 0.0037 (0.001) | **0.0024 (0.0002)** |
| (9 ,11) | 0.0022 (0.0001) | 0.0026 (0.0001) |

Table 3.8: Mean and standard deviation of the MSE values for the 'Aral sea' image.

MI is preferable for a greater degree of similarity. Mathematically, MI between reference and registered image can be expressed as

$$MI(I_R, \widehat{I}_R) = H(I_R) + H(\widehat{I}_R) - H(I_R, \widehat{I}_R),$$

where $H(I_R)$ and $H(\widehat{I}_R)$ are the information contained in the reference image and registered image, respectively. $H(I_R, \widehat{I}_R)$ is the joint entropy of $I_R$ and $\widehat{I}_R$. The normalized mutual information (NMI) is calculated as

$$NMI(I_R, \widehat{I}_R) = \frac{2MI(I_R, \widehat{I}_R)}{H(I_R) + H(\widehat{I}_R)}.$$

MSE is defined as a similarity measure based on the difference in gray level between two images. It can be expressed as,

$$MSE(I_R, \widehat{I}_R) = \sum_{(u,v)\in\Omega} (\widehat{I}_R(u,v) - I_R(u,v))^2,$$

where $\widehat{I}_R(u,v)$ and $I_R(u,v)$ are the image intensity value at the $(u,v)-$th pixel coordinate for the registered image and reference image, respectively. Initially, we apply these methods to the zoomed images and create the registered images. After that, we calculate the MSE and MI between the reference image and the registered images obtained with respect to these methods and provide a table for comparison. We also provide the reference image, the zoomed image, and the registered images side by side to facilitate visual comparison among these methods.

| | Mean squared Error (MSE) | | | |
|---|---|---|---|---|
| Image type | Proposed ($L_1$-norm) | ORB | AKAZE | BRISK |
| simulated | **0.0073** | NA | NA | NA |
| lake | **0.0034** | 0.0120 | 0.0490 | 0.0060 |
| brain | 0.0124 | **0.0112** | 0.0556 | 0.0129 |
| mountain | **0.0436** | 0.0717 | NA | NA |
| Aral sea | **0.0022** | 0.0092 | 0.0180 | 0.0081 |
| | Normalized Mutual Information (NMI) | | | |
| Image type | Proposed ($L_1$-norm) | ORB | AKAZE | BRISK |
| simulated | **1.1331** | NA | NA | NA |
| lake | 1.1567 | 1.2306 | 1.0732 | **1.3234** |
| brain | 1.1466 | 1.2436 | 1.0712 | **1.2603** |
| mountain | **1.2051** | 1.1740 | NA | NA |
| Aral sea | 1.2347 | 1.1974 | 1.1589 | **1.2778** |

Table 3.9: Mean squared error (MSE) and normalized mutual information (NMI) for various methods and images.

Next, we present Table 3.9 containing MSE and NMI values computed for different images, viz., the simulated image, the 'lake' image, the 'brain' image, the 'mountain' image, and the 'Aral sea' image. From Table 3.9, we see that, for the simulated image, as the number of point features of the image (according to the definition of the features for the methods ORB, AKAZE, and BRISK) are sparse, these methods fail to even generate the registered image, whereas our proposed method is performing well. In the case of the 'lake' image, our proposed method clearly outperforms other methods, with the closest competitor being the BRISK method with respect to MSE (with the MSE value being almost twice as high as ours), but BRISK beats our method with respect to NMI. In the case of the 'brain' image, the ORB method performs marginally better than our proposed method as far as MSE is concerned, and BRISK, along with ORB, outperforms our method if NMI is taken into consideration. For the 'mountain' image, AKAZE and BRISK are not able to generate the registered images, and our method outperforms the other method, ORB, convincingly by both MSE and NMI. In this case, ORB generates a distorted registered image. Last but not least, our method works very well on the satellite image of 'Aral Sea' and outperforms ORB and AKAZE with respect to both MSE and NMI. In the case of 'Aral Sea', our method outperforms BRISK in terms of MSE, and is comparable in terms of NMI. In this case, all competing methods are able to generate registered images, but they are somewhat distorted.

Along with Table 3.9, we also provide the registered images created by different methods (the proposed and others) for the purpose of comparisons among these methods, which can be found in Figure 3.11. In Figure 3.11, each row represents the results corresponding

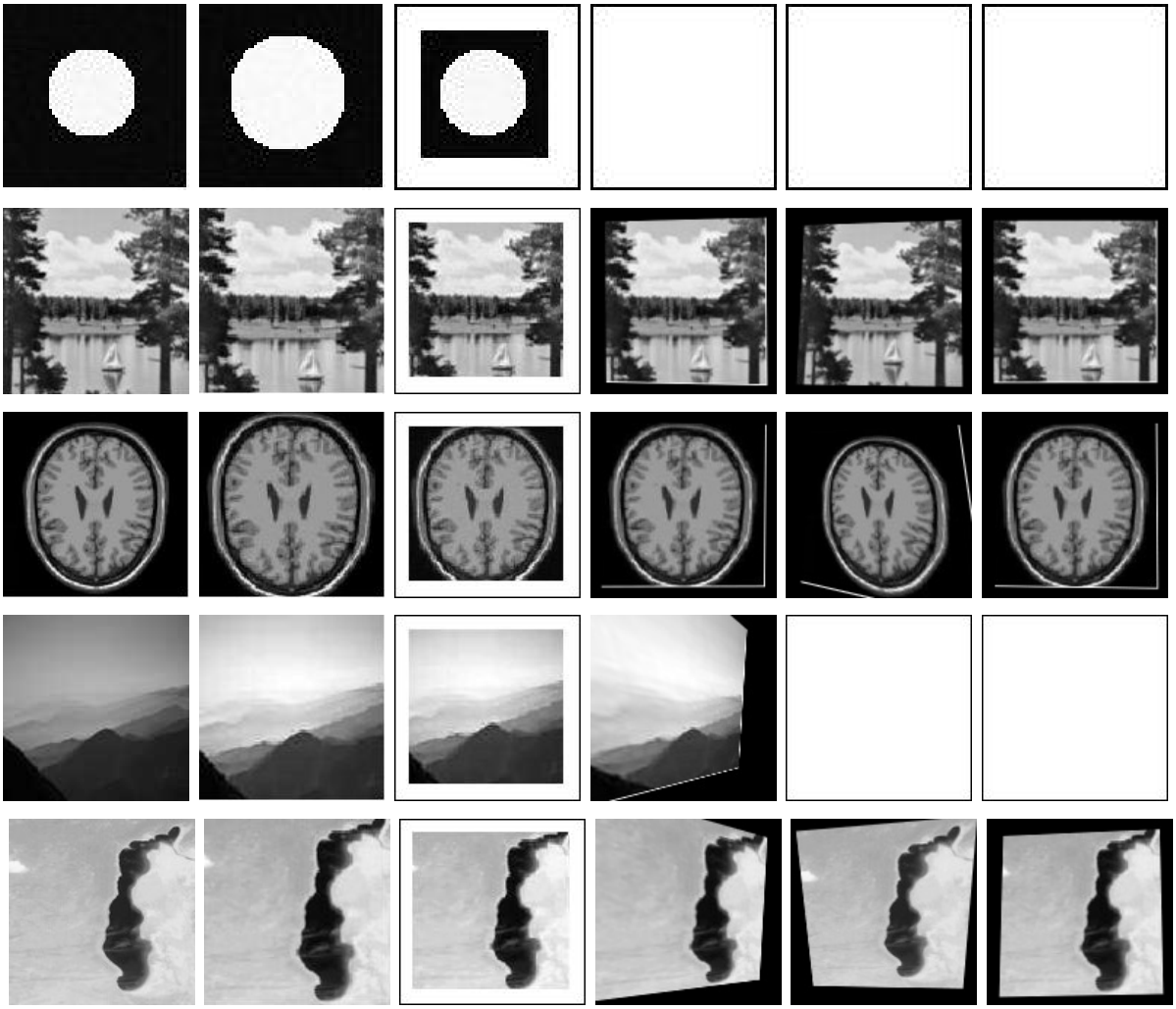


Figure 3.11: The reference image, the zoomed image, and the registered images under the proposed method, ORB, AKAZE, and BRISK.

to a specific image. In each row, the placement of the images is in the following order: the reference image, the zoomed image, the registered images under the proposed method, ORB, AKAZE, and BRISK, respectively.

We see from Figure 3.11 that our proposed method is the clear winner in the case of the simulated image, as the other methods are not able to even generate the registered image. In the case of the 'lake' image and the 'brain' image, our method works reasonably well compared to the other methods. We also observe that the AKAZE method generates a somewhat distorted image in the case of the 'brain' image. If we consider the 'mountain' image and the 'Aral sea' image, our method comfortably outperforms other methods.

## 3.6 Discussions

In this chapter, we propose a feature-based registration method to register a zoomed image to its original or reference version. This method relies on the principle of extraction of the most basic features, i.e., the edges of the reference and the zoomed image, mapping those features, and finally, finding the relationship between the features to create the registered image.

One major advantage of the proposed method is that it works even when the number of point features is small in the images, when most of the competing methods fail to perform. However, there are still some issues with this method that should be addressed in future research. In the theoretical justification, the assumption $r_2/n \to 0$ states that the amount of zooming or the zooming factor is small. We prove the consistency of the registration procedure under this assumption. Dropping this assumption can certainly cover a more general case of large zooming. In that scenario, stronger assumptions on the true image intensity function will be required. For example, if the true image intensity function is piecewise constant, then a large $r_2$ can be used and the assumption $r_2/n \to 0$ will not be needed. The given proof can be easily extended to handle this situation. However, this new assumption is also restrictive in the sense that it does not hold true in many real-life images. The given proof considers the situation when the zooming factor is small, but it works for a wide range of images. Another drawback is that the computational complexity of the procedure, though significantly less than the approach mentioned in the previous chapter, is still on the heavier side in comparison to some state-of-the-art approaches. Since we consider edges as the primary feature here, this procedure would take more time for an image with a larger number of edges. This scenario would substantially improve if we selected something else as the primary feature, which is few in number and also available in almost all non-trivial images.

Our method works well in many situations, as shown in 3.4, and it also outperforms some state-of-the-art methods, as displayed in Section 3.5. We also see from Section 3.3 that the performance of the feature mapping improves as the image resolution increases, and this method is also able to estimate the zooming center and the zooming factor consistently. A much more complex and challenging problem to deal with would be registering images when translation and rotation are also involved, along with zooming.

# Chapter 4

# Image Registration for Rigid-body Transformation and Zooming: Feature-based Approach

## 4.1 Introduction

Previous chapters deal with an image registration problem for two images where one is a zoomed-in version of the other. The success of our proposed methods in previous chapters motivates us to take up a more complex image registration problem where the image transformation involves translation and rotation, along with zooming. We display one such example in Figure 4.1. In this case also, we propose a feature-based image registration approach, where we consider edges as the primary image feature, as it is readily available in almost all non-trivial images.

Figure 4.1: The reference image, and the moved image. The image transformation involves translation, rotation, and zooming.

In this chapter, we propose a novel and efficient feature-based image registration technique that takes image transformations such as translation, rotation, and zooming/ scaling into consideration. For this feature-based image registration technique, non-point feature edge points or jump points are taken into consideration as the primary image feature. It is to be noted that, in almost all non-trivial images, edge points exist. We tackle the image registration problem under the aforementioned transformations in the following way. First, we identify the edge points in the moved image as well as in the reference image. Then, before explaining the relationship between the edge pixels of the reference and the moved image, we take recourse to the feature mapping algorithm, as the numbers of edge pixels of the reference and the moved image are not the same. The image object in the zoomed image gets bigger, and as a result, the number of detected edge pixels in the zoomed image is generally larger than those in the original reference image. This is because the number of pixels near the edges is usually large in the zoomed image compared to the reference image. However, in some cases, if some part of the image object goes out of the reference frame while zooming in, the number of edge pixels may decrease as well. Once the feature mapping step is done, we get a set of feature points for each of the reference and moved images having the same cardinality. Next, we suggest a method to estimate the rotation angle after adjusting for the translation. Finally, a mathematical model is used to explain the relationship between the edge pixels of the reference and the moved image. We then generate the registered image using the said relationship. The performance of our proposed method is pretty satisfactory in many situations, as discussed in this chapter. But this method performs remarkably well compared to some of the state-of-the-art methods mentioned in the previous paragraph if the number of point features is scarce in the concerned image.

We organize the rest of the chapter as follows. In Section 4.2, we describe the proposed image registration procedure. Some theoretical justifications are discussed in Section 4.3. Section 4.4 presents the application of the procedure on several images and provides the corresponding numerical results. We then compare the performance of our procedure with several state-of-the-art procedures in Section 4.5. Some remarks on the proposed method and some future directions are provided in Section 4.6.

## 4.2 Proposed Image Registration Procedure

We list the assumptions on the image intensity functions and the nature of image transformations in Section 4.2.1. Section 4.2.2 describes the jump regression model of the image intensity function and an edge detection method. We state the transformation model in Section 4.2.3. Next, the pivotal steps of the proposed registration procedure—feature

selection and feature mapping—are described in Sections 4.2.4 and 4.2.5, respectively. In Section 4.2.6, we explain the process of the rotation angle estimation after the initial adjustment of the shift due to translation. Furthermore, we describe the mathematical model that explains the relationship between the detected features and the mapped features in Section 4.2.7. Finally, the parameter selection process is mentioned in 4.2.9.

### 4.2.1 Assumptions

In this chapter, we assume that the true intensity functions of the 2-D grayscale images are piecewise constants. We consider only three types of transformations: translation, rotation, and zooming/scaling, where the first two types are called rigid-body transformations, while the last one is a special case of non-rigid-body transformation. Additionally, we further assume that the image center is the center of both rotation and zooming. This assumption would be usually appropriate if the image object in the reference image is close to the center of the image, which is often the case in many real-life images. In case the image object is away from the center of the reference image, then one can translate the image in a way that the image object is at the center of that image. In order to check whether this assumption is at least reasonable, one needs to detect the edge pixels in the reference image, and find the center of those edge pixels. If the image object is close to the center of the reference image, then the center of those edge pixels would be close to the center of the reference image. For numerical studies, we work on several pairs of reference and transformed images of the same image resolution.

### 4.2.2 Edge Detection by Local Kernel smoothing

The detection of jump points, or edges, using a suitable edge detector is a crucial step in our proposed image registration procedure. Theoretically, any reasonable edge detector can be used. In this chapter, we employ a local linear kernel estimator based edge detection technique. The following regression model can be used to represent a 2-D grayscale image under the jump regression analysis (JRA) framework (See Qiu (2005)):

$$I_f(x_i, y_j) = f(x_i, y_j) + \varepsilon_f(x_i, y_j) \quad \text{for} \quad i, j = 1, 2, ....n, \tag{4.2.1}$$

where $\{(x_i, y_j) : i, j = 1, 2, ...., n\}$ are equally spaced pixel coordinates in the design space $\Omega = [0, 1] \times [0, 1]$. The unknown image intensity function is denoted by $f$, the sample size is $N = n^2$, and the random errors $\varepsilon_f$ are independent and identically distributed with a mean of 0 and a variance of $\sigma^2 > 0$. We now consider the circular neighborhood of radius $h_n$ around each pixel $(x, y)$, denoted by $O(x, y; h_n) = \{(u, v) : \sqrt{(u - x)^2 + (v - y)^2} \leq h_n\}$,

where $h_n > 0$ is a bandwidth parameter in our edge detection step. Next, we perform local linear kernel (LLK) smoothing by minimizing the following objective function:

$$\min_{\alpha,\beta_1,\beta_2} \sum_{(x_i,y_j)\in O(x,y;h_n)} \left[ I_f(x_i,y_j) - \Big\{ \alpha + \beta_1(x_i - x) + \beta_2(y_j - y) \Big\} \right]^2 K_{ij}, \tag{4.2.2}$$

where $K_{ij} = K\left(\frac{y_j - y}{h_n}, \frac{x_i - x}{h_n}\right)$, $K$ being a bivariate density kernel that is radially symmetric, and has support $\{(x,y) : x^2 + y^2 \leq 1\}$. From the aforementioned minimization problem (4.2.2), the estimate of $(\alpha, \beta_1, \beta_2)$ is expressed as $\Big(\widehat{\alpha}(x,y), \widehat{\beta}_1(x,y), \widehat{\beta}_2(x,y)\Big)$. The LLK estimate of $f(x,y)$ is $\widehat{\alpha}$, and the LLK estimate of the gradient vector $(f'_x, f'_y)^T$ is $(\widehat{\beta}_1, \widehat{\beta}_2)^T$, when there is no jump in $O(x,y;h_n)$. We now split $O(x,y;h_n)$ into two halves, $O_1(x,y)$ and $O_2(x,y)$, in the direction perpendicular to the estimated gradient vector, and compute Nadaraya–Watson local constant kernel estimators of $f(x,y)$ using the intensity values in these two halves, which we label as $\widehat{f}_{LCK,O_1}$ and $\widehat{f}_{LLK,O_2}$, respectively. Then, if $|\widehat{f}_{LCK,O_1}(x,y) - \widehat{f}_{LCK,O_2}(x,y)|$ is large, the corresponding design point $(x,y)$ is marked as a detected jump pixel (Kang and Qiu, 2014). For additional information regarding the jump detection process, readers are referred to (Kang and Qiu (2014); Qiu (2004)).

### 4.2.3 Model Description

We intend to register two images of the same scene: the actual image, also known as the reference image, and we denote it by R, and the moved image, i.e., translated, rotated, and zoomed version of the reference image. We denote the latter by M. Registration of these two images calls for finding a one-to-one mapping between these two images. In other words, we wish to derive a mathematical function such that it generates the reference image, once applied to the moved image. We express this problem more precisely with the help of the following statistical model. Suppose we have two images of resolutions $n \times n$. We denote their true image intensity functions as $R(x,y)$ and $M(x,y)$. It is assumed that they have the following relationship:

$$M(T_1(x,y), T_2(x,y)) = R(x,y), \text{ for } (x,y) \in \Omega, \tag{4.2.3}$$

where $\Omega = [0,1] \times [0,1]$ is the design space of image R and $T(x,y) = (T_1(x,y), T_2(x,y))$ is the transformation we wish to estimate. We estimate $T(x,y)$ from observed image intensities of the two images defined below:

$$\begin{aligned} I_R(x_i,y_j) &= R(x_i,y_j) + \varepsilon_R(x_i,y_j) \quad \text{for } i,j = 1,2,....n, \\ I_M(x_i,y_j) &= M(x_i,y_j) + \varepsilon_M(x_i,y_j) \quad \text{for } i,j = 1,2,....n, \end{aligned} \tag{4.2.4}$$

where $\{(x_i, y_j) : i, j = 1, 2, ...., n\}$ are equally spaced pixel locations and $\varepsilon_R(x_i, y_j)$ and $\varepsilon_M(x_i, y_j)$ are i.i.d. random errors in these images with mean 0 and unknown variances $\sigma_R^2$ and $\sigma_M^2$, respectively.

In order to register these two images, we use the intensities of a number of specific points, known as feature points, rather than the intensities of all pixels. We elaborate on the feature points and their selection procedure in the following subsections.

### 4.2.4 Feature Selection

Features can broadly be classified into two groups: point features and non-point features. Corner points, landmark points, and centroid points are examples of point features, whereas non-point based features include curves, edges, regions, and so forth. In this chapter, we use edge, the most popular feature of an image, for registration purposes. The reason behind choosing edge as the primary feature is that it is the most basic feature of an image, and is readily available in all images in practice. Next, we detect the edge locations of the reference image (R) and the moved image (M) using the method described in Section 4.2.2.

Let $\{(u_i', v_i') : i = 1, 2, ..., n_1\}$ be the detected edge pixels for the reference image and $\{(U_i, V_i) : i = 1, 2, ..., n_2\}$ be the detected edge pixels for the moved image, where $n_2 \geq n_1$ usually, as the moved image is zoomed. Now, we aim to find a relationship between these edge pixels of the moved image and the reference image to create the registered version of the moved image. A widely used 2D parametric function that captures the relationship between the reference image and the moved image is stated below:

$$\begin{aligned} \Big(T_1(x, y) - c_x\Big) &= \quad s\Big((x - c_x).\ \cos\theta + (y - c_y).\ \sin\theta\Big) + \ell_x, \\ \Big(T_2(x, y) - c_y\Big) &= s\Big(-(x - c_x).\ \sin\theta + (y - c_y).\ \cos\theta\Big) + \ell_y, \end{aligned} \tag{4.2.5}$$

where $(\ell_x, \ell_y)$ are the translation parameters, $\theta$ is the rotation parameter, $s$ is the zooming/scaling parameter, and $(c_x, c_y)$ is the image center. Eqn. (4.2.5) is constructed in such a way that it takes into account the fact that the rotation and the zooming center of the transformation are the image center, which is one of the assumptions of our procedure. A point to be noted is that we cannot readily use this model for our purpose as the cardinalities of the sets of edge pixels for the reference image and the moved image are not the same. In addition to that, it is also not known to us which $(u_i', v_i')$ of the reference image has shifted to a given $(U_i, V_i)$ of the moved image. The next subsection discusses a feature mapping procedure.

### 4.2.5 Feature Mapping

We propose the following steps for performing feature mapping:

**Step 1:** We examine a circular neighborhood of radius $r_1$ around a given edge pixel $(U, V)$ in the moved image, denoted as $O(U, V; r_1)$.

**Step 2:** Next, we analyze a circular neighborhood of radius $r_2$ in the reference image, surrounding the same coordinates as the given pixel, denoted as $O(U, V; r_2)$, where $r_2 \geq r_1$. We assume a circular neighborhood of radius $r_1$ around $(u, v)$, denoted as $O(u, v; r_1)$, for any pixel $(u, v)$ in that neighborhood $O(U, V; r_2)$. Figure 4.2 explains the procedure.

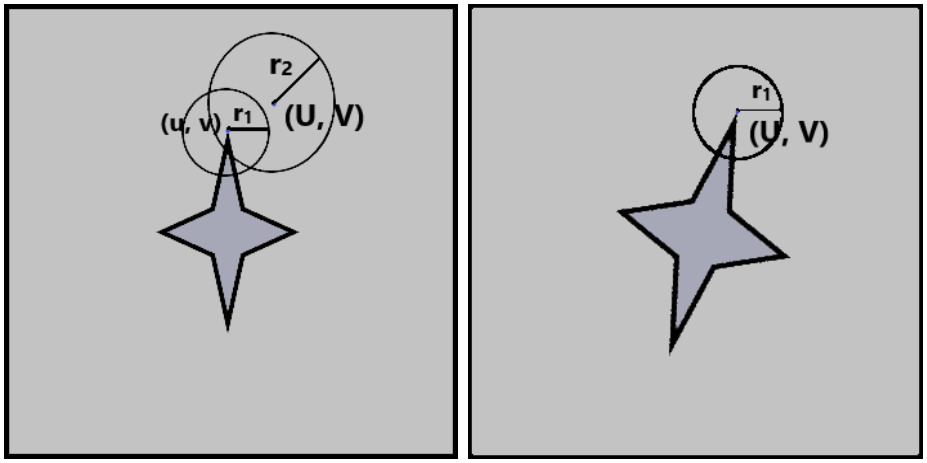



Figure 4.2: A demonstration of the feature mapping algorithm. (a): Reference image, (b): Moved image.

**Step 3:** Using two measures, viz., mean squared difference (MSD) and mean absolute difference (MAD), we compare the two neighborhoods, $O(U, V; r_1)$ of the moved image and all possible choices of $O(u, v; r_1)$ corresponding to the reference image. Based on these two measures, we select the neighborhood $O(u, v; r_1)$ of the reference image that is most similar to the neighborhood $O(U, V; r_1)$ of the moved image. The moved image's pixel $(U, V)$ is then mapped to the reference image's pixel $(u, v)$. Below is a definition of these two similarity measures:

$$\text{MSD}((u, v); (U, V)) = \frac{1}{N} \sum_{s,t} (I_R(u + s, v + t) - I_M(U + s, V + t))^2, \tag{4.2.6}$$

$$\text{MAD}((u, v); (U, V)) = \frac{1}{N} \sum_{s,t} \Big| I_R(u + s, v + t) - I_M(U + s, V + t) \Big|, \tag{4.2.7}$$

where both summations are over the set $\{(s, t) : (u + s, v + t) \in O(u, v; r_1)\}$, N is the number of pixels in $O(u, v; r_1)$, $I_{(.)}(a, b)$ indicates the intensity of the image pixel (a, b), and R and M represent the reference and the moved image, respectively. The following

are the mapped pixels:

$$\arg\min_{(u,v)\in O(U,V;r_2)} \text{MSD}((u,v);(U,V)), \tag{4.2.8}$$

$$\arg\min_{(u,v)\in O(U,V;r_2)} \text{MAD}((u,v);(U,V)), \tag{4.2.9}$$

corresponding to the measures MSD and MAD, respectively. The neighborhood radii, $r_1$ and $r_2$, are the two tuning parameters for this procedure. The choice of these parameters has a significant impact on how well this process performs. We discuss the parameter selection procedure in Section 4.2.9.

### 4.2.6 Estimation of the Rotation Angle

The estimation of the rotation angle is a pivotal step in this registration procedure. After we perform feature mapping, each edge pixel of the moved image gets mapped to a single edge pixel of the reference image. Suppose $(U_i, V_i)$ of the moved image gets mapped to $(u_i, v_i)$ of the reference image for $i = 1 \ldots, n_2$, where $n_2$ is the number of edge pixels of the moved image. In this chapter, we assume that the image center is the center of both rotation and zooming as well. Suppose the center of each image is $(c_x, c_y)$, and $(u_i', v_i')$'s are the original locations of the reference image that has shifted to $(U_i, V_i)$'s in the moved image after transformation. Therefore, we can calculate the rotation angle by calculating the angle between $(u_i', v_i')$ and $(U_i, V_i)$ with respect to $(c_x, c_y)$. However, the $(u_i, v_i)$'s obtained from the feature mapping method may not always be exactly the same as the corresponding $(u_i', v_i')$'s, i.e., the correct locations. Thus, some estimation error, however small, might creep into the estimation of the rotation angle. We resolve this by estimating the rotation angle for each pair of $(U_i, V_i)$ and $(u_i, v_i)$ and then using a measure of central tendency to estimate the rotation angle of the image transformation.

Note that after rotation and zooming of an image, the image center is still the rotation and zooming center. However, after subsequent translation, the image center is no longer the same as the rotation and zooming center. Therefore, if we estimate the angle of rotation with respect to $(c_x, c_y)$ it will be erroneous. In the presence of translation, we first find preliminary estimates of the translation parameters and use those estimates to bring the edge pixels of the moved image back in the opposite direction so that translation in the modified moved image is essentially removed. Now we can safely assume that the center of the modified edge pixels is $(c_x, c_y)$ and find the angle of rotation between the edge pixels of the modified moved image and $(u_i, v_i)$'s of the reference image with respect to $(c_x, c_y)$. We perform the following transformation on $(U_i, V_i)$ of the moved image to

create the modified moved image:

$$\begin{aligned} U_i^* &= U_i - \bar{U} + \bar{u}, \\ V_i^* &= V_i - \bar{V} + \bar{v}, \end{aligned} \tag{4.2.10}$$

where $(U_i^*, V_i^*)$, for $i = 1, \ldots, n_2$ are the edge pixels of the modified moved image and $(\bar{u} - \bar{U}, \bar{v} - \bar{V})$ are the preliminary estimates of the translation parameters.

Readers are referred to Figure 4.3 for the visual demonstration of the main idea behind the estimation of the rotation angle. The left and middle panels in Figure 4.3 show the reference image and the moved image, where translation, rotation, and zooming are all present, respectively. The angle estimation from the image in the middle panel is erroneous. Note that in this case, the estimated angle $\theta'$ is greater than the true angle $\theta$. The right panel of Figure 4.3 shows the correct procedure to estimate the rotation angle after adjusting for the translation. Here, $\theta$ is the angle between $(u_i', v_i')$ of the reference image and the modified edge pixel $(U_i^{*'}, V_i^{*'})$ of the moved image, which is the true angle of rotation and can be calculated using the following formula:

$$\theta = \cos^{-1}\left[\frac{\{(u_i' - c_x, v_i' - c_y).(U_i^{*'} - c_x, V_i^{*'} - c_y)\}}{|(u_i' - c_x, v_i' - c_y)||(U_i^{*'} - c_x, V_i^{*'} - c_y)|}\right],$$

where, $U_i^{*'} = U_i - \bar{U} + \bar{u'}$, $V_i^{*'} = V_i - \bar{V} + \bar{v'}$, $\bar{u'} = \sum_{i=1}^{n_2} u_i'/n_2$, and $\bar{v'} = \sum_{i=1}^{n_2} v_i'/n_2$. Here, modification has been done on the basis of the true edge pixel of the reference image. As this true angle of rotation can not be found readily, we take recourse to the following procedure to estimate the rotation angle $\theta$. After applying the transformation mentioned in Eqn. (4.2.10) to the edge pixels of the moved image, we can find the estimate of the angle of rotation by following the method mentioned in the first paragraph of Section 4.2.6. Let $\theta_i$ be the angle between the matched pair of edge pixels $(u_i, v_i)$ and $(U_i^*, V_i^*)$ of the reference image and moved images with respect to the image center $(c_x, c_y)$. Then, for $i = 1, \ldots, n_2$, $\theta_i$ can be calculated using the following formula:

$$\theta_i = \cos^{-1}\left[\frac{\{(u_i - c_x, v_i - c_y).(U_i^* - c_x, V_i^* - c_y)\}}{|(u_i - c_x, v_i - c_y)||(U_i^* - c_x, V_i^* - c_y)|}\right],$$

where (.) used in the numerator of the formula indicates the dot product between the vectors, and $|(x, y)|$ is the magnitude of the vector $(x, y)$, i.e., $\sqrt{(x^2 + y^2)}$. Next, we use the sample mean of these $\theta_i$ values to estimate the rotation angle $\theta$. Therefore, $\widehat{\theta} := \sum_{i=1}^{n_2} \theta_i/n_2$. After estimating the rotation angle, we move on to estimate other parameters of the model mentioned in equation (4.2.5), which is described in Section 4.2.7, and finally create the registered version of the moved image.

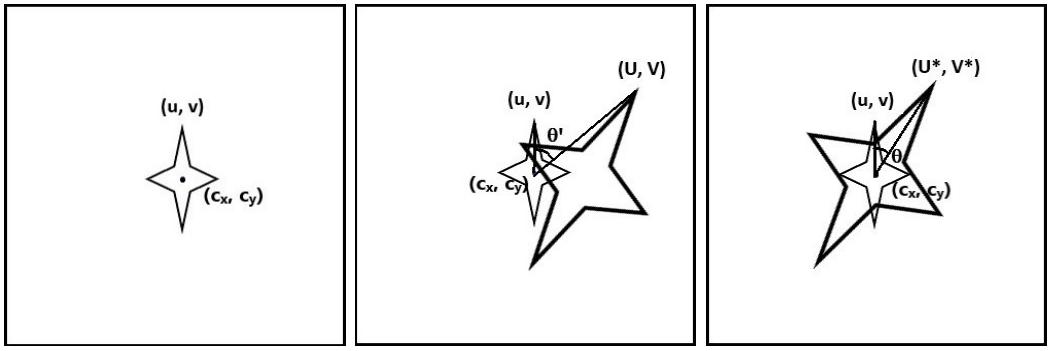



Figure 4.3: A demonstration of the rotation angle. (a): Reference image, (b): Moved image, (c): Moved image after adjusting for translation.

### 4.2.7 Final Steps to Create the Registered Image

At this stage, we use the previously discussed feature mapping method to identify the mappings of the reference and the moved image's edge pixels and create the registered version of the moved image. We start with the moved image's edge pixels and use the feature mapping to find their counterparts in the reference image. We fit the model in Eqn. (4.2.5) to the data points, i.e., the edge pixels of the reference and moved image.

The sets of detected edge pixels for the reference and the moved image are $D_R$ and $D_M$, respectively. From (4.2.5), we have,

$$\begin{aligned}\Big(U_i - c_x\Big) &= s\Big((u_i - c_x).\ \cos\theta + (v_i - c_y).\ \sin\theta\Big) + \ell_x + \zeta_{U_i} \\ \Big(V_i - c_y\Big) &= s\Big(-(u_i - c_x).\ \sin\theta + (v_i - c_y).\ \cos\theta\Big) + \ell_y + \zeta_{V_i},\end{aligned} \tag{4.2.11}$$

where $\zeta_{U_i}$ and $\zeta_{V_i}$ are the errors associated with feature mapping. If the feature mapping were perfect, the equation on the right-hand side without the error terms would have perfectly explained $(U_i, V_i)$.

In the above equations, we need to estimate $s$, $\ell_x$, and $\ell_y$. As $\theta$ has already been estimated as $\widehat{\theta}$, we can rewrite the above equations as follows:

$$U_i - c_x = su_i^* + \ell_x + \zeta_{U_i},\ \ V_i - c_y = sv_i^* + \ell_y + \zeta_{V_i},$$

where $u_i^* = \Big((u_i - c_x)\cos\widehat{\theta} + (v_i - c_y)\sin\widehat{\theta}\Big)$, $v_i^* = \Big(-(u_i - c_x)\sin\widehat{\theta} + (v_i - c_y)\cos\widehat{\theta}\Big)$. To estimate $\phi = (s, \ell_x, \ell_y)^T$ from the above equations, we consider minimizing the following objective function:

$$S(s, \ell_x, \ell_y) = \sum_{(U_i, V_i) \in D_M} \{(\overline{U_i - c_x} - su_i^* - \ell_x)^2 + (\overline{V_i - c_y} - sv_i^* - \ell_y)^2\}. \tag{4.2.12}$$

The estimator obtained from the above minimization is:

$$\widehat{\phi} = \begin{pmatrix} \widehat{s} \\ \widehat{\ell}_x \\ \widehat{\ell}_y \end{pmatrix} = C^{-1}D, \tag{4.2.13}$$

where

$$C = \begin{pmatrix} \sum_{(U_i,V_i)\in D_M}(u_i^{*2} + v_i^{*2}) & \sum_{(U_i,V_i)\in D_M} u_i^* & \sum_{(U_i,V_i)\in D_M} v_i^* \\ \sum_{(U_i,V_i)\in D_M} u_i^* & |D_M| & 0 \\ \sum_{(U_i,V_i)\in D_M} v_i^* & 0 & |D_M| \end{pmatrix},$$

$$D = \begin{pmatrix} \sum_{(U_i,V_i)\in D_M}(u_i^*(U_i - c_x) + v_i^*(V_i - c_y)) \\ \sum_{(U_i,V_i)\in D_M}(U_i - c_x) \\ \sum_{(U_i,V_i)\in D_M}(V_i - c_y) \end{pmatrix}.$$

It is not difficult to check that $C^{-1}$ always exists when $|D_M| > 0$.

### 4.2.8 Steps summarizing the proposed registration algorithm

**Step 1:** Detect the edge pixels, i.e., the features of the reference and the moved image, following the method described in Section 4.2.2. For each edge pixel, execute Steps 2-4.

**Step 2:** Follow the feature mapping procedure described in Section 4.2.5 to find the best match of an edge pixel of the moved image in the reference image.

**Step 3:** Estimate the rotation angle by following the steps described in Section 4.2.6.

**Step 4:** Estimate other parameters of the model, e.g., $s$, $\ell_x$, and $\ell_y$ from the equation 4.2.11 and 4.2.13.

**Step 5:** Finally, create the registered image using Equation 4.2.11.

### 4.2.9 Selection of Procedure Parameters

There are three procedure parameters in the proposed image registration method: the bandwidth parameter $h_n$ for detecting the edge pixels, and the radii $r_1$ and $r_2$ of the two circular neighborhoods associated with the feature mapping step. We suggest the following procedure for choosing the procedure parameters.

**Selection of $h_n$:** Since the choice of this parameter does not significantly affect the registration performance, we select $h_n$ in an ad hoc manner in this study. Our recommen-

dation is to select $h_n \in [\frac{2}{n}, \frac{3}{n}]$ based on our numerical experience. In all numerical studies presented in Section 4.4, we use $h_n = \frac{2}{n}$.

**Selection of $r_1$ and $r_2$:** We explore the mapping between $(u, v)$ and $(U, V)$ over a range of potential values for $r_1$ and $r_2$. Then, using the procedures outlined in Sections 4.2.4, 4.2.5, and 4.2.6, we generate the registered version of the moved image. By calculating their mean squared error (MSE), we may determine how close the reference image and its registered version are to each other. We express this as

$$MSE(\widetilde{r}_1, \widetilde{r}_2) = \sum_{(u,v)\in R} (\widehat{I}_M(u, v, \widetilde{r}_1, \widetilde{r}_2) - I_R(u, v))^2,$$

where $\widetilde{r}_1$ and $\widetilde{r}_2$ are the feature mapping parameters, and $\widehat{I}_M(u, v, \widetilde{r}_1, \widetilde{r}_2)$ is the intensity of the pixel $(u, v)$ of the registered version $(\widehat{I}_M)$ of the moved image M. We select those values of $\widetilde{r}_1$ and $\widetilde{r}_2$ for which $MSE(\widetilde{r}_1, \widetilde{r}_2)$ is minimum.

## 4.3 Statistical Properties

In this section, we look at various statistical aspects of the proposed image registration procedure. The proposed strategy consists of a few fundamental steps, which we justify theoretically.

**Proposition 4.3.1.** *Consider a reference image with a resolution of $n \times n$, and let $(u', v')$ be an edge pixel of the said image. The image has been moved to some extent using the transformations, translation, rotation, and zooming. The pixel $(u', v')$ in the reference image has been shifted to $(U, V)$ in the moved image, where $U = u' + c_1$ and $V = v' + c_2$, with $(c_1, c_2) \in \mathbb{R}^2$. While registering these two images, the feature mapping technique of our algorithm with window widths $r_1$ and $r_2$ maps the pixel $(U, V)$ of the moved image to the pixel $(u, v)$ of the reference image. Then we show that,*

$$\frac{1}{n^2}[(u - u')^2 + (v - v')^2] \xrightarrow{P} 0,$$

*if $n \to \infty$ and $r_2 \to \infty$ such that $r_2/n \to 0$.*

**Proposition 4.3.2.** *Under the same assumptions as in Proposition 1, $\widehat{\theta} \xrightarrow{P} \theta$, and $\widehat{\phi} \xrightarrow{P} \phi$.*

The above propositions establish the statistical consistency of our proposed image registration method. The sketches of the proofs are provided in Appendix A.3.

## 4.4 Numerical Studies

This section evaluates the numerical performance of the proposed registration method. In this section, we study the performance of the proposed method on a variety of images, both simulated and real. We begin with the moved image and construct the registered version of the moved image using the technique outlined in Section 4.2. We begin with the application of our method on a simulated image of resolution $128 \times 128$, which is followed by two subsequent applications on a medical image and a satellite image of resolution $128 \times 128$ each.

**Performance on a simulated image:**

We first simulate a reference image and its moved version, both of resolution $128 \times 128$ (see Figure 4.4). In the reference image, we first set the intensity value 0 for all pixel locations. Then we construct an elliptical region of the reference image having a center at the image center, which is $(64, 64)$ in this case, and a major axis (parallel to the Y-axis) and a minor axis (parallel to the X-axis) of lengths 20 and 15, respectively, and update the intensity values of all pixels inside the elliptical region to 1. Finally, we construct a rectangular region inside the same image with vertices $(51, 41)$, $(51, 90)$, $(70, 90)$, and $(70, 41)$, and update the intensity values of all pixels inside the rectangular region to 1. This concludes the creation of the reference image. Next, we zoom in on the core region of resolution $112 \times 112$ while keeping the resolution of the moved image the same as the reference image, and then rotate the image clock-wise by 10 degrees. In order to add the translation component, we finally move the zoomed and rotated image two rows above and two columns to the right, and this completes the creation of the simulated moved image. Figure 4.4 shows both the reference and moved image. The black border in Figure 4.4(b) is generated by the software while incorporating the translation component.

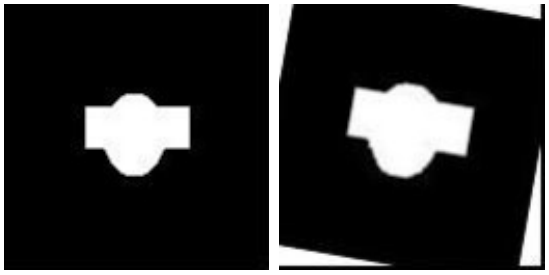


Figure 4.4: The Simulated image: (a) reference (left), (b) moved (right).

Then, we detect the edge pixels of the reference and the moved images using the method described in Section 4.2.2. With the intention of further shortening the runtime of the mentioned feature mapping procedure, we perform the feature mapping on the 20% feature points of the shifted or moved image that are closest to the center, and feature

mapping is done on those points. The idea behind choosing the feature points closest to the center is that the feature points chosen in the reference image will be inside the frame of the moved image. We then determine $r_1$ and $r_2$ for feature mapping using the procedures described in Section 4.2.5. In order to achieve this goal, we take 13 values of $r_2$ ranging from 4 to 16, and we consider $r_1$ to be between 1 and the value of $r_2$ for each $r_2$. We carry out the process for each of these $r_1$ and $r_2$ choices and produce the registered version of the moved image. Next, we determine the mean squared error (MSE) between the intensities of the reference image and the registered version of the moved image, corresponding to two similarity metrics: MAD and MSD. Table 4.1 presents a partial depiction of the results that contains the choices of $r_1$ and $r_2$ for which the MSE obtained from the previously indicated process is minimum for the registration process using $L_1$-norm and $L_2$-norm, respectively.

| Mean Squared Error | | |
|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm |
| (7, 10) | 0.0241 | 0.0241 |
| (8, 10) | 0.0227 | 0.0227 |
| (9, 10) | 0.0186 | 0.0186 |
| (10, 10) | 0.0207 | 0.0207 |
| (1, 11) | 0.1317 | 0.1317 |
| (2, 11) | 0.0762 | 0.0762 |
| (3, 11) | **0.003** | **0.003** |
| (4, 11) | 0.0062 | 0.0062 |
| (5, 11) | 0.0173 | 0.0173 |
| (6, 11) | 0.0156 | 0.0156 |
| (7, 11) | 0.0139 | 0.0139 |
| (8, 11) | 0.0136 | 0.0136 |
| (9, 11) | 0.0109 | 0.0109 |

Table 4.1: MSE values for the Simulated image.

The MSE is minimum if $(r_1, r_2)$ is selected to be $(3, 11)$ for the registration technique utilizing both $L_1$-norm and $L_2$-norm, according to Table 4.1. Thus, in order to perform the registration procedure and produce the registered images, we employ these values of $r_1$ and $r_2$. The residual images for the $L_1$ and $L_2$-norm registration processes are then produced. Figure 4.5 shows both the registered and residual images.

In the course of creating the registered images, we also estimate the parameters of the image transformation in the moved image. As we have considered translation, rotation, and scaling as the image transformations, we provide the estimates of the translation parameters $(\ell_x, \ell_y)$, the rotation angle $(\theta)$, and the zooming factor $(s)$ for the selected $(r_1, r_2)$ values for the registration technique utilizing both $L_1$-norm and $L_2$-norm, along with their true values, in Table 4.2.

| | True value | Estimated value ($L_1$-norm) | Estimated value ($L_2$-norm) |
|---|---|---|---|
| $(\ell_x, \ell_y)$ | $(-2, -2)/128$ | $(-1.9586, -1.4505)/128$ | $(-1.9586, -1.4505)/128$ |
| $\theta$ (in degrees) | 10 | 8.5927 | 8.5927 |
| $s$ | 1.14 | 1.1447 | 1.1447 |

Table 4.2: Estimates and the true values of the translation parameter, the rotation angle and the zooming factor for the 'simulated' image.

**Performance on a medical image:**

To extend our horizons on the applicability of the proposed method, we then examine our method on a medical X-ray image. We select an X-ray image of an ankle of resolution $128 \times 128$ and create the moved version of the said image. First, we rotate the image by 10 degrees in the clock-wise direction, and then we zoom in on the central region with a resolution of $112 \times 112$ to make it $128 \times 128$. Finally, we create the moved image by shifting the rotated and zoomed image two pixel lengths upward and two pixel lengths to the right. We present both the reference and moved image in Figure 4.6. The black border in Figure 4.6(b) is generated by the software while incorporating the translation component.

Next, we detect the edge pixels of the reference and the moved image using the approach described in Section 4.2.2. In order to further shorten the runtime of the feature mapping procedure, we, as discussed earlier, consider the 20% of the feature points of the moved image that are closest to the center and execute the feature mapping on those feature points. We follow the processes outlined in Section 4.2.5 to determine the parameters $r_1$ and $r_2$ for feature mapping. With this objective in mind, we consider 13 values of $r_2$ ranging from 4 to 16, and for each $r_2$, we consider $r_1$ to be between 1 and the value of $r_2$. We execute the procedure for each such choice of $r_1$ and $r_2$ and generate the registered version of the moved image. We then compute the mean squared error (MSE) between the intensities of the reference image and the registered version of the moved image, corresponding to two similarity measures, MAD and MSD, corresponding to each

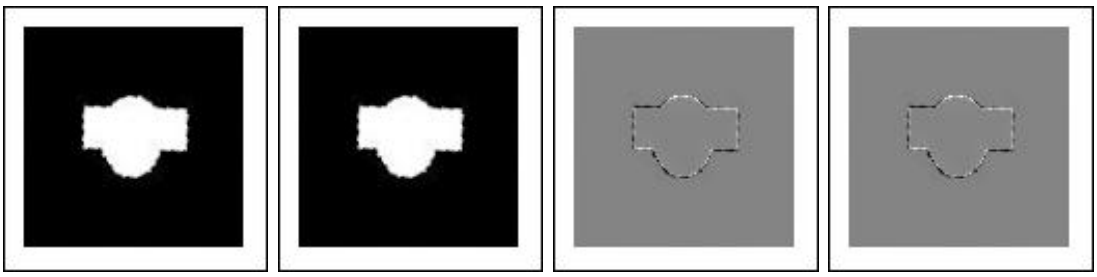


Figure 4.5: From left to right: (a) registered image using $L_1$-norm, (b) registered image using $L_2$-norm, (c) residual image using $L_1$-norm, (d) residual image using $L_2$-norm.

Figure 4.6: The ankle image: (a) reference (left), (b) moved (right).

specific $r_1$ and $r_2$. Table 4.3 shows partial results containing the instances where the MSE of the above exercise is minimum.

| Mean Squared Error | | |
|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm |
| (8, 16) | 0.0112 | 0.0141 |
| (9, 16) | 0.0085 | 0.009 |
| (10, 16) | 0.0064 | 0.0059 |
| (11, 16) | 0.0058 | 0.0039 |
| (12, 16) | 0.0055 | 0.0033 |
| (13, 16) | **0.0053** | **0.0032** |
| (14, 16) | 0.0065 | 0.0044 |
| (15, 16) | 0.0077 | 0.0068 |
| (16, 16) | 0.0083 | 0.0069 |

Table 4.3: MSE values for the 'Ankle' image.

According to Table 4.3, the MSE is minimum if $(r_1, r_2)$ is chosen to be $(13, 16)$ for the registration procedure using both $L_1$-norm and $L_2$-norm. Therefore, we use these values of $r_1$ and $r_2$ for the appropriate operations to execute the registration technique. Next, we create the residual images for the registration process involving $L_1$-norm and $L_2$-norm. Both registered and residual images are presented side by side in Figure 4.7.

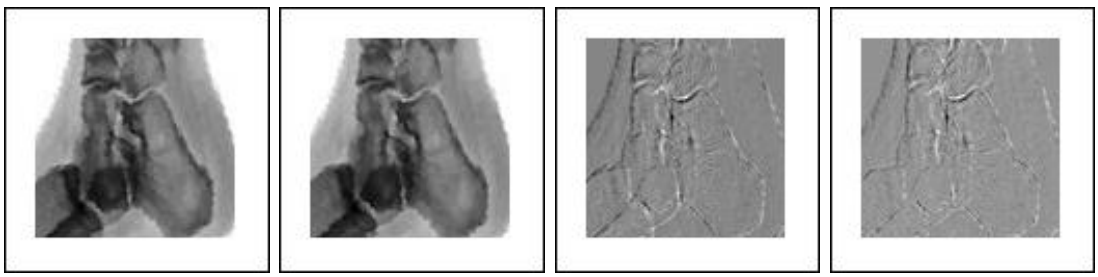


Figure 4.7: From left to right: (a) registered image using $L_1$-norm, (b) registered image using $L_2$-norm, (c) residual image using $L_1$-norm, (d) residual image using $L_2$-norm.

Next, we also provide the true values and estimates of the translation parameter $(\ell_x, \ell_y)$, the rotation angle $(\theta)$, and the zooming factor $(s)$ for the selected $(r_1, r_2)$ values for the registration technique using both $L_1$-norm and $L_2$-norm, in Table 4.4.

| | True value | Estimated value ($L_1$-norm) | Estimated value ($L_2$-norm) |
|---|---|---|---|
| $(\ell_x, \ell_y)$ | $(-2, -2)/128$ | $(-2.2613, -2.4591)/128$ | $(-2.1165, -2.3172)/128$ |
| $\theta$ (in degrees) | 10 | 9.8733 | 9.5796 |
| $s$ | 1.14 | 1.1774 | 1.1691 |

Table 4.4: Estimates and the true values of the translation parameter, the rotation angle and the zooming factor for the 'Ankle' image.

**Performance on a satellite image:**

We conclude this section by applying our registration method to a satellite image. We consider a satellite image of the Aral Sea of resolution $128 \times 128$ as the reference image. To create the moved version of the image, we then rotate the image by 10 degrees clockwise, zoom in on the central region with a resolution of $112 \times 112$, and make it $128 \times 128$. To conclude the construction of the moved image, we shift the rotated and zoomed image two rows upwards and two columns to the right. Figure 4.8 shows both the reference and moved image. In Figure 4.8(b), we see a black border that has been formed by the

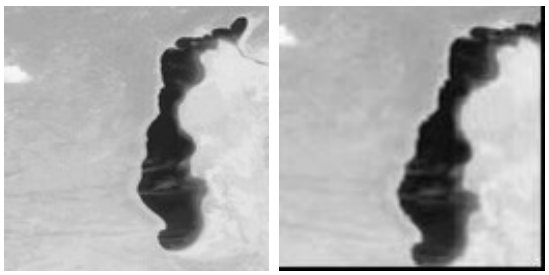


Figure 4.8: The images of Aral sea: (a) reference (left), (b) moved (right).

software when including the translation component.

We then use the edge detection method outlined in Section 4.2.2 to detect the edges of the reference and the moved images. As previously mentioned, we consider 20% of the feature points of the moved image that are closest to the center and do the feature mapping on those feature points in order to further reduce the runtime of the feature mapping algorithm. We determine $r_1$ and $r_2$ for feature mapping using the procedures described in Section 4.2.5. To this end, we consider 13 values of $r_2$ ranging from 4 to 16, and we consider $r_1$ to be between 1 and the value of $r_2$ for each $r_2$. We carry out the process for each pair of these $r_1$ and $r_2$ values and generate the registered version of the moved image. We next calculate the mean squared error (MSE) between the intensities of the registered version of the moved image and the reference image, corresponding to two similarity measures: MAD and MSD, as before. We record these values for each unique $r_1$ and $r_2$. The findings of the aforementioned procedure are displayed partially in Table 4.5. The selection of entries in Table 4.5 is such that it contains those choices of $(r_1, r_2)$

where the MSE of the process is minimum. Table 4.5 shows that if $(r_1, r_2)$ is taken to be

| Mean Squared Error | | |
|---|---|---|
| $(r_1, r_2)$ | $L_1$ norm | $L_2$ norm |
| (4, 16) | 0.0146 | 0.0148 |
| (5, 16) | 0.0107 | 0.0121 |
| (6, 16) | 0.0123 | 0.0119 |
| (7, 16) | 0.0059 | 0.0059 |
| (8, 16) | **0.0028** | 0.0042 |
| (9, 16) | 0.003 | **0.0038** |
| (10, 16) | 0.0137 | 0.0121 |
| (11, 16) | 0.0195 | 0.0098 |
| (12, 16) | 0.0204 | 0.0101 |
| (13, 16) | 0.0166 | 0.0153 |
| (14, 16) | 0.0256 | 0.015 |
| (15, 16) | 0.0267 | 0.019 |
| (16, 16) | 0.028 | 0.0226 |

Table 4.5: MSE values for the Aral Sea image.

$(8, 16)$, the MSE is minimum for the registration operation using $L_1$-norm, and if $(r_1, r_2)$ is chosen to be $(9, 16)$, the minimum is reached for the registration procedure while using $L_2$-norm. Thus, we execute the registration technique and produce the registered images using these values of $r_1$ and $r_2$ for the corresponding procedures. Next, we generate the residual images for the registration process involving $L_1$-norm and $L_2$-norm in the same manner as before. Figure 4.9 presents the registered and residual images.

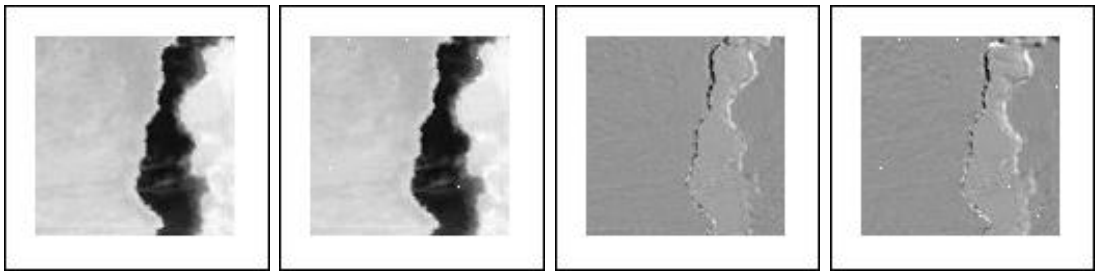


Figure 4.9: From left to right: (a) registered image using $L_1$-norm, (b) registered image using $L_2$-norm, (c) residual image using $L_1$-norm, (d) residual image using $L_2$-norm.

Proceeding in the same way as earlier, we present the true values and estimates of the translation parameter $(\ell_x, \ell_y)$, the rotation angle $(\theta)$, and the zooming factor $(s)$ for the selected $(r_1, r_2)$ values for the registration approach using both $L_1$-norm and $L_2$-norm, in Table 4.6.

**A comparative study of the registration error against various noise levels:**

| | True value | Estimated value ($L_1$-norm) | Estimated value ($L_2$-norm) |
|---|---|---|---|
| $(\ell_x, \ell_y)$ | $(-2, -2)/128$ | $(-1.3317, -2.2627)/128$ | $(-0.7922, -2.1374)/128$ |
| $\theta$ (in degrees) | 10 | 9.0140 | 9.1523 |
| $s$ | 1.14 | 1.1453 | 1.1208 |

Table 4.6: Estimates and the true values of the translation parameter, the rotation angle, and the zooming factor for the 'Aral sea' image.

We wish to study the Mean Squared Error (MSE), capturing the registration error under various noise levels. For that purpose, we take the simulated image and its transformed, i.e, translated, rotated, and zoomed version, shown in Figure 4.4 and add various levels of noise to their intensity values. In this regard, we add additive and spatially independent noise that follows $N(0, \sigma^2)$ distribution. We choose four different levels of $\sigma$ for our analysis which are 0.02, 0.04, 0.06, and 0.08, respectively. We then execute our registration procedure under both $L_1$-norm and $L_2$-norm for the two images: reference and transformed, and note down the registration error in terms of mean squared error (MSE). We present a graphical representation of the comparative study below:

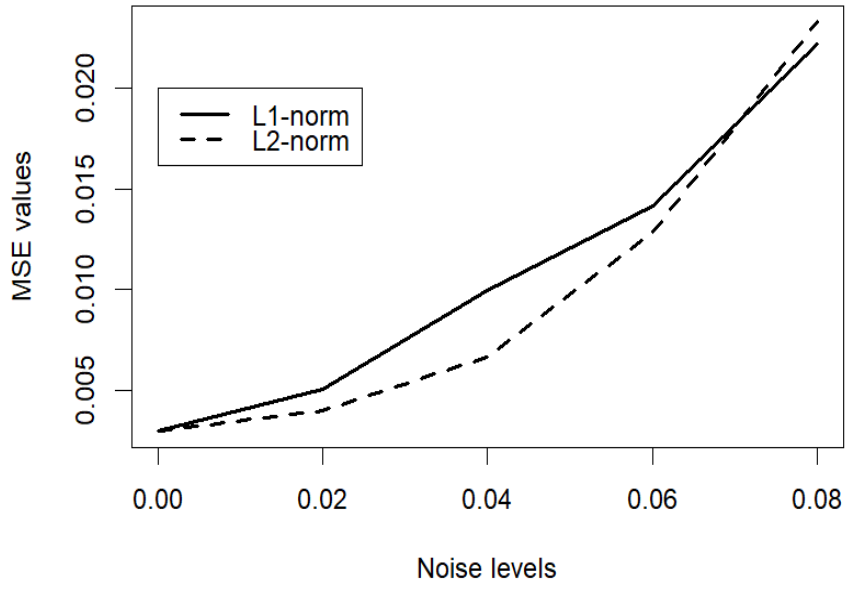



Figure 4.10: Plot of MSE capturing the registration error against various levels of noise.

We see that the registration error keeps increasing with the noise level, which is expected. Please note that our proposed method will not work for noise levels so heavy that the edge detection step cannot perform well.

## 4.5 Comparison with State-of-the-art Methods

We now compare the proposed method with a number of well-known state-of-the-art techniques, all of which are based on specific feature matching algorithms. Popular feature-based image registration methods include Scale Invariant Feature Transform (SIFT) (Lowe, 2004), Speeded Up Robust Features (SURF) (Bay et al., 2008), Oriented Fast and Rotated Brief (ORB) (Rublee et al., 2011), KAZE (Alcantarilla et al., 2012), Accelerated-KAZE (AKAZE) (Alcantarilla and Solutions, 2011), and Binary Robust Invariant Scalable points (BRISK) (Leutenegger et al., 2011), and so forth. The primary operating principles of these techniques are the identification and description of characteristics/features present in both the moved and reference images. After that, registered images are created by utilizing an algorithm, such as RANSAC (Derpanis, 2010), to determine the relations between these characteristics in the two images. In this chapter, we select ORB, AKAZE, and BRISK as the competitors of the proposed method. This is a reasonable choice because they are feature-based methods and can register images when translation, rotation, and scaling are involved. MSE and normalized mutual information (NMI) (Cover, 1999) are taken into consideration as the performance evaluation metric in order to assess the algorithm's performance in relation to the existing techniques. These metrics, which quantify the degree of similarity between two images, are frequently used in the literature on image registration and are useful in evaluating registration performance. Mean squared error (MSE), a similarity metric, is evaluated on the basis of the difference in gray level between two images. The expression for MSE is as follows:

$$\mathrm{MSE}(I_R, \widehat{I}_M) = \sum_{(u,v)\in\Omega} (\widehat{I}_M(u,v) - I_R(u,v))^2,$$

where $\widehat{I}_M(u,v)$ and $I_R(u,v)$ represent the image intensity values at the $(u,v)$-th pixel coordinate for the registered and reference images, respectively. MI measures the mutual dependence between two images. Intuitively, it measures the reduced uncertainty of one image in the presence of the information of another image. Therefore, a larger value of MI is preferable for a greater degree of similarity. Mathematically, MI between reference and registered image can be expressed as

$$MI(I_R, \widehat{I}_M) = H(I_R) + H(\widehat{I}_M) - H(I_R, \widehat{I}_M),$$

where $H(I_R)$ and $H(\widehat{I}_M)$ are the entropies of the reference image and registered image, respectively, $H(I_R, \widehat{I}_M)$ is the joint entropy of $I_R$ and $\widehat{I}_R$. The normalized mutual infor-

mation (NMI) is calculated as

$$NMI(I_R, \widehat{I}_M) = \frac{2MI(I_R, \widehat{I}_M)}{H(I_R) + H(\widehat{I}_M)}.$$

We first create the registered version of the moved image by applying these methods to the moved image. Following that, we calculate the MSE and NMI between the reference image and the registered images acquired using various methods and present a table for comparison. To aid visual comparison between these different methods, we then display the reference image, the moved image, and the registered images side by side.

The proposed image registration method involves a feature mapping step that is executed with two different similarity metrics, viz., mean squared deviation (MSD) and mean absolute deviation (MAD), giving rise to two parallel approaches, viz., the registration approach using $L_2$-norm and the registration approach using $L_1$-norm, respectively. Therefore, as far as our top-performing method is concerned, it is not unique for all the images used here. In the case of the simulated image, the performances of both of our suggested approaches are similar. Hence, we have used the approach involving $L_1$-norm while comparing it with other state-of-the-art methods. The registration approach involving $L_2$-norm and the registration approach involving $L_1$-norm are taken into consideration for the images 'Ankle' and 'Aral Sea', respectively, as our top performing methods for similar reasons.

We include MSE and NMI values that were calculated for several images, including the simulated image, the X-ray image of an ankle, and the satellite image of the "Aral Sea" in Table 4.7. MSE part of Table 4.7 shows that, for the simulated image, our proposed method is performing well while BRISK fails to even generate the registered image due to the sparseness of the image's point features, as defined by the feature definitions for these methods. ORB and AKAZE, as compared to the proposed method, are performing poorly in the case of the simulated image. Our proposed method performs way better than other approaches in the case of the 'ankle' image, while the BRISK method becomes a distant second. We observe a similar phenomenon in the case of the 'Aral Sea' image. Though in the case of the 'Aral Sea' image, BRISK comes pretty close to our proposed method as far as MSE is concerned. From NMI part of Table 4.7, we see that the performance of our method remains superior in the case of simulated image as far as NMI is concerned. But in the case of the other two images, our method is comparable to BRISK.

For the purpose of comparing these approaches, in addition to Table 4.7, we also provide the registered images produced by various methods (the proposed and others), which are available in Figure 4.11. Each row in Figure 4.11 shows the outcomes that correlate to a certain image. We arrange images in a row in the following order: the

| | Mean Squared Error (MSE) | | | |
|---|---|---|---|---|
| Image type | Proposed Method | ORB | AKAZE | BRISK |
| Simulated | **0.0030** | 0.1423 | 0.0630 | NA |
| Ankle | **0.0032** | 0.0558 | 0.0405 | 0.0220 |
| Aral Sea | **0.0028** | 0.0086 | 0.0253 | 0.0036 |
| | Normalized Mutual Information (NMI) | | | |
| Image type | Proposed Method | ORB | AKAZE | BRISK |
| Simulated | **1.1356** | 1.0084 | 1.0123 | NA |
| Ankle | 1.2466 | 1.0869 | 1.0798 | **1.2686** |
| Aral Sea | 1.2358 | 1.2407 | 1.1375 | **1.3251** |

Table 4.7: Mean squared error (MSE) and normalized mutual information (NMI) for various methods and images.

reference image, the zoomed image, the registered images under the proposed method, ORB, AKAZE, and BRISK, respectively.

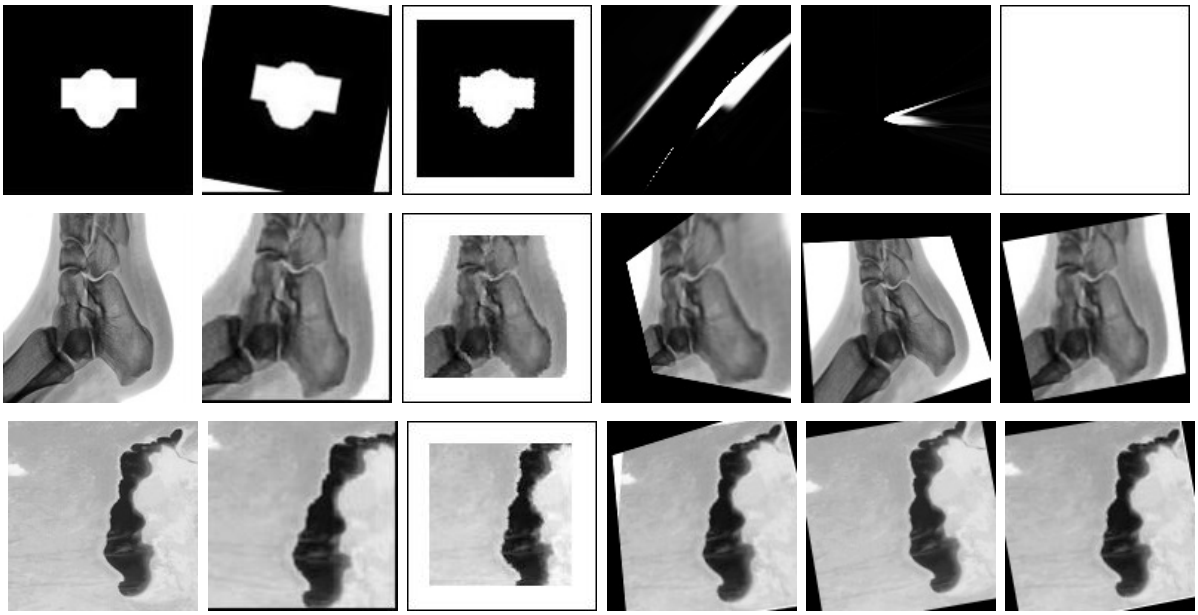


Figure 4.11: The reference image, the moved image, and the registered images under the proposed method, ORB, AKAZE, and BRISK.

Figure 4.11 shows that, when it comes to the simulated image (see first row of Figure 4.11), our proposed technique wins convincingly, while the other methods are unable to even provide a meaningful registered image at all. In the case of the simulated image, one of the competing methods, viz., BRISK, could not even generate the registered image. Second and third rows of Figure 4.11 correspond to the reference image, moved image, and registered images by the proposed and competing methods for the 'Ankle' image and 'Aral Sea' image, respectively. Compared to the other methods, our approach performs

reasonably in the cases of the 'Ankle' and 'Aral Sea' images. For these images, the registered images generated by the competing methods seem a bit distorted at first glance. So to have a clearer idea about the performance of these methods, we create the residual images for the proposed and competing methods and display them in Figure 4.12. Each row in Figure 4.12 corresponds to the residual images for a specific image. In the first row, we see the residual images corresponding to the simulated image, and two subsequent rows display the residual images corresponding to the 'Ankle' image and the 'Aral Sea' image, respectively. In each row, images are arranged in the following order: the residual images under the proposed method, ORB, AKAZE, and BRISK, respectively.

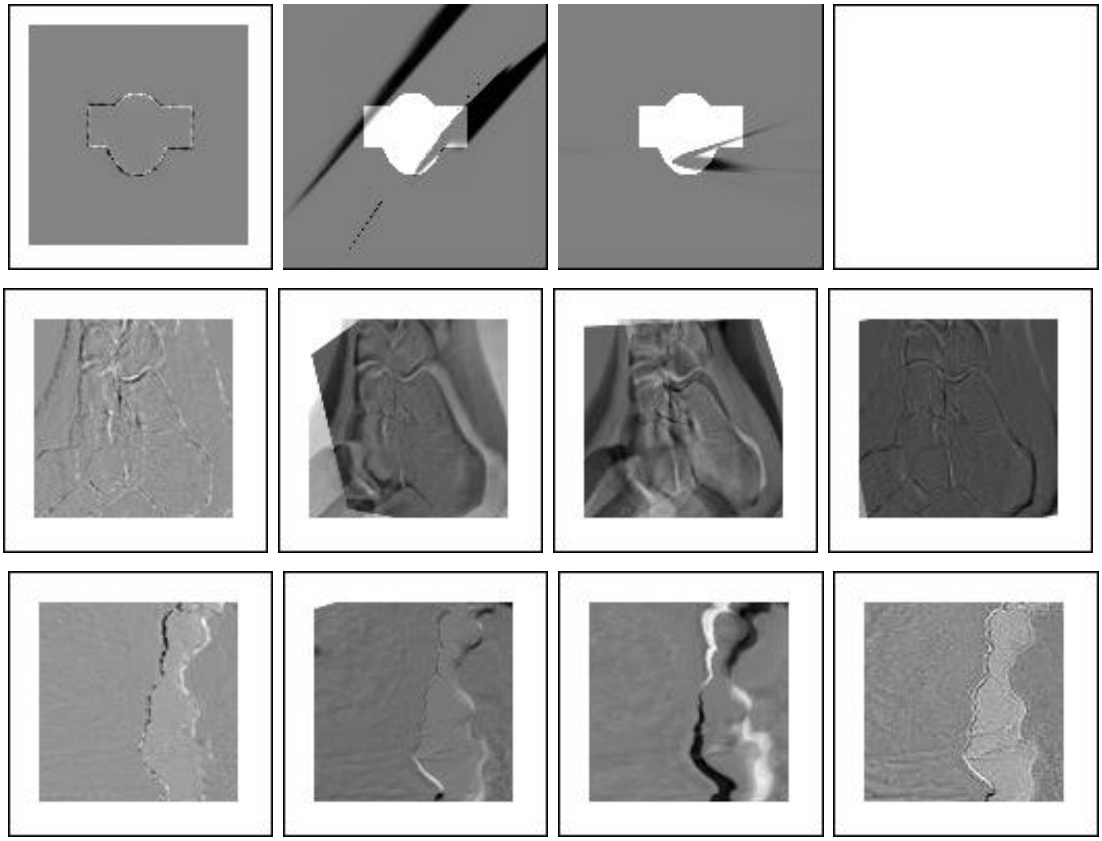


Figure 4.12: The residual images under the proposed method, ORB, AKAZE, and BRISK.

It becomes evident from Figure 4.12 that our proposed method is working way better than the other competing methods considered here. In the case of a simulated image, there is no residual image for the BRISK method, as it could not generate the registered image. Also, it is quite difficult to interpret the residual images for the methods ORB and AKAZE in the case of simulated images, as the registered images generated by them were not even close to the reference image. On the contrary, our proposed method works reasonably well in this case. Residual images for 'Ankle' and 'Aral Sea' images also justify our claim of superiority over other competing methods.

### 4.5.1 Additional simulation studies

Next, we check the performance of the proposed method with other types of state-of-the-art image registration techniques. We choose a deep-learning-based diffeomorphic image registration method (DLBDR) (Mok and Chung, 2020), and an optical flow-based image registration method, which is a duality-based algorithm for total variation, TV-L1 (Pock et al., 2007). Note that both the chosen methods are flexible enough to incorporate non-linear image registration. Hence, the comparison with the proposed method is not particularly an apple-to-apple comparison. Nonetheless, this comparison is still practically useful, as these two competing methods are very popular.

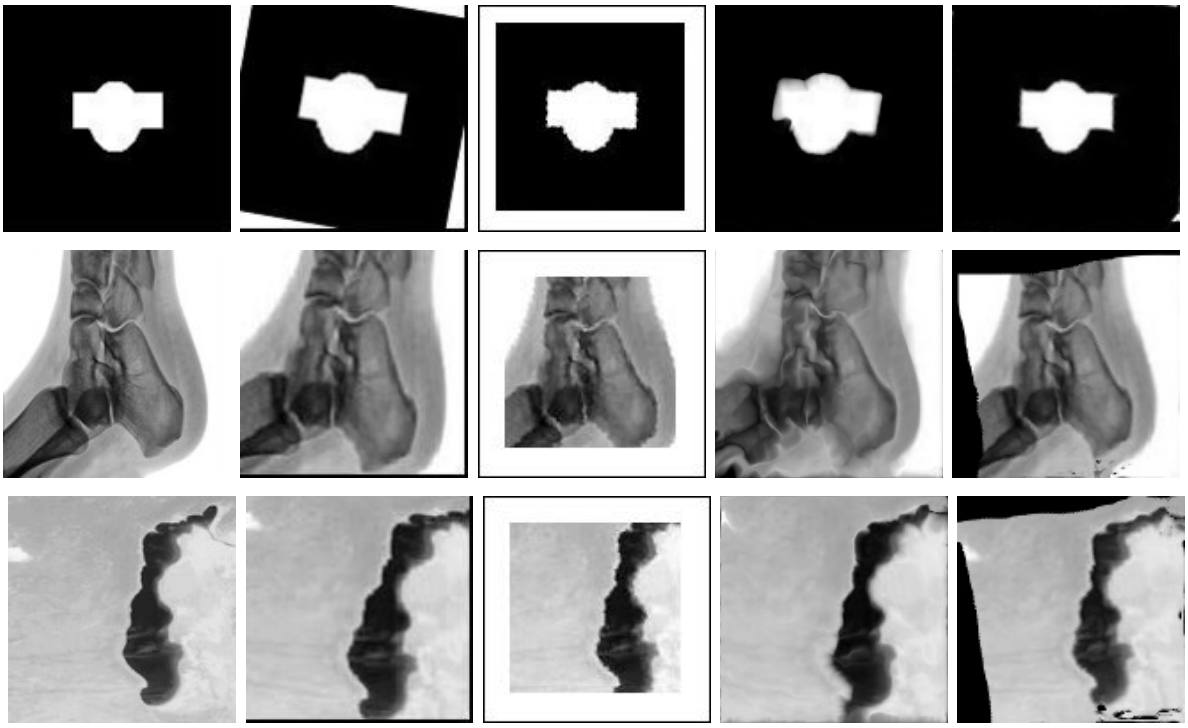


Figure 4.13: The reference image, the moved image, and the registered images under the proposed method, deep-learning based diffeomorphic image registration, and Optical Flow-TVL1.

For the implementation of DLBDR and Optical Flow-TVL1, we use the pre-trained model provided by (Mok and Chung, 2020), and the available computer codes of the method proposed by (Pock et al., 2007). From Figure 4.13, it is clear that the DLBDR method (Mok and Chung, 2020) blurs the sharp edges, and it has limited ability to shrink the zoomed versions. The Optical flow-based TV-L1 method (Pock et al., 2007) performs better than DLBDR but is still unable to match the performance of the proposed method. Table 4.8 shows that the proposed method outperforms DLBDR and optical flow-based TV-L1 by big margins in terms of MSE, while the NMI values are comparable.

| | Mean Squared Error (MSE) | | |
|---|---|---|---|
| Image type | Proposed Method | DLBDR | Optical Flow-TVL1 |
| Simulated | **0.0030** | 0.0191 | 0.0065 |
| Ankle | **0.0032** | 0.0205 | 0.0056 |
| Aral Sea | **0.0028** | 0.0229 | 0.0088 |
| | Normalized Mutual Information (NMI) | | |
| Image type | Proposed Method | DLBDR | Optical Flow-TVL1 |
| Simulated | 1.1356 | **1.3526** | 1.1708 |
| Ankle | 1.2466 | 1.2037 | **1.3286** |
| Aral Sea | 1.2358 | 1.2820 | **1.3056** |

Table 4.8: Mean squared error (MSE) and normalized mutual information (NMI) for the proposed method, DLBDR, and Optical Flow-TVL1 on various images.

## 4.6 Discussions

This chapter proposes a feature-based image registration approach that registers a translated, rotated, and zoomed image with its original version. The main working principle of this method is to detect the edges as the primary image feature and explain them through a mathematical model. One major advantage of this method is that it works well consistently in almost all non-trivial images, even when the number of point features is scarce, as compared to many other state-of-the-art methods available in the literature that fail to even generate the registered images in such situations. Another advantage of our method is that, along with generating the registered images, it also estimates the parameters of the image transformations, viz., translation, rotation, and zooming/scaling, which are statistically consistent. One disadvantage is that this approach is computationally intensive. Because we consider edges to be the major feature here, this technique will take longer for images with abundant edge points. This scenario would significantly improve if we chose something else as the major feature, which is limited in quantity and present in practically all non-trivial images.

The advantages of the proposed method are: (i) It works well consistently in almost all non-trivial images, even when the number of point features is scarce. (ii) Along with generating the registered images, it also estimates the parameters of the image transformations, viz., translation, rotation, and zooming/scaling, which are statistically consistent. One restriction in the proof of theoretical justification is that it does not consider a large amount of zooming, similar to the methods in the previous chapters. Another disadvantage is that this approach is computationally intensive. This method will take a longer time for images with abundant edge points. This scenario would significantly improve if we chose something else as the major feature, which is limited in quantity and

present in practically all non-trivial images.

In this chapter, we assume that the centers for rotation and zooming are the same, and that is the image center itself, which is known. A much more complex and challenging problem to deal with would be to register the images under the image transformations of translation, rotation, and zooming after relaxing the aforementioned assumptions. Much future research is needed in this direction.

# Chapter 5

# Concluding Remarks and Future Directions

In this dissertation, we present a number of novel image registration algorithms to solve various problems of image registration involving different image transformations. This dissertation's primary strengths are its original concepts, ease of use, and broad applicability to real-world image data. There are some limitations to the proposed methods. We discuss a few of those limitations and the corresponding remedial steps that can be incorporated into future research in Section 5.1. We dedicate Section 5.2 to make the readers aware of possible research areas other than but related to image registration.

## 5.1 Limitations and Future Scopes

In Chapter 2, we introduce an intensity-based image registration technique for the zooming transformation. This method is computationally complex as the similarity matching algorithm is applied to each pixel location in the moved image. We try to address this shortcoming in Chapter 3 by resorting to a feature-based approach for the same problem, where we take edge points as the primary feature for the approach. In Chapter 4, we solve a more complex registration problem involving translation, rotation, and zooming as the image transformations, opting for a feature-based approach. Here, we also take edge points as a primary feature, as in the previous chapter. Though using edges as feature points instead of an intensity-based approach reduces the run-time of the procedure substantially, it still takes a lot of time when the number of edge points in the image is abundant. In order to further reduce the computational complexity of the discussed approaches, we should look for some features of the images that are readily available and are few in number in the future. In Chapter 4, we use a model to describe the mapped

features of the reference and the moved image. The main assumption of the model is that the centers for rotation and zooming are the same, and that is the image center itself, which is known. It will be interesting to relax this assumption in the future and get the estimate of the image center from the model itself.

The methods discussed in this dissertation are not capable of handling anomalies in the image data. Nowadays, several satellites are orbiting the earth continuously and taking several pictures of the same landscape at different times, from different viewpoints, etc. Figure 5.1 demonstrates one such instance where two images of the same landscape captured by some satellite appear different only because of the fact that in one image there is cloud cover, whereas in another image there is no cloud cover. If we want to register such images, then we need to devise more sophisticated approaches that will be able to ignore the anomalies and register the images.

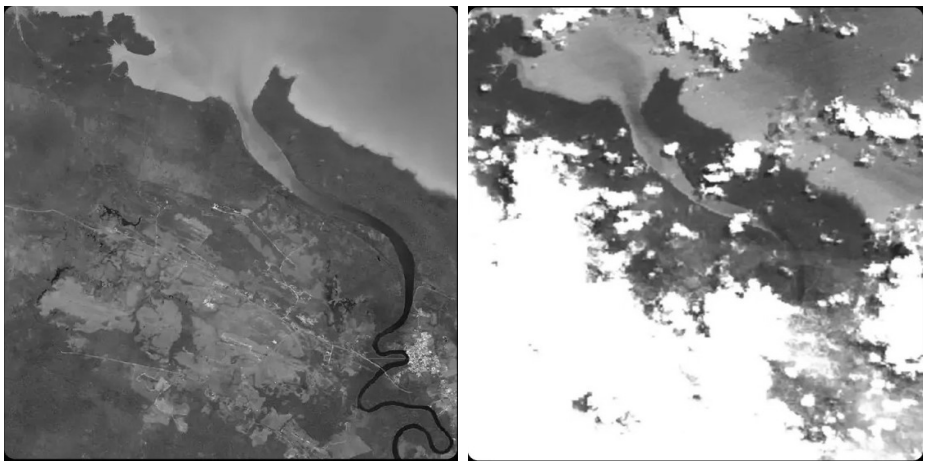


Figure 5.1: A demonstration of the feature mapping problem in the presence of anomalies such as clouds.

The proposed methods use simpler, exhaustive approaches and take much longer to execute than the competing methods. The focus of this dissertation is to solve the problem under the jump regression framework, and it does not focus on reducing the computational complexity. The method proposed in Chapter 2 takes around 5-8 minutes, and the methods proposed in Chapters 3 and 4 take around 6-10 seconds to execute for a given choice of $(r_1, r_2)$, whereas the competing methods take a little less than a second. More computationally efficient codes of the proposed methods will reduce the computation times, but they will still be longer than the competing methods. Much future research is needed in this direction.

## 5.2 Possible Future Research Areas

As we have discussed in the previous section, there has been a lot of interest in using satellite photos to track the Earth's surface over time. The USGS and NASA deployed nine satellites as part of the well-known LANDSAT Project, which had its start in 1972 and aims to gradually gather photographs of the Earth's surface. One can keep an eye on the Earth's surface and the surrounding natural resources by comparing the series of images. As a result, keeping an eye on the size and structure of these satellite image types has become crucial.

For two reasons, though, monitoring this kind of image data is a difficult task. First of all, noise is present in discontinuous image surfaces. More significantly, and secondly, actual images of the same object taken with various sensors or at various times are frequently not geometrically aligned. It is clear that improper image alignment during image monitoring (e.g., Roy and Mukherjee (2024a, 2025)) results in serious errors in inference. Such problems of image monitoring use image registration as the pre-processing step. Hence, there is a scope to use the registration techniques discussed here as an intermediate step to image monitoring and to develop sophisticated cutting-edge methods of image monitoring.

Monitoring of medical images, such as brain MRIs of various subjects, is an even more challenging problem, as it involves non-linear image registration. However, the proposed methods in this dissertation can be developed further to accomplish these goals under the jump regression framework.

# Appendix A

# Appendix with Additional Details

## A.1 Proof of the Stated Proposition in Chapter 2:

### A.1.1 Proof of Proposition 2.3.1:

Fix any small $\varepsilon > 0$. Using Markov's inequality,

$$P\left[\frac{1}{n^2}\{(x_2 - x)^2 + (y_2 - y)^2\} > \varepsilon\right] \leq \frac{1}{\varepsilon} E\left[\frac{1}{n^2}\{(x_2 - x)^2 + (y_2 - y)^2\}\right].$$

Therefore,

$$\frac{E(x_2 - x)^2}{n^2\varepsilon} = \frac{E[x_2 - E(x_2) + E(x_2) - x]^2}{n^2\varepsilon} = \frac{V(x_2) + (E(x_2) - x)^2}{n^2\varepsilon}. \tag{A.1.1}$$

Next, we find upper bounds for $V(x_2)$ and $(E(x_2) - x)^2$. We first consider two scenarios where $V(x_2)$ and $(E(x_2) - x)^2$ are extreme. Note that we consider suitable integer values for $x_2$ in $[x_1 - r_2, x_1 + r_2]$. Therefore, $(x_1 - r_2)$, $(x_1 - r_2 + 1)$, ..., $(x_1 - 1)$, $x_1$, $(x_1 + 1)$, ..., $(x_1 + r_2 - 1)$, and $(x_1 + r_2)$ are possible realizations of $x_2$ in the vicinity of width $r_2$, i.e., there are $(2r_2 + 1)$ number of possible values of $x_2$. The distribution of $(2r_2 + 1)$ realizations in the first extreme case, i.e., when $V(x_2)$ would be the largest, is as follows. One realization at $x_1$, $r_2$ realizations at $(x_1 + r_2)$, and $r_2$ realizations at $(x_1 - r_2)$ are all present. In this case, $E(x_2) = x_1$ and $V(x_2) = \frac{2{r_2}^3}{2r_2+1}$. In the second extreme case, i.e., when $(E(x_2) - x)^2$ would be the largest is when all $(2r_2 + 1)$ realizations of $x_2$ are at $x_1 + r_2$. Therefore, in this case, $E(x_2) = \frac{(2r_2+1)(x_1+r_2)}{(2r_2+1)} = (x_1 + r_2)$. Hence, from Eqn. (A.1.1), under the given conditions,

$$\frac{V(x_2) + (E(x_2) - x)^2}{n^2\varepsilon} \leq \frac{2{r_2}^3}{(2r_2 + 1)n^2\varepsilon} + \frac{(x_1 + r_2 - (x_1 - c_1))^2}{n^2\varepsilon} \to 0.$$

A similar result can be proved for $E(y_2 - y)^2/(n^2\varepsilon)$. Hence, the proposition is proved. ■

## A.2 Proofs of the Stated Propositions in Chapter 3:

### A.2.1 Proof of Proposition 3.3.1:

Fix any small $\varepsilon > 0$. Using Markov's inequality, $P[\frac{1}{n^2}\{(u'-u)^2 + (v'-v)^2\} > \varepsilon] \leq \frac{1}{\varepsilon}E[\frac{1}{n^2}\{(u'-u)^2 + (v'-v)^2\}]$. Therefore,

$$\frac{E(u'-u)^2}{n^2\varepsilon} = \frac{E[u' - E(u') + E(u') - u]^2}{n^2\varepsilon} = \frac{V(u') + (E(u') - u)^2}{n^2\varepsilon}. \tag{A.2.1}$$

Next, we find upper bounds for $V(u')$ and $(E(u') - u)^2$. We first consider two scenarios where $V(u')$ and $(E(u') - u)^2$ are extreme. Note that we consider suitable integer values for $u'$ in $[U - r_2, U + r_2]$. Therefore, $(U - r_2)$, $(U - r_2 + 1)$, ..., $(U - 1)$, $U$, $(U + 1)$, ..., $(U + r_2 - 1)$, and $(U + r_2)$ are possible realizations of $u'$ in the vicinity of width $r_2$, i.e., there are $(2r_2 + 1)$ number of possible values of $u'$. The distribution of $(2r_2 + 1)$ realizations in the first extreme case, i.e., when $V(u')$ would be the largest, is as follows. One realization at $U$, $r_2$ realizations at $(U + r_2)$, and $r_2$ realizations at $(U - r_2)$ are all present. In this case, $E(u') = U$ and $V(u') = \frac{2{r_2}^3}{2r_2+1}$. In the second extreme case, i.e., when $(E(u') - u)^2$ would be the largest is when all $(2r_2 + 1)$ realizations of $u'$ are at $U + r_2$. Therefore, in this case, $E(u') = \frac{(2r_2+1)(U+r_2)}{(2r_2+1)} = (U + r_2)$. Hence, from Eqn. A.2.1, under the given conditions,

$$\frac{V(u') + (E(u') - u)^2}{n^2\varepsilon} \leq \frac{2{r_2}^3}{(2r_2+1)n^2\varepsilon} + \frac{(U + r_2 - (U - c_1))^2}{n^2\varepsilon} \to 0. \tag{A.2.2}$$

A similar result can be proved for $\frac{E(v'-v)^2}{n^2\varepsilon}$. Hence, the proposition is proved. ■

### A.2.2 Proof of Proposition 3.3.2:

We first provide a sketch of the proof of the consistencies of the estimators $\widehat{s}$ and $\widehat{\ell_x^*}$ of the parameters $s$ and $\ell_x^*$, respectively, obtained from the first equation of the model 3.2.10.

To estimate $s$ and $\ell_x^*$ from the equation mentioned earlier, we minimize $S^2 = \sum_{i=1}^{N}(U_i - \ell_x^* - su_i')^2$ w.r.t. $s$ and $\ell_x^*$. The desired values are obtained by solving the following

simultaneous equations, called the normal equations:

$$\frac{\partial S^2}{\partial \ell_x^*} = 0, \quad \frac{\partial S^2}{\partial s} = 0$$

which give the following equations,

$$\begin{aligned} \sum_{i=1}^{N} \zeta_{U_i} = 0 \quad i.e. \quad \sum_{i=1}^{N} (U_i - \ell_x^* - s u_i') = 0, \\ \sum_{i=1}^{N} u_i' \zeta_{U_i} = 0 \quad i.e. \quad \sum_{i=1}^{N} u_i' (U_i - \ell_x^* - s u_i') = 0. \end{aligned} \tag{A.2.3}$$

Solving the above equations, we get the estimates of $s$ and $\ell_x^*$ as

$$\begin{aligned} \widehat{s} &= \frac{\sum_{i=1}^{N} (u_i' - \bar{u}')(U_i - \bar{U})}{\sum_{i=1}^{N} (u_i' - \bar{u}')^2}, \\ \widehat{\ell_x^*} &= \frac{1}{N} \sum_{i=1}^{N} U_i - \widehat{s} \frac{1}{N} \sum_{i=1}^{N} u_i'. \end{aligned} \tag{A.2.4}$$

Now, to check the consistency of $\widehat{s}$ and $\widehat{\ell_x^*}$, we examine their behaviors when $N \to \infty$. First, we check the consistency of $\widehat{s}$.

$$\begin{aligned} \widehat{s} &= \frac{\sum_{i=1}^{N} (u_i' - \bar{u}')(U_i - \bar{U})}{\sum_{i=1}^{N} (u_i' - \bar{u}')^2} \\ &= \frac{\sum_{i=1}^{N} (u_i' - \bar{u}') U_i}{\sum_{i=1}^{N} (u_i' - \bar{u}')^2} \\ &= \frac{\sum_{i=1}^{N} (u_i' - \bar{u}')(\ell_x^* + s u_i' + \zeta_{U_i})}{\sum_{i=1}^{N} (u_i' - \bar{u}')^2} \\ &= s + \frac{\sum_{i=1}^{N} (u_i' - \bar{u}') \zeta_{U_i}}{\sum_{i=1}^{N} (u_i' - \bar{u}')^2} \\ &= s + \frac{cov(\zeta_U, u')}{var(u')} \end{aligned} \tag{A.2.5}$$

Now, from the normal equations, we get $\frac{cov(\zeta_U, u')}{var(u')} = \frac{E(\zeta_U . u') - E(\zeta_U) E(u')}{var(u')} \to 0$ as $N \to \infty$, because $E(\zeta_U) = 0$.

As we take limit on both sides of equation A.2.5 we get, as $N \to \infty$,

$$\widehat{s} \xrightarrow{P} s.$$

Next, we check the consistency of $\widehat{\ell_x^*}$. As we take the limit on both sides of the second

equation of Equation A.2.4 we get, as $N \to \infty$,

$$\lim_{N\to\infty} \widehat{\ell}_x^* = \bar{U} - s\bar{u}' = \ell_x^*.$$

As $\widehat{\ell}_x^* \xrightarrow{P} \ell_x^*$ and $\widehat{s} \xrightarrow{P} s$, then from Slutsky's rule we can say, if $s \neq 1$,

$$\widehat{\ell}_x = \frac{\widehat{\ell}_x^*}{(1-\widehat{s})} \xrightarrow{P} \ell_x.$$

This condition is satisfied as long as there is some amount of zooming.

Similarly, it can be easily shown that $\widehat{\ell}_y \xrightarrow{P} \ell_y$. Hence, the proposition is proved. ■

# A.3 Proofs of the Stated Propositions in Chapter 4:

## A.3.1 Proof of Proposition 4.3.1:

We first fix any small $\varepsilon > 0$ and then, we use Markov's inequality and get, $P[\frac{1}{n^2}\{(u-u')^2 + (v-v')^2\} > \varepsilon] \leq \frac{1}{\varepsilon}E[\frac{1}{n^2}\{(u-u')^2 + (v-v')^2\}]$.

The first part of the right hand side of the aforesaid inequality gives

$$\frac{1}{n^2\varepsilon}E(u-u')^2 = \frac{1}{n^2\varepsilon}E[u - E(u) + E(u) - u']^2 = \frac{1}{n^2\varepsilon}\{V(u) + (E(u)-u')^2\}. \quad \text{(A.3.1)}$$

Next, in order to find the upper bounds for $V(u)$ and $(E(u)-u')^2$, we first consider two scenarios where $V(u)$ and $(E(u)-u')^2$ are extreme. Note that, $u$ has realizations $(U-r_2)$, $(U-r_2+1)$, ..., $(U-1)$, $U$, $(U+1)$, ..., $(U+r_2-1)$, $(U+r_2)$ in the neighborhood of width $r_2$ around itself. The idea of this part of the proof is similar to A.2.1.

The distribution of $(2r_2+1)$ realizations is as follows in the first extreme case. The $r_2$ realizations are located at $(U-r_2)$, the $r_2$ realizations at $(U+r_2)$, and one realization at $U$. In this case, $E(u) = U$ and $V(u) = 2{r_2}^3/(2r_2+1)$. On the other hand, all $(2r_2+1)$ realizations in the second extreme scenario are at $(U+r_2)$. Therefore, in this case, $E(u) = \{(2r_2+1)(U+r_2)\}/(2r_2+1) = (U+r_2)$. Hence, from Eqn. (A.3.1), under the given conditions, as $n \to \infty$,

$$\frac{V(u) + (E(u)-u')^2}{n^2\varepsilon} \leq \frac{2{r_2}^3}{(2r_2+1)n^2\varepsilon} + \frac{(U+r_2-(U-c_1))^2}{n^2\varepsilon} \to 0. \quad \text{(A.3.2)}$$

It is possible to show a similar result for $\frac{E(v-v')^2}{n^2\varepsilon}$. Thus, the proposition is validated. ■

Our next propositions are based on the estimates of the parameters of the transformations, viz., translation, rotation, and zooming.

### A.3.2 Proof of Proposition 4.3.2:

The true angle of rotation,

$$\theta = \cos^{-1}\left[\frac{\{(u_i' - c_x, v_i' - c_y).(U_i^{*'} - c_x, V_i^{*'} - c_y)\}}{|(u_i' - c_x, v_i' - c_y)||(U_i^{*'} - c_x, V_i^{*'} - c_y)|}\right],$$

where, $(u_i', v_i')$ is the edge pixel of the reference image that has been shifted to $(U_i, V_i)$ of the moved image after applying the transformations, viz., translation, rotation, and zooming, $(c_x, c_y)$ is the image center, and $(U_i^{*'}, V_i^{*'})$ are the modified edge pixel of the moved image and this modification is done as per the true edge pixel of the reference image $(u_i', v_i')$'s.

Now, after using the feature mapping algorithm, the $(U_i, V_i)$ of the moved image gets mapped to $(u_i, v_i)$ of the reference image. Then, we compute the angle between $(u_i, v_i)$ of the reference image and the modified pixel $(U_i^*, V_i^*)$ of the moved image as,

$$\theta_i = \cos^{-1}\left[\frac{\{(u_i - c_x, v_i - c_y).(U_i^* - c_x, V_i^* - c_y)\}}{|(u_i - c_x, v_i - c_y)||(U_i^* - c_x, V_i^* - c_y)|}\right], \forall i = 1(1)n_2,$$

where the modified pixel of the moved image has been computed according to Eqn. (4.2.10).

Proposition 1 proves that, $u \xrightarrow{P} u'$ and $v \xrightarrow{P} v'$. As an extension to the aforementioned statement, we can say, $\bar{u} \xrightarrow{P} \bar{u'}$ and $\bar{v} \xrightarrow{P} \bar{v'}$.

So, $U_i^* = (U_i - \bar{U} + \bar{u}) \xrightarrow{P} (U_i - \bar{U} + \bar{u'}) = U_i^{*'}$ and similarly, we can show, $V_i^* \xrightarrow{P} V_i^{*'}$.

Now, using the above-mentioned probability convergences, we can say that,

$$\begin{aligned}\cos\theta_i &= \left[\frac{\{(u_i - c_x, v_i - c_y).(U_i^* - c_x, V_i^* - c_y)\}}{|(u_i - c_x, v_i - c_y)||(U_i^* - c_x, V_i^* - c_y)|}\right] \\ &\xrightarrow{P} \left[\frac{\{(u_i' - c_x, v_i' - c_y).(U_i^{*'} - c_x, V_i^{*'} - c_y)\}}{|(u_i' - c_x, v_i' - c_y)||(U_i^{*'} - c_x, V_i^{*'} - c_y)|}\right] = \cos\theta, \qquad \text{(A.3.3)}\end{aligned}$$

as long as, $u_i \neq c_x$, $v_i \neq c_y$, $U_i^* \neq c_x$, and $V_i^* \neq c_y$.

Next, we consider the random elements $T_i = \cos\theta_i, \forall i = 1(1)n_2$, defined on $[-1, 1]$, and also consider the function $g = cos^{-1} : [-1, 1] \to [0, \pi]$, that has a set of discontinuity points $D_g$ such that $P[T \in D_g] = 0$.

Now, applying the continuous mapping theorem on Eqn. (A.3.3), we have, $\theta_i \xrightarrow{P} \theta$.

It then follows that,

$$\widehat{\theta} = \sum_{i=1}^{n_2} \theta_i / n_2 \xrightarrow{P} \theta.$$

Next, let us assume that, $\widehat{\phi} \xrightarrow{P} \phi.$, as $n \to \infty$. Consider the model described in Eqn. (4.2.11). In this model, the errors $\zeta_U$ and $\zeta_V$ are associated with feature mapping. As we have already discussed, the pixel $(u'_i, v'_i)$ of the reference image has been moved to $(U_i, V_i)$ of the moved image. But, while performing the feature mapping mentioned in Section 4.2.5, this $(U_i, V_i)$ of the moved image gets mapped to $(u_i, v_i)$ of the reference image. This error in mapping, however small, gives rise to the aforementioned errors.

Now, in Eqn. (4.2.11), if we replace the $(u_i, v_i)$ by $(u'_i, v'_i)$, and then the error part of the model will get eliminated only under the condition that $s$, $\ell_x$, $\ell_y$, and $\theta$ are the true values of the transformation parameters, and the corresponding model will be as follows:

$$\begin{aligned}\Big(U_i - c_x\Big) =& s\Big((u'_i - c_x).\ \cos\theta + (v'_i - c_y).\ \sin\theta\Big) + \ell_x \\ \Big(V_i - c_y\Big) =& s\Big( - (u'_i - c_x).\ \sin\theta + (v'_i - c_y).\ \cos\theta\Big) + \ell_y,\end{aligned} \tag{A.3.4}$$

which is the ideal scenario, i.e., the pixels of the moved image are being completely explained by the model involving the pixels of the reference image and the true transformation parameters. This aforementioned model will hold true as $n \to \infty$ under Proposition 1, i.e., when $u \to u'$ and $v \to v'$, as $n \to \infty$ and also under the assumption that the estimates of the transformation parameters, i.e., $\widehat{s}$, $\widehat{\theta}$, $\widehat{\ell_x}$, and $\widehat{\ell_y}$ are consistent. From the above equation, we can find the form of $(u'_i, v'_i)$, the true pixel of the reference image, and it is stated below:

$$\begin{aligned}u'_i =& \frac{U_i - c_x - \ell_x}{s}.\cos\theta - \frac{V_i - c_y - \ell_y}{s}.\sin\theta + c_x \\ v'_i =& \frac{U_i - c_x - \ell_x}{s}.\sin\theta + \frac{V_i - c_y - \ell_y}{s}.\cos\theta + c_y.\end{aligned} \tag{A.3.5}$$

From Proposition 1, we have $u \xrightarrow{P} u'$, and $v \xrightarrow{P} v'$, as $n \to \infty$ and we have already proved that, $\widehat{\theta} \xrightarrow{P} \theta$.

Now, in Eqn.(4.2.11), if we put $\widehat{s}$, $\widehat{\theta}$, $\widehat{\ell_x}$, and $\widehat{\ell_y}$ in place of $s$, $\theta$, $\ell_x$, and $\ell_y$, and take

limits on both sides. Then as $n \to \infty$,

$$\Big(U_i - c_x\Big) = \widehat{s}\Big((u_i' - c_x).\ \cos\theta + (v_i' - c_y).\ \sin\theta\Big) + \widehat{\ell_x} + \psi_U$$
$$\Big(V_i - c_y\Big) = \widehat{s}\Big(-(u_i' - c_x).\ \sin\theta + (v_i' - c_y).\ \cos\theta\Big) + \widehat{\ell_y} + \psi_V,$$

where, the errors $\psi_U$ and $\psi_V$ are associated with the inconsistency of $s$, $\ell_x$, and $\ell_y$. Now, if we compute $u_i'$ and $v_i'$ from the above model, those expressions will not be equal to the expressions mentioned in Eqn. (A.3.5). This contradicts Proposition 1. Hence, $\widehat{\phi} \xrightarrow{P} \phi$. This completes the proof of Proposition 2. ■

# Bibliography


Alcantarilla, P. F., Bartoli, A., and Davison, A. J. (2012). Kaze features. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*, pages 214–227. Springer.

Alcantarilla, P. F. and Solutions, T. (2011). Fast explicit diffusion for accelerated features in nonlinear scale spaces. *IEEE Trans. Patt. Anal. Mach. Intell*, 34(7):1281–1298.

Ashburner, J. (2007). A fast diffeomorphic image registration algorithm. *NeuroImage*, 38(1):95–113.

Ashburner, J. and Friston, K. J. (2007). Rigid body registration. *Statistical parametric mapping: The analysis of functional brain images*, pages 49–62.

Basri, R. and Jacobs, D. W. (1997). Recognition using region correspondences. *International Journal of Computer Vision*, 25:145–166.

Baumberg, A. (2000). Reliable feature matching across widely separated views. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 1, pages 774–781. IEEE.

Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359.

Bay, H., Tuytelaars, T., and Van Gool, L. (2006). Surf: Speeded up robust features. In *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I 9*, pages 404–417. Springer.

Beg, M. F., Miller, M. I., Trouvé, A., and Younes, L. (2005). Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International journal of computer vision*, 61:139–157.

Bookstein, F. L. (1989). Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on pattern analysis and machine intelligence*, 11(6):567–585.

Calonder, M., Lepetit, V., Ozuysal, M., Trzcinski, T., Strecha, C., and Fua, P. (2011). Brief: Computing a local binary descriptor very fast. *IEEE transactions on pattern analysis and machine intelligence*, 34(7):1281–1298.

Carneiro, G. and Jepson, A. D. (2002). Phase-based local features. In *Computer Vision—ECCV 2002: 7th European Conference on Computer Vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part I 7*, pages 282–296. Springer.

Castanheira de Souza, R. H., Okutomi, M., and Torii, A. (2012). Real-time image mosaicing using non-rigid registration. In *Advances in Image and Video Technology: 5th Pacific Rim Symposium, PSIVT 2011, Gwangju, South Korea, November 20-23, 2011, Proceedings, Part I 5*, pages 311–322. Springer.

Catarious Jr, D. M., Baydush, A. H., and Floyd Jr, C. E. (2006). Characterization of difference of gaussian filters in the detection of mammographic regions. *Medical physics*, 33(11):4104–4114.

Chen, C., Li, Y., Liu, W., and Huang, J. (2015). Sirf: Simultaneous satellite image registration and fusion in a unified framework. *IEEE Transactions on Image Processing*, 24(11):4213–4224.

Christensen, G. E. and Johnson, H. J. (2001). Consistent image registration. *IEEE transactions on medical imaging*, 20(7):568–582.

Cover, T. M. (1999). *Elements of information theory*. John Wiley & Sons.

Crowley, J. L. and Parker, A. C. (1984). A representation for shape based on peaks and ridges in the difference of low-pass transform. *IEEE transactions on pattern analysis and machine intelligence*, 6(2):156–170.

Crum, W. R., Hartkens, T., and Hill, D. (2004). Non-rigid image registration: theory and practice. *The British journal of radiology*, 77(suppl_2):S140–S153.

Das, S. and Mukherjee, P. S. (2024). Image registration for zooming using similarity matching. *Journal of Data Science*, 22(4):558–574.

Das, S., Roy, A., and Mukherjee, P. S. (2024). Image registration for zooming: A statistically consistent local feature mapping approach. *Stat*, 13(1):e664.

Debayle, J. and Presles, B. (2016). Rigid image registration by general adaptive neighborhood matching. *Pattern Recognition*, 55:45–57.

Derpanis, K. G. (2010). Overview of the ransac algorithm. *Image Rochester NY*, 4(1):2–3.

Fu, Y., Lei, Y., Wang, T., Curran, W. J., Liu, T., and Yang, X. (2020). Deep learning in medical image registration: a review. *Physics in Medicine & Biology*, 65(20):20TR01.

Giri, R. and Sharma, R. (2011). Image registration and atmospheric motion vectors retrieval from geo-stationary satellite. *Indian Journal of Science and Technology*, pages 1218–1225.

Gonzalez, R. C. and Woods, R. E. (2018). *Digital Image Processing*. Pearson Education, India.

Harris, C. (1993). *Geometry from visual motion*, page 263–284. MIT Press, Cambridge, MA, USA.

Harris, C., Stephens, M., et al. (1988). A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Citeseer.

Juan, L. and Gwun, O. (2009). A comparison of sift, pca-sift and surf. *International Journal of Image Processing (IJIP)*, 3(4):143–152.

Kang, Y. and Qiu, P. (2014). Jump detection in blurred regression surfaces. *Technometrics*, 56(4):539–550.

Kim, T. and Im, Y.-J. (2003). Automatic satellite image registration by combination of matching and random sample consensus. *IEEE transactions on geoscience and remote sensing*, 41(5):1111–1117.

Klein, A., Andersson, J., Ardekani, B. A., Ashburner, J., Avants, B., Chiang, M.-C., Christensen, G. E., Collins, D. L., Gee, J., Hellier, P., et al. (2009). Evaluation of 14 nonlinear deformation algorithms applied to human brain mri registration. *Neuroimage*, 46(3):786–802.

Koenderink, J. J. (1984). The structure of images. *Biological cybernetics*, 50(5):363–370.

Leutenegger, S., Chli, M., and Siegwart, R. Y. (2011). Brisk: Binary robust invariant scalable keypoints. In *2011 International conference on computer vision*, pages 2548–2555. Ieee.

Li, H. and Fan, Y. (2017). Non-rigid image registration using fully convolutional networks with deep self-supervision. *arXiv preprint arXiv:1709.00799*.

Li, S. (2017). A review of feature detection and match algorithms for localization and mapping. In *IOP Conference Series: Materials Science and Engineering*, volume 231, page 012003. IOP Publishing.

Lindeberg, T. (1993). Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention. *International Journal of Computer Vision*, 11(3):283–318.

Liu, L., Jiang, T., Yang, J., and Zhu, C. (2006). Fingerprint registration by maximization of mutual information. *IEEE Transactions on image processing*, 15(5):1100–1110.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110.

Mair, E., Hager, G. D., Burschka, D., Suppa, M., and Hirzinger, G. (2010). Adaptive and generic corner detection based on the accelerated segment test. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part II 11*, pages 183–196. Springer.

Matas, J., Chum, O., Urban, M., and Pajdla, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. *Image and vision computing*, 22(10):761–767.

Mikolajczyk, K. and Schmid, C. (2002). An affine invariant interest point detector. In *Computer Vision—ECCV 2002: 7th European Conference on Computer Vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part I 7*, pages 128–142. Springer.

Mikolajczyk, K., Zisserman, A., and Schmid, C. (2003). Shape recognition with edge-based features. In *British Machine Vision Conference (BMVC'03)*, volume 2, pages 779–788. The British Machine Vision Association.

Mok, T. C. and Chung, A. C. (2020). Fast symmetric diffeomorphic image registration with convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Moravec, H. P. (1981). Rover visual obstacle avoidance. In *IJCAI*, volume 81, pages 785–790.

Nelson, R. C. and Selinger, A. (1998). Large-scale tests of a keyed, appearance-based 3-d object recognition system. *Vision research*, 38(15-16):2469–2488.

Pock, T., Urschler, M., Zach, C., Beichel, R., and Bischof, H. (2007). A duality based algorithm for tv-l 1-optical-flow image registration. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 511–518. Springer.

Pope, A. R. and Lowe, D. G. (2000). Probabilistic models of appearance for 3-d object recognition. *International Journal of Computer Vision*, 40:149–167.

Qiu, P. (2004). The local piecewisely linear kernel smoothing procedure for fitting jump regression surfaces. *Technometrics*, 46(1):87–98.

Qiu, P. (2005). *Image processing and jump regression analysis*. John Wiley & Sons, New York.

Qiu, P. and Xing, C. (2010). Intensity based nonparametric image registration. In *Proceedings of the international conference on Multimedia information retrieval*, pages 221–226.

Qiu, P. and Xing, C. (2013a). Feature based image registration using non-degenerate pixels. *Signal Processing*, 93(4):706–720.

Qiu, P. and Xing, C. (2013b). On nonparametric image registration. *Technometrics*, 55(2):174–188.

Rosin, P. L. (1999). Measuring corner properties. *Computer Vision and Image Understanding*, 73(2):291–307.

Rosten, E. and Drummond, T. (2006). Machine learning for high-speed corner detection. In *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I 9*, pages 430–443. Springer.

Roy, A. and Mukherjee, P. S. (2024a). A control chart for monitoring images using jump location curves. *Quality Engineering*, 36(2):439–452.

Roy, A. and Mukherjee, P. S. (2024b). Image comparison based on local pixel clustering. *Technometrics*, 66(4):495–506.

Roy, A. and Mukherjee, P. S. (2025). Upper quantile-based cusum-type control chart for detecting small changes in image data. *Journal of Applied Statistics*, 52(11):2156–2171.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee.

Schaffalitzky, F. and Zisserman, A. (2002). Multi-view matching for unordered image sets, or "how do i organize my holiday snaps?". In *Computer Vision—ECCV 2002: 7th European Conference on Computer Vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part I 7*, pages 414–431. Springer.

Schiele, B. and Crowley, J. L. (2000). Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision*, 36:31–50.

Schmid, C. and Mohr, R. (1997). Local grayvalue invariants for image retrieval. *IEEE transactions on pattern analysis and machine intelligence*, 19(5):530–535.

Shen, Z., Han, X., Xu, Z., and Niethammer, M. (2019). Networks for joint affine and non-parametric image registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4224–4233.

Shokoufandeh, A., Marsic, I., and Dickinson, S. J. (1999). View-based object recognition using saliency maps. *Image and Vision Computing*, 17(5-6):445–460.

Song, G., Han, J., Zhao, Y., Wang, Z., and Du, H. A. (2017). A review on medical image registration as an optimization problem. *Current Medical Imaging Reviews*, 13(3):274–283.

Song, H. and Qiu, P. (2017a). Intensity-based 3d local image registration. *Pattern Recognition Letters*, 94:15–21.

Song, H. and Qiu, P. (2017b). A parametric intensity-based 3d image registration method for magnetic resonance imaging. *Signal, Image and Video Processing*, 11(3):455–462.

Song, H. and Qiu, P. (2018). Three-dimensional image registration using distributed parallel computing. *IET Image Processing*, 12(10):1713–1720.

Song, Z., Zhou, S., and Guan, J. (2013). A novel image registration algorithm for remote sensing under affine transformation. *IEEE Transactions on Geoscience and Remote Sensing*, 52(8):4895–4912.

Srivastava, A. and Klassen, E. P. (2016). *Functional and Shape Data Analysis*. Springer, New York, NY.

Torr, P. H. S. (1995). *Motion segmentation and outlier detection*. PhD thesis, University of Oxford UK.

Weickert, J., Grewenig, S., Schroers, C., and Bruhn, A. (2016). Cyclic schemes for pde-based image analysis. *International Journal of Computer Vision*, 118:275–299.

Wensley, J. H., Lamport, L., Goldberg, J., Green, M. W., Levitt, K. N., Melliar-Smith, P. M., Shostak, R. E., and Weinstock, C. B. (1978). Sift: Design and analysis of a fault-tolerant computer for aircraft control. *Proceedings of the IEEE*, 66(10):1240–1255.

Xing, C. and Qiu, P. (2011). Intensity-based image registration by nonparametric local smoothing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):2081–2092.

Yang, K., Pan, A., Yang, Y., Zhang, S., Ong, S. H., and Tang, H. (2017). Remote sensing image registration using multiple image features. *Remote Sensing*, 9(6):581.

Ye, Y., Bruzzone, L., Shan, J., Bovolo, F., and Zhu, Q. (2019). Fast and robust matching for multimodal remote sensing image registration. *IEEE Transactions on Geoscience and Remote Sensing*, 57(11):9059–9070.

Ye, Y., Shan, J., Bruzzone, L., and Shen, L. (2017). Robust registration of multimodal remote sensing images based on structural similarity. *IEEE Transactions on Geoscience and Remote Sensing*, 55(5):2941–2958.

Zhang, M., Liao, R., Dalca, A. V., Turk, E. A., Luo, J., Grant, P. E., and Golland, P. (2017). Frequency diffeomorphisms for efficient image registration. In Niethammer, M., Styner, M., Aylward, S., Zhu, H., Oguz, I., Yap, P.-T., and Shen, D., editors, *Information Processing in Medical Imaging*, pages 559–570, Cham. Springer International Publishing.

Zhang, Z., Deriche, R., Faugeras, O., and Luong, Q.-T. (1995). A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial intelligence*, 78(1-2):87–119.

Zhou, H., Yuan, Y., and Shi, C. (2009). Object tracking using sift features and mean shift. *Computer vision and image understanding*, 113(3):345–352.