



Faculty of Engineering

**Enhanced Sparse Keypoint Detection in Near-Uniform Scenes for Robust
Image Stitching**

Jong Tze Kian

**Doctor of Philosophy
2025**

Enhanced Sparse Keypoint Detection in Near-Uniform Scenes for Robust Image Stitching

Jong Tze Kian

A thesis submitted

In fulfillment of the requirements for the degree of Doctor of Philosophy

(Electronic Engineering)

Faculty of Engineering
UNIVERSITI MALAYSIA SARAWAK
2025

DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Malaysia Sarawak. Except where due acknowledgements have been made, the work is that of the author alone. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.



.....
Signature

Name: Jong Tze Kian

Matric No.: 17010104

Faculty of Engineering

Universiti Malaysia Sarawak

Date: 13 August 2025

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to my supervisor, Associate Prof. Ir. Dr. David Bong Boon Liang, for his exceptional guidance, encouragement, and advice throughout my PhD study. His profound expertise and logical approach have been of immense value to me.

I am thankful to the Public Service Department (JPA) for providing me with the Hadiah Latihan Persekutuan (HLP) scholarship, which alleviated financial concerns during my PhD study. Additionally, I appreciate the support from the Ministry of Higher Education Malaysia through the Fundamental Research Grant Scheme (FRGS), with project code: FRGS/1/2016/TK04/UNIMAS/02/1.

I am grateful to everyone who contributed to my research by exchanging ideas, discussing problems, and sharing valuable information. Their guidance, advice, and insights are indispensable for the successful completion of this work.

Finally, my heartfelt gratitude goes to my beloved family, especially my wife, daughters, and parents, for their love, unwavering support, and encouragement. Their invaluable love and support are indispensable throughout the completion of this thesis.

ABSTRACT

Image stitching is a vital image processing technique in many modern multimedia applications, particularly for creating panoramic images and virtual reality environments. However, this processing presents significant challenges in near-uniform scenes, where theoretical feature detectors often struggle to identify distinctive interest points. This problem arises when images contain homogeneous content that lacks salient features, making it difficult to generate a sufficient number of well-distributed matching keypoints. The absence of such features often leads to an inadequate number of matched inliers between overlapping images, resulting in undesirable visual effects such as seams, ghosting, and geometric distortions. To address these limitations, this thesis proposes two novel feature detection approaches aimed at improving image stitching performance in near-uniform scenes. The first method, referred to as the enhanced partial differential equation (ePDE), introduces a feature detection technique based on nonlinear diffusion. This approach applies the Lorentz factor, inspired by Einstein's theory of special relativity, to modify the conductivity function within the partial differential equation. This enhancement enables the construction of multiple enhanced edge-preserving nonlinear scale-spaces, significantly improve feature detection in homogeneous regions and outperforming several state-of-the-art detectors. The second method employs the Log-Gabor feature transform (LGFT), which generates multiple oriented and scaled Log-Gabor scale-spaces to enhance the extraction of distinctive and matchable interest points, thereby addressing the limitations of intensity-based feature detection techniques. Additionally, two new evaluation metrics are introduced: the spread-overlap measure (S_o) and the recall-to-spread-overlap ratio (RC/S_o). These metrics offer a more comprehensive performance evaluation by considering both the accuracy and spatial distribution of inliers. Experimental results show that the proposed

LGFT and ePDE methods consistently improve image stitching performance, especially for near-uniform image datasets. Both methods achieve over a 14.63% higher image stitching success rate compared to current state-of-the-art techniques. The top-performing LGFT-1.4.3 variant achieves an average improvement of 4.8% in inlier accuracy and distribution, while the ePDE- g_5 variant records an average improvement of 4.3%, both outperforming existing feature detectors. These results highlight the robustness and effectiveness of the proposed methods in addressing the challenges of stitching near-uniform scenes.

Keywords: Image stitching, feature detection, near-uniform scene, nonlinear diffusion, Log-Gabor feature

Penjahitan Imej untuk Pemandangan Hampir-Seragam

ABSTRAK

Penjahitan imej merupakan suatu teknik pemprosesan imej yang penting dalam pelbagai aplikasi multimedia moden, terutamanya dalam pembinaan imej panorama dan persekitaran realiti maya. Walau bagaimanapun, pemprosesan ini menghadapi cabaran yang ketara dalam pemandangan hampir-seragam, di mana pengesanan ciri teori sering menghadapi kesukaran untuk mengenal pasti titik-titik kepentingan yang distingtif. Masalah ini timbul apabila imej mengandungi kandungan homogen dan kekurangan ciri-ciri yang ketara, menjadikannya sukar untuk menjana bilangan titik pepadanan yang mencukupi dan bertaburan luas. Ketiadaan ciri-ciri ini sering mengakibatkan ketidakcukupan titik pepadanan antara imej-imej yang bertindih, menyebabkan kesan visual yang tidak diinginkan seperti kelim, bayang berganda dan herotan geometri. Bagi mengatasi masalah ini, tesis ini mencadangkan dua pendekatan baharu bagi pengesanan ciri bertujuan untuk meningkatkan prestasi penjahitan imej dalam pemandangan hampir-seragam. Kaedah pertama, dikenali sebagai persamaan pembezaan separa yang dipertingkatkan (ePDE), memperkenalkan teknik pengesanan ciri berasaskan difusi tak linear. Pendekatan ini menggunakan faktor Lorentz yang diilhamkan daripada teori relativiti khas Einstein untuk mengubahsuaikan fungsi kekonduksian dalam persamaan pembezaan separa. Penambahbaikan ini membolehkan pembentukan pelbagai ruang skala tak linear yang dapat mengekalkan sempadan objek dalam imej dengan lebih jelas, sekali gus meningkatkan keupayaan pengesanan ciri dalam kawasan homogen dan mengatasi prestasi beberapa pengesanan ciri terkini sedia ada. Kaedah kedua menggunakan transformasi ciri Log-Gabor (LGFT), yang menghasilkan pelbagai ruang skala Log-Gabor dalam orientasi dan skala berbeza bagi mempertingkatkan pengekstrakan titik kepentingan yang lebih distingtif dan boleh dipadankan. Pendekatan ini

sekali gus mengatasi kekangan teknik pengesan ciri yang berasaskan keamatan imej. Selain itu, dua metrik penilaian baharu turut diperkenalkan: ukuran sebaran-liputan (S_o) dan nisbah ingatan terhadap sebaran-liputan (RC/S_o). Kedua-dua metrik ini menawarkan penilaian prestasi yang lebih komprehensif dengan mengambil kira aspek ketepatan dan taburan ruang inlier. Keputusan eksperimen menunjukkan bahawa kaedah LGFT dan ePDE yang dicadangkan secara konsisten meningkatkan prestasi penjahitan imej, terutamanya bagi set data imej hampir-seragam. Kedua-dua kaedah ini mencapai kadar kejayaan penjahitan imej yang lebih tinggi melebihi 14.63% berbanding dengan teknik terkini sedia ada. Varian LGFT-1.4.3 yang menunjukkan prestasi terbaik mencatat purata peningkatan 4.8% dari segi ketepatan dan taburan inlier, manakala varian ePDE- g_5 merekodkan purata peningkatan 4.3%, kedua-duanya mengatasi pengesan ciri sedia ada. Keputusan ini membuktikan keteguhan dan keberkesanan kaedah-kaedah yang dicadangkan dalam menangani cabaran penjahitan image yang berpemandangan hampir seragam.

Kata kunci: Penjahitan imej, pengesanan ciri, pemandangan hampir-seragam, difusi tak linear, ciri Log-Gabor

TABLE OF CONTENTS

	Page
DECLARATION	i
ACKNOWLEDGEMENT	ii
ABSTRACT	iii
ABSTRAK	v
TABLE OF CONTENTS	vii
LIST OF TABLES	xi
LIST OF FIGURES	xii
LIST OF ABBREVIATIONS	xvii
LIST OF SYMBOLS	xx
CHAPTER 1 INTRODUCTION	1
1.1 Study Background	1
1.2 Problem Statement	3
1.3 Objectives	8
1.4 Significance and Contribution	9
1.5 Thesis Outline	10
CHAPTER 2 LITERATURE REVIEW	12
2.1 Introduction	12

2.2	Image Stitching Review	14
2.3	Feature Detectors	21
2.3.1	Scale Invariant Feature Transform (SIFT) Detector	26
2.3.2	Speeded-Up Robust Features (SURF) Detector	28
2.4	Feature Descriptors	32
2.4.1	SIFT Descriptor	33
2.4.2	SURF Descriptor	34
2.5	Performance Evaluation	37
2.5.1	Repeatability Metric	39
2.5.2	Recall Metric	40
2.5.3	Precision Metric	41
2.5.4	Limitations	42
2.6	Theoretical versus Deep Learning	43
2.7	Research Gap	46
2.8	Chapter Summary	49
	CHAPTER 3 METHODOLOGY	51
3.1	Introduction	51
3.2	Proposed Method 1: Enhanced Partial Differential Equation (ePDE)	52
3.2.1	Nonlinear Diffusion Filtering	54
3.2.2	Lorentz Factor	56

3.2.3	Improved Conductivity Function	56
3.2.4	Lorentz Modulated Nonlinear Scale-spaces	59
3.2.5	ePDE Feature Detection	61
3.3	Proposed Method 2: Log-Gabor Feature Transform (LGFT)	64
3.3.1	2D Log-Gabor Filtering and Scale-spaces	65
3.3.2	LGFT Feature Detection	69
3.4	Feature Description	71
3.5	Image Stitching Algorithm	73
3.6	Proposed Performance Evaluation Metrics	75
3.6.1	Spread-Overlap Measure	76
3.6.2	RC/S_O Score	77
3.7	Chapter Summary	83
CHAPTER 4 RESULTS AND DISCUSSION		85
4.1	Introduction	85
4.2	Experimental Tools and Datasets	86
4.3	Pre-analysis and Parameter Settings	93
4.3.1	Proposed Method 1: ePDE Method	94
4.3.2	Proposed Method 2: LGFT Method	97
4.4	Scale-space Comparisons	104
4.4.1	Proposed Method 1: ePDE Method	105

4.4.2	Proposed Method 2: LGFT Method	107
4.5	Performance Comparisons	109
4.5.1	Quantitative Analysis	109
4.5.2	Qualitative Comparison	118
4.5.3	Execution Efficiency	137
4.6	Chapter Summary	139
	CHAPTER 5 CONCLUSION AND FUTURE WORKS	142
5.1	Conclusion	142
5.2	Limitations	143
5.3	Contributions	144
5.4	Future Works	146
	REFERENCES	149
	APPENDICES	166

LIST OF TABLES

	Page
Table 2.1 Summary of some state-of-the-art feature detectors and the datasets used for their performance evaluation	47
Table 3.1 Summary of the proposed enhanced Partial Differential Equation (ePDE) feature detector algorithm	63
Table 3.2 Summary of the proposed Log-Gabor Feature Transform (LGFT) feature detector algorithm	70
Table 4.1 Summary of feature detectors and descriptors used for validation, including feature types, data formats, associated tools, key functions, and parameter settings	92
Table 4.2 Parameter settings for the Log-Gabor filter and the box filter approximation of the Laplacian of Gaussian. The radial bandwidth σfs and scaling factor C are based on Kovese (2003), while the parameter w is derived from the results shown in Figures 4.6 and 4.7	98
Table 4.3 Comparison of the average processing time for each feature detector, calculated over 20 images and measured in seconds	137

LIST OF FIGURES

	Page
Figure 1.1 Examples of near-uniform scene images	4
Figure 1.2 Examples of "clustering" interest points in the stitched images. Yellow crosses represent interest points from one image, while magenta circles mark their corresponding points in the overlapping image. The seam is retained to highlight the stitching alignment	6
Figure 2.1 Classification of image stitching techniques based on registration methods	15
Figure 2.2 Common image stitching processing steps for mosaic stitching and panoramic stitching	18
Figure 2.3 Classification of common feature detection techniques	22
Figure 2.4 <i>DoG</i> representations for local extrema detection are generated by subtracting adjacent Gaussian images in each octave. Each subsequent octave involves down-sampling the Gaussian images by a factor of 2 (Lowe, 2004)	27
Figure 2.5 Calculating area using integral images (Evans, 2009)	29
Figure 2.6 Approximation of the Laplacian of Gaussian. The discretised and cropped Laplacian of Gaussians: (a) L_{xx} , (b) L_{yy} and (c) L_{xy} . The corresponding weighted box filter approximations: (d) D_{xx} , (e) D_{yy} and (f) D_{xy} (Bay et al., 2008)	30
Figure 2.7 Scale-spaces pyramids: (a) SIFT varies the image size and applies Gaussian smoothing to the resulting layers. (b) SURF retains the image size and changes the filter size instead (Bay et al., 2008)	31
Figure 2.8 Construction of a 128-element SIFT feature vector derived from a 16×16 sample grid centered around the interest point position (Awad & Hassaballah, 2016)	34
Figure 2.9 As the circular sector rotates around the origin, moving from blue to red to green areas, the Haar responses (marked as red dots) within each sector are summed to produce vectors a , b and c . The longest of these vectors represents the dominant orientation	35
Figure 2.10 Construction of a 64-element SURF feature vector derived from a 4×4 grid of subregions (Awad & Hassaballah, 2016; Bay et al., 2008)	36

Figure 3.1	The relationship between conductivity coefficients g_2 , g_4 and g_5 and image spatial gradients ∇I with a fixed parameter $k = 0.7$	58
Figure 3.2	Non-maxima suppression. The pixel marked '×' is selected as interest point if it is greater than neighboring pixels in its interval and adjacent intervals (Awad & Hassaballah, 2016)	62
Figure 3.3	Comparison between (a) Gabor and (b) Log-Gabor functions (Field, 1987)	65
Figure 3.4	Orientation assignment process. Haar wavelet responses in the x and y directions are summed within a rotating circular segment. The segment with the largest resultant vector indicates the dominant orientation of the interest point	72
Figure 3.5	Descriptor construction process. The blue window (sized at $24\alpha i$) is centered on the interest point (red dot) and aligned with the dominant orientation (green arrow). The green square is one of 16 subregions; blue dots indicate pixels used to compute Haar wavelet responses for constructing the $4 \times 4 \times 4$ -element descriptor	73
Figure 3.6	Flowchart outlining the entire process of image stitching	74
Figure 3.7	Example of spread-overlap, So calculation: 50 out of 400 grid cells within a 100×100 pixel overlapping area contain at least one inlier	77
Figure 3.8	Visualization of RC/So concept with 30 inliers (marked with red dots) across a 6×6 grid in the overlapping region. RC is fixed at 0.714	78
Figure 3.9	Examples of the proposed evaluation metrics, So and RC/So , for the LGFT, RIFT, ORB, and SURF. Yellow crosses (reference image) and magenta circles (target image) mark inlier locations. The yellow zoom-in highlights the misalignment, and the blue trapezoid shows distortion in the stitched images	79
Figure 3.10	Examples of the proposed evaluation metrics, So and RC/So , for the LGFT, KAZE, SIFT and AGAST. Yellow crosses (reference image) and magenta circles (target image) mark inlier locations. The yellow zoom-in highlights the misalignment	81
Figure 3.11	Examples of the proposed evaluation metrics, So and RC/So , for the ePDE, SIFT, BRISK and MSER. Yellow crosses (reference image) and magenta circles (target image) mark inlier locations. The yellow zoom-in highlights the misalignment	82

Figure 4.1	The benchmark image pairs include: (a)-(b) for blurred images, (c) for light variation, (d) for JPEG compression, and (e) for Gaussian noise. Each series contains 6 sequences of progressively transformed images (from top to bottom)	88
Figure 4.2	Examples of general image pairs illustrating real-world scenes with partially near-uniform regions	89
Figure 4.3	Examples of cropped HiRISE image pairs featuring predominantly near-uniform regions in grayscale	90
Figure 4.4	Performance comparison of ePDE- $g4$ and ePDE- $g5$ based on the number of inliers, Nm and spread-overlap, So relative to different contract factors k . The metrics are averaged over 25 benchmark image pairs	95
Figure 4.5	Performance comparison of ePDE- $g4$ and ePDE- $g5$ based on the recall, RC and RC/So relative to different contract factors k . The metrics are averaged over 25 benchmark image pairs	96
Figure 4.6	Analysis of parameter w using the first image pair from each benchmark dataset, with parameter sets, Set 1 and Set 2 (see Table 4.2)	99
Figure 4.7	Analysis of parameter w using the first image pair from each benchmark dataset, with parameter sets, Set 3 and Set 4 (see Table 4.2)	100
Figure 4.8	Evaluation of octave-sublevel parameters for the number of inliers, Nm and spread-overlap, So , using parameter sets from Table 4.2, applied across all benchmark datasets	102
Figure 4.9	Evaluation of octave-sublevel parameters for RC/So and the execution time t for single image, using parameter sets from Table 4.2, applied across all benchmark datasets	103
Figure 4.10	An example of nonlinear scale-spaces generated with the conductivity functions $g2$, $g4$ and $g5$ at different contrast factors k . Each image, cropped into a square shape, represents the 12th layer of the scale-space	106
Figure 4.11	Comparison of various scale-space generation methods. Top row: Input image pair. Middle row: Cropped images showing (a) Gaussian, L ; (b) Different of Gaussian, DoG ; (c) nonlinear diffusion, Lt ; and (d) Log-Gabor, As, o scale-spaces. Bottom row: Matching and stitching effects produced by SIFT, KAZE, and LGFT	108
Figure 4.12	Success rate of different feature-based image stitching across 95 pairs of benchmark and general images	110

Figure 4.13	The average number of inliers, Nm varies among different feature detectors for 60 sample image pairs	112
Figure 4.14	The average spread-overlap, So and average number of inliers, Nm varies among different feature detectors for 60 sample image pairs	113
Figure 4.15	The average recall, RC and average number of inliers, Nm varies among different feature detectors for 60 sample image pairs	114
Figure 4.16	The average RC/So scores and average number of inliers, Nm varies among different feature detectors for 60 sample image pairs	115
Figure 4.17	Success rate of different feature-based image stitching across 123 pairs of HiRISE images	117
Figure 4.18	Comparison of inlier count and distribution in the stitched images of blurry 'bikes' using different feature detectors. Inliers are marked by white circles and crosses. (m) & (n) include stitched images without markers for clarity	120
Figure 4.19	Comparison of inlier count and distribution in the stitched images of light-changed 'leuven' using different feature detectors. Inliers are marked by white circles and crosses. (m) & (n) include stitched images without markers for clarity	121
Figure 4.20	Comparison of resulting stitched image produced by the MSER, ePDE- $g5$, and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers	123
Figure 4.21	Comparison of resulting stitched image produced by the SURF, ePDE- $g5$, and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers	124
Figure 4.22	Comparison of resulting stitched image produced by the BRISK, ePDE- $g5$, and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers	125
Figure 4.23	Comparison of resulting stitched image produced by the KAZE, ePDE- $g5$, and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers	126
Figure 4.24	Comparison of resulting stitched image produced by the SIFT, ePDE- $g5$, and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers	127
Figure 4.25	Comparison of resulting stitched image produced by the AKAZE, ePDE- $g5$, and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers	127

Figure 4.26	Comparison of resulting stitched image produced by the ORB, ePDE- <i>g5</i> , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers	128
Figure 4.27	Comparison of resulting stitched image produced by the U-RIFT, ePDE- <i>g5</i> , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers	129
Figure 4.28	Comparison of resulting stitched image produced by the AGAST, ePDE- <i>g5</i> , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers	130
Figure 4.29	Comparison 1 of resulting stitched images produced by the KAZE, AKAZE, ePDE- <i>g5</i> , and LGFT-1.4.3 methods. The approximate inlier locations are marked by white circles and cross markers	132
Figure 4.30	Comparison 2 of resulting stitched images produced by the KAZE, AKAZE, ePDE- <i>g5</i> , and LGFT-1.4.3 methods. The approximate inlier locations are marked by white circles and cross markers	133
Figure 4.31	Comparison 3 of resulting stitched images produced by the KAZE, AKAZE, ePDE- <i>g5</i> , and LGFT-1.4.3 methods. The approximate inlier locations are marked by white circles and cross markers	134
Figure 4.32	Comparison 4 of resulting stitched images produced by the KAZE, AKAZE, ePDE- <i>g5</i> , and LGFT-1.4.3 methods. The approximate inlier locations are marked by white circles and cross markers	135

LIST OF ABBREVIATIONS

AKAZE	Accelerated KAZE
ANAP	Adaptive as-Natural-as-Possible
AOS	Additive Operator Splitting
APAP	As-Projective-As-Possible
ASIFT	Affine-SIFT
BFSIFT	Bilateral Filter SIFT
BRIEF	Binary Robust Independent Elementary Features
BRISK	Binary Robust Invariant Scalable Keypoints
CenSurE	Center Surround Extremas
CNN	Convolutional Neural Network
CSIFT	Colored SIFT
CV	Computer Vision
DL	Deep Learning
DLNR-SIQA	Deep Learning based No-Reference SIQA
ePDE	Enhanced Partial Differential Equation
FAST	Features from Accelerated Segment Test
FOV	Field-of-view
FREAK	Fast Retina Keypoint
GIS	Geographic Information System
GLOH	Gradient Location-Orientation Histogram
HiRISE	High Resolution Imaging Science Experiment images
HFI-SSIM	High-Frequency Information Structural Similarity

IQA	Images Quality Assessment
JPEG	Joint Photographic Experts Group
KAZE	KAZE Features
LoG	Laplacian of Gaussian
LGFT	Log-Gabor Feature Transform
LGFT-1.4.3	LGFT with 1 orientation, 4 scales and Set3 parameters
LGFT-2.4.3	LGFT with 2 orientations, 4 scales and Set3 parameters
LGFT-4.4.3	LGFT with 4 orientations, 4 scales and Set3 parameters
mexOpenCV	MATLAB mex functions that interface OpenCV APIs
MATLAB	MATrix LABoratory developed by MathWorks
MIM	Maximum Index Map
MSAC	M-estimator SAmple Consensus
MSER	Maximally Stable Extremal Regions
MSFT	Multi-scale Structural Feature Transform
M-LDB	Modified Local Difference Binary
NASA	National Aeronautics and Space Administration
NCC	Normalized Cross Correlation
OpenCV	Real-time optimized Computer Vision library
OpenSURF	Open-source library of SURF detector-descriptor scheme
ORB	Oriented FAST and Rotated BRIEF
PCA-SIFT	Principal Components Analysis SIFT
PDE	Partial Differential Equation
PRBFA	Patch Registration and Bidimensional Feature Aggregation
RANSAC	Random Sample Consensus
RIFT	Radiation-variation Insensitive Feature Transform

SEAGULL	Seam-Guided Local Alignment
SIFT	Scale-Invariant Feature Transform
SIQA	Stitched Images Quality Assessment
SIQE	Stitched Image Quality Evaluator
SLAM	Simultaneous Localization and Mapping
SPHP	Shape-Preserving Half Projective
SRP-AKAZE	Sparse Random Projection AKAZE
SSD	Sum of Squared Difference
SSIM	Structural Similarity Index
SURF	Speeded-Up Robust Features
SUSAN	Smallest Univalued Segment Assimilating Nucleus
SVA	Smoothly Varying Affine
TDMA	Thomas algorithm or Tri-Diagonal Matrix Algorithm
UAV	Unmanned Aerial Vehicle
U-RIFT	Upright version of RIFT
VLFeat	Open-source computer vision library

LIST OF SYMBOLS

CM	Number of correct matches
$\det(H_{approx})$	Determinant of Hessian
DoG	Difference of Gaussian
DoH	Determinant of Hessian
d_x	Haar wavelet responses in the x direction
d_y	Haar wavelet responses in the y direction
D_{xx}	Weighted box filters of LoG in the x direction
D_{xy}	Weighted box filters of LoG in the x and y directions
D_{yy}	Weighted box filters of LoG in the y direction
$D(x, y, \sigma)$	Difference of Gaussian
ePDE- g_4	ePDE based on conductivity function of g_4
ePDE- g_5	ePDE based on conductivity function of g_5
FM	Number of false matches
$\mathcal{F}(I)$	Fourier-transformed image
$g(x, y, t)$	Nonlinear diffusion conductivity functions
$G_o(\theta)$	Angular component of Log-Gabor function
$G_s(f)$	Radial component of Log-Gabor function
$G_{s,o}(f, \theta)$	2D Log-Gabor filter in frequency domain
$G_{s,o}(\rho, \theta)$	2D Log-Gabor filter in log-polar coordinates
$G(x, y, \sigma)$	Gaussian kernels
$H(x, y, \sigma)$	Hessian matrix

$L(x, y, \sigma)$	Laplacian of Gaussian
$Prec$	Precision
R	Repeatability metric
RC	Recall metric
RC/S_o	Ratio of Recall-to-Spread-overlap
S_o	Spread-overlap metric

CHAPTER 1

INTRODUCTION

1.1 Study Background

Image stitching is a technique that aligns and combines multiple overlapping images of a scene into a single, seamless image on a common reference frame or canvas. The primary goal is to create an image that offers a wider field of view (FOV) and enhanced resolution. Over the past two decades, significant advancements in image stitching research have accompanied numerous technological innovations. These developments are critical to various modern visual communication systems, including mobile devices (Xiong & Pulli, 2010), surveillance systems (Yang et al., 2025), virtual reality environments (Nam & Han, 2021), biomedical imaging (Bergen & Wittenberg, 2016) and social media platforms (Huang, 2016). As these technologies have become more widespread, they produce and process vast amounts of diverse visual data. Integrating image stitching with other fields unlocks new opportunities for users to interact with visual data in innovative and immersive ways. For instance, image stitching enables the creation of panoramic street views (Biljecki & Ito, 2021), enhances omnidirectional backgrounds in virtual reality environments (Nam & Han, 2021), and supports 3D surface reconstruction (Fu et al., 2023; Jiang et al., 2023). It also plays a crucial role in applications like simultaneous localization and mapping (SLAM) (Dong & Ganz, 2019) and biomedical imaging (Bergen & Wittenberg, 2016), among others.

According to Laato et al. (2024), geographic information systems (GIS) and online mapping services like Google Maps and OpenStreetMap have evolved from using raster image tiles to providing higher-fidelity 3D spatial information. This evolution has been accelerated by advancements in image stitching techniques, particularly when combined

with unmanned aerial vehicles (UAV), enabling large-scale 3D geographical mapping at relatively low costs. As algorithms in this field have progressed and been increasingly integrated, image stitching has become a fundamental aspect of image processing. However, achieving efficient and high-quality image stitching remains a significant challenge and has been a persistent concern in computer vision (CV) since its early days. The subsequent section will provide a more detailed discussion of the specific challenges associated with efficient and high-quality image stitching for near-uniform datasets.

The popularity of image stitching lies in its cost-effective and capability to capture panoramic and wide FOV images without the need for expensive and specialized cameras. The effectiveness of image stitching tools integrated into handheld mobile cameras to create panoramic views often leads to the misconception that the challenges associated with image stitching have been completely addressed. However, many of these tools rely on a mix of basic image stitching techniques and stabilization algorithms (Lyu et al., 2019). These tools typically require a controlled acquisition environment, where the camera is moved or rotated horizontally by a fixed angle around its vertical axis in a single direction to ensure a consistent overlapping area. Moving the camera too quickly or in the incorrect direction can lead the stitching process to fail, resulting in choppy or misaligned images.

Additionally, handheld mobile cameras often simplify the image stitching process by bypassing steps like feature extraction, correspondence matching, and transformation estimation under certain conditions (Ha et al., 2007). The embedded stitching algorithms are typically fine-tuned through pre-calibration of the vision sensors' intrinsic and extrinsic parameters, creating a consistent stitching template and reducing cumulative errors to produce smooth panoramas (Lyu et al., 2019). This suggests that current image stitching

algorithms still struggle with many real-world challenges and often rely on additional support from pre-calibration processes, controlled acquisition conditions, and image stabilization to perform effectively in practical applications.

Even in the era of deep learning (DL), many applications still benefit from theoretical CV algorithms that can be more suitable and effective for certain tasks, such as image stitching and 3D mesh reconstruction. In contrast, DL can sometimes be unnecessarily applied to simpler tasks like colour classification (O’Mahony et al., 2020). Therefore, theoretical CV methods offer a practical alternative, especially in situations where DL might be excessive or resource intensive. Hybrid approaches that combine DL with theoretical feature-based methods have shown advantages in feature detection applications (O’Mahony et al., 2020), demonstrating that theoretical CV remains relevant and valuable today. Since DL generally requires large datasets for training and demands substantial computational resources, this research deliberately focuses on enhancing and applying theoretical image stitching methods. In particular, it aims to improve feature detection algorithms to address challenges associated with near-uniform datasets.

1.2 Problem Statement

Image stitching technology has seen significant advancements, enhancing many everyday applications by addressing the limitations of field-of-view (FOV) in images and videos. Most image stitching techniques perform well on many published datasets and images that are carefully chosen, typically featuring natural baselines and minimal or no parallax in feature-rich scenes. However, the stitching of real-world images remains

challenging due to wide baselines, large parallax, variations in illumination and contrast, moving objects, low-texture, and occlusion (Lyu et al., 2019; Pandey & Pati, 2019).

Images with low-texture, also referred to as near-uniform scenes, can appear either in part or as whole images. These scenes often contain large areas of uniform background, such as smooth flooring surfaces, skies with fine clouds, oceans, deserts, grass fields, forests, and smooth planetary surfaces. Figure 1.1 presents several examples of near-uniform scenes. These types of images present significant challenges for theoretical feature extraction methods compared to those with rich and distinctive textures. Theoretical feature-based image stitching algorithms often find it challenging to identify sufficient distinctive corresponding interest points for accurately matching and aligning near-uniform images. The

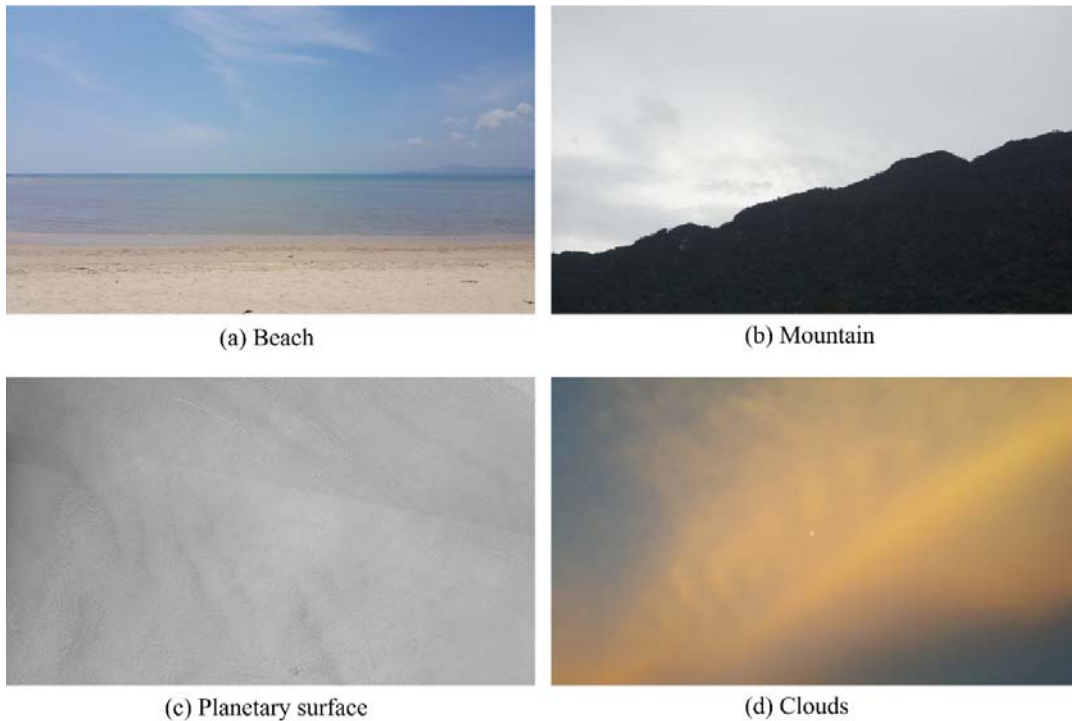


Figure 1.1: Examples of near-uniform scene images

lack of distinctive information in these scenes makes it difficult for feature detectors to find stable corresponding points between images, resulting in noticeable stitching artifacts such as seams, ghosting effects, and significant geometric distortion in the final stitched image.

Current feature detectors are fundamentally designed to identify interest points based on semantic structures in image content, such as ridges, edges, corners, and blobs. When images contain partially near-uniform regions, the corresponding interest points in overlapping areas are often distributed unevenly. This uneven distribution can lead to substantial transformation distortions between images, resulting in poor stitching quality. The problem arises because corresponding interest points are typically absent in featureless or homogeneous regions and are instead concentrated in small, feature-rich areas, leading to "clustering" within a narrow portion of the overlapping region, as shown in Figure 1.2. As a result, fitting this unevenly distributed data into a geometric transformation model becomes challenging, ultimately degrades the quality of the stitching. Therefore, it is essential to achieve a uniform distribution of corresponding interest points in the overlapping areas of the image pairs to accurately estimate the geometric transformation before alignment, ensuring the highest possible stitching quality.

While perfect alignment is rarely achieved for real-world scene images, many researchers have worked on developing improved alignment and composition techniques to effectively reduce or hide stitching artifacts (Nie et al., 2020; Zaragoza et al., 2014). However, addressing severe misalignment or distortion remains challenging (Xiang et al., 2018). Some researchers have developed mesh-based alignment optimization algorithms to minimize misalignment and address challenging issues, especially those related to parallax and wide baselines (Zaragoza et al., 2014). However, to the best of author's understanding,

no efforts have yet been made to specifically address misalignment caused by near-uniform or low-texture scenes.

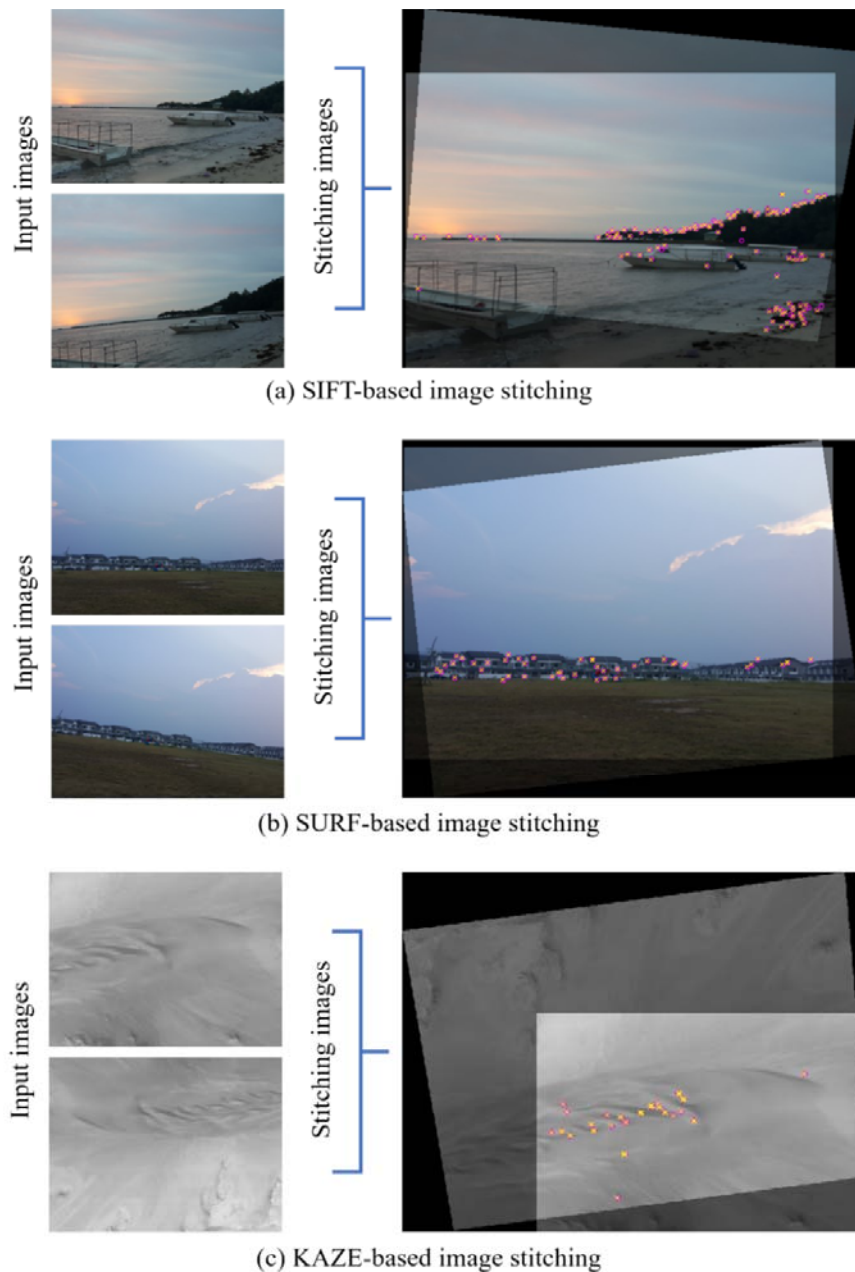


Figure 1.2: Examples of "clustering" interest points in the stitched images. Yellow crosses represent interest points from one image, while magenta circles mark their corresponding points in the overlapping image. The seam is retained to highlight the stitching alignment

In near-uniform scenes, Fu et al. (2023) points out that existing algorithms typically use a multi-threshold point filter to enhance the uniformity of feature point distribution within an image. However, this approach is prone to feature point mismatches. Deep learning techniques can also be employed to insert feature points into near-uniform regions, potentially helping to address this issue, but they require significant computational resources (Fu et al., 2023). Some researchers have emphasized the need to focus more on image stitching in real-world and complex scenes, including those with near-uniform or low-texture scenes (Fu et al., 2023; Lyu et al., 2019).

Since image stitching combines multiple images, evaluating the resulting quality is essential to assess the effectiveness of stitching algorithms. Typically, this evaluation considers several factors, including registration accuracy, structural similarity, smooth transitions between images, and computational efficiency (Bergen & Wittenberg, 2016). Common metrics such as repeatability (Mikolajczyk & Schmid, 2001) as well as recall and precision (Mikolajczyk & Schmid, 2005) are widely used to assess the effectiveness of feature detectors and descriptors. However, these metrics do not directly address the impact of matched corresponding points between images on the overall stitching quality. High repeatability and recall do not guarantee that the number and spatial distribution of corresponding points are adequate to achieve high-quality stitching results.

The Structural Similarity Index (SSIM), introduced by Wang et al. (2004), is one of the earliest metrics used to evaluate image quality, but it is not well-suited for capturing stitching artifacts present in panoramic stitched images (Ullah et al., 2020). Since then, numerous methods for Stitched Image Quality Assessment (SIQA) have been established (Tian et al., 2021). While these methods show promising results in assessing stitched image

quality, several challenges remain. One major issue is the lack of a publicly available, general-purpose benchmark dataset that includes distorted panoramic images covering a wide range of distortion types (Ullah et al., 2022; Yang et al., 2025). The absence of datasets, particularly those featuring near-uniform scenes, limits the effectiveness of existing SIQA methods to detect subtle stitching artifacts that are also difficult for human perception. Besides, current assessment approaches do not directly correlate the performance of feature detectors with the overall quality of the stitched images.

1.3 Objectives

The primary goal of this thesis is to develop new feature-based stitching techniques to address the challenges posed by stitching near-uniform images. It also introduces new evaluation metrics that better reflect the relationship between the inliers detected by feature detectors and the resulting quality of stitched images. Using these metrics, the effectiveness of the proposed methods is evaluated and compared with several advanced stitching techniques.

The research addresses key issues in image stitching, particularly in near-uniform scenes, through the following objectives:

- i. To enhance keypoint feature detection through a Lorentz-modulated nonlinear diffusion approach in the spatial domain.
- ii. To improve the representation of texture details in near-uniform scenes using a Log-Gabor feature transform approach in the frequency domain.

- iii. To develop new evaluation metrics for feature-based image stitching that capture both inlier accuracy and their spatial distribution.

1.4 Significance and Contribution

This research explores advanced feature detection techniques aimed at improving image stitching quality and mitigating critical misalignment and geometric distortion, particularly in near-uniform or low-texture scenes. It focuses on developing and validating new feature detection methods based on two approaches: Lorentz-modulated nonlinear diffusion and Log-Gabor feature transforms. These approaches are proposed to address different limitations in near-uniform image stitching, where the former improves scale-space representation by preserving object boundaries in the spatial domain, and the latter enhances robustness by capturing local texture granularity in the frequency domain. These methods are designed to increase the number of reliable matchable interest points, or inliers, across a broader area within the overlapping images. As a result, the proposed techniques seek to generate high-quality image stitching and minimize visible artifacts in the resulting stitched images, especially for near-uniform scenes.

This study focuses on the algorithmic development and assessment of the proposed methods, excluding practical implementation aspects such as hardware considerations or real-time application deployment. The thesis addresses theoretical approaches to image stitching by developing new algorithms focused on feature-based methods, deliberately excluding deep learning (DL) techniques due to their requirement for larger datasets and high-performance computing hardware. Furthermore, two new evaluation metrics, the spread-overlap, S_o and the recall-to-spread-overlap ratio, RC/S_o , are introduced to provide

a more comprehensive evaluation of feature detection performance. These metrics assess both the accuracy of detected inliers and their spatial distribution in relation to the resulting image stitching quality. The performance of the proposed methods is evaluated using 218 image pairs, including 157 newly generated near-uniform pairs in this thesis.

1.5 Thesis Outline

This thesis presents an in-depth study on the development of multi-scale feature-based image stitching algorithms and evaluation metrics, with a particular focus on addressing the challenges posed by near-uniform scene images. It also includes performance evaluations of the proposed methods using various datasets. The thesis is structured into five chapters, briefly described as follows.

Chapter 2, Literature Review, explores existing research on image stitching, highlighting both its advancements and limitations. This chapter provides an overview of various feature detectors and feature descriptors utilized within the computer vision community, along with a discussion of performance evaluation metrics such as repeatability, recall, and precision. The chapter also examines the relevance of theoretical methods in the era of deep learning. It concludes by identifying the research gaps that this study aims to address and provides a summary of the reviewed literature.

Chapter 3, Methodology, outlines the proposed methods. It describes the first proposed method: Enhanced Partial Differential Equation (ePDE), including discussions on nonlinear diffusion filtering, the improved conductivity function based on the Lorentz factor, and Lorentz-modulated nonlinear scale-spaces. The chapter then covers the second proposed

method: Log-Gabor Feature Transform (LGFT), including 2D Log-Gabor filtering and its scale-spaces. It also explains feature description, the image stitching algorithm, and introduces new performance evaluation metrics: the spread-overlap, S_o and RC/S_o scores.

Chapter 4, Results and Discussion, presents the experimental findings and analyses the performance of the proposed methods. It begins with an overview of the experimental tools and datasets, followed by a discussion of the pre-analysis and parameter settings for both ePDE and LGFT methods. The chapter then compares scale-spaces and evaluates the performance of the proposed methods through quantitative, qualitative, and efficiency comparisons, validating their effectiveness in minimizing stitching distortion and artifacts.

Finally, Chapter 5, Conclusion and Future Works, summarizes the key findings of the research and discusses its limitations. It outlines the study's contributions to the field of image stitching, specifically in enhancing stitching quality in near-uniform scenes. The chapter also suggests directions for future research, identifying potential areas for further investigation and improvement. In essence, this concluding chapter wraps up the thesis by emphasizing the significance of the research and its implications for future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Image stitching aligns and merges multiple images to create a high-resolution wide field of view image. With the advancements in camera technology and the increasing use of drones, image stitching has become essential and is widely applied in various fields, including panoramic street views (Biljecki & Ito, 2021), virtual reality (Yang et al., 2025; Zhou et al., 2023), aerial and satellite mapping (Li & Zhou, 2024; Wei et al., 2024), medical imaging (Mohammadi et al., 2024), and astronomical imaging (Wang et al., 2024), among others. Despite significant progress in image stitching research, not all techniques effectively address the diverse challenges encountered, particularly in handling low-texture or near-uniform scene images (Lyu et al., 2019; Pandey & Pati, 2019; Yang et al., 2025).

To implement an image stitching algorithm, the process typically begins with feature detection to identify interest points across overlapping images (Awad & Hassaballah, 2016). These interest points are then described using robust feature descriptors that encode the local neighbourhood information and ensure consistency under changes in scale, rotation, and illumination (Awad & Hassaballah, 2016). Accurate keypoints matching enables precise homography estimation is essential for image alignment and blending. However, each feature detection method has its strengths and limitations depending on the type of images being processed. Many existing feature detection algorithms rely on complex geometric features for image alignment. However, these features become unreliable in low-resolution or texture-scarce scenarios, making image stitching more challenging (Yang et al., 2025).

Performance evaluation is crucial in image stitching, as it directly impacts the quality, reliability, applicability, and efficiency of the stitched images. While there is no single definitive standard for assessing the performance of image stitching (Bergen & Wittenberg, 2016), several commonly used metrics, such as repeatability (Mikolajczyk & Schmid, 2001; Schmid et al., 2000), recall and precision (Mikolajczyk & Schmid, 2005), are widely applied to assess the effectiveness of feature detectors and descriptors. The Structural Similarity Index (SSIM) introduced by Wang et al. (2004) remains one of the earliest and most widely adopted metrics for general image quality assessment. However, standard image quality metrics are often insufficient for stitched images and require specific assessment approaches such as Stitched Image Quality Assessment (SIQA) methods (Tian et al., 2021).

Deep learning (DL) has emerged as a major trend in computer vision (CV), raising questions about the relevance of theoretical methods that have dominated the field for the past two decades. Reviewing the strengths and limitations of DL methods and exploring how theoretical techniques complement DL approaches in the context of image stitching is an intriguing topic and provides valuable insights. This perspective helps in identifying research gaps, particularly in addressing the challenges of stitching near-uniform images, which is the primary focus of this thesis.

This chapter is organized into eight sections. Section 2.2 provides a brief overview of image stitching classifications and common processing steps. Section 2.3 reviews existing feature detectors, categorized into single-scale, multi-scale, affine-invariant, and region-based detectors. Section 2.4 describes feature descriptors that represent keypoints robustly under geometric and photometric changes. Section 2.5 outlines commonly used evaluation methods, while Section 2.6 compares theoretical techniques with deep learning approaches.

Section 2.7 highlights current research gaps in image stitching. Finally, Section 2.8 presents the chapter summary.

2.2 Image Stitching Review

Modern image stitching techniques can be broadly divided into two main approaches: theoretical methods and deep learning-based methods (see Figure 2.1). Theoretical methods depend on handcrafted features to find correspondences between images, while deep learning approaches utilize neural networks to learn these correspondences directly from data. Since deep learning-based methods are beyond the scope of this thesis, the focus will be on theoretical methods. Theoretical image stitching methods can be further classified into two categories, namely the mosaic and panoramic stitching methods (see Figure 2.2). Mosaic stitching is used to create stitched planar images, whereas panoramic stitching is for producing stitched panoramic images (Fu et al., 2023). Both approaches stem from the same foundational principles. Mosaic stitching involves several key steps, including image registration, alignment, and blending of multiple overlapping images to create a larger field-of-view (FOV) image. In panoramic stitching, after alignment, an additional step of modelling and mapping is applied, resulting a panoramic image based on 3D models (Fu et al., 2023).

Image registration is probably the most critical step in image stitching, as it is nearly impossible to develop a successful stitching algorithm without accurate registration. In the registration step, stitching algorithms can be based on either the spatial domain or the frequency domain, as illustrated in Figure 2.1. Spatial domain-based image stitching uses pixel properties to extract features for registration, and most existing image stitching

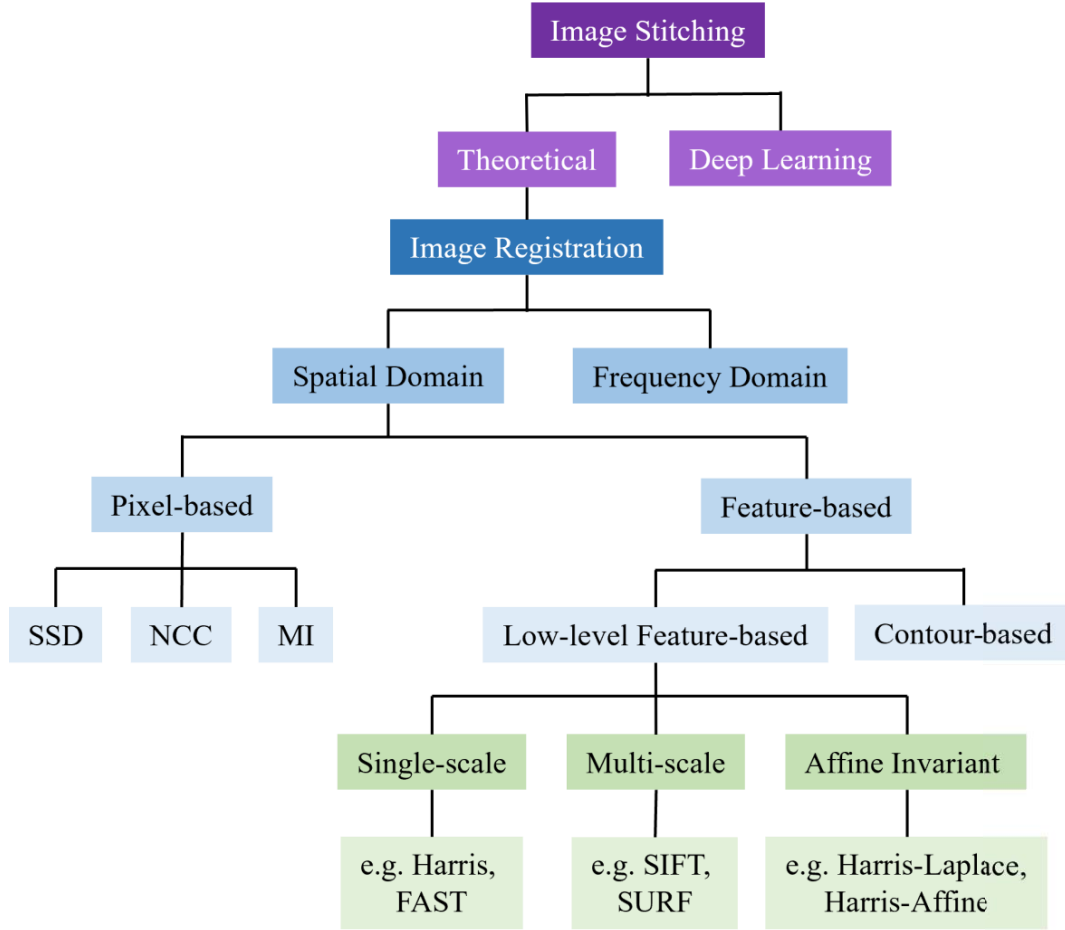


Figure 2.1: Classification of image stitching techniques based on registration methods

methods fall into this category. In contrast, frequency domain-based image stitching involves computing phase correlation properties in the frequency domain to identify the optimal transformation models between images.

Spatial domain-based image stitching can be further divided into pixel-based and feature-based approaches. Pixel-based image stitching relies on pixel-to-pixel matching to determine the geometric transformation between image pairs. This method works by shifting a small window of pixels across the overlapping regions of the images and evaluating how well the pixels align. Common criteria used in pixel-based image stitching include sum of

squared differences (SSD), normalized cross correlation (NCC) and mutual information (MI) stitching (Bergen & Wittenberg, 2016).

SSD measures similarity by summing the squared intensity differences between corresponding pixels, with lower SSD values indicating better alignment. NCC determines similarity by comparing image intensity values, normalized by their means and variances, where higher NCC values suggest better alignment. MI measures similarity based on the amount of shared information between two images, derived from the difference between the sum of individual entropies and the joint entropy of the images.

Unlike pixel-based methods, feature-based image stitching primarily relies on feature extraction algorithms to identify salient features in the images and match them to estimate the geometric transformation between image pairs. This approach involves detecting corresponding interest points in the images and estimating a homography from the correct matches or inliers, allowing the images to be warped and stitched together into a common reference of coordinates. Although feature-based stitching generally produces better results than pixel-based methods, it comes with higher computational costs.

Feature-based stitching techniques can be categorized into two categories: low-level feature-based stitching and contour-based stitching. Common low-level features include semantic structures within the image, such as ridges, edges, corners, and blobs. Ideally, these features should be distinct, evenly distributed across the image, easily detectable in both images, robust against various geometric and photometric changes, and have a high repeatability rate (Ghosh & Kaabouch, 2016). Figure 2.1 illustrates some popular low-level feature extraction methods used in the stitching literature, including their classification into single-scale, multi-scale, and affine-invariant feature detectors.

In contrast, contour-based stitching algorithms focus on extracting mid-level features from images, which are considered more natural than low-level features. Contour features are concerned with identifying the position, orientation, and size of geometric shapes or structures within the image. These features are then matched to compute the geometric transformation based on correspondences. However, contour features typically require more computational resources than low-level features.

Image stitching involves multiple image processing steps, generally categorized into four main stages: image preprocessing, image registration, image alignment, and image blending. Figure 2.2 outlines the common processing steps for both mosaic stitching and panoramic stitching. Depending on the specific challenges of a computer vision (CV) task, real-world scene images may be captured using various types of cameras or imaging devices, such as handheld mobile cameras, endoscopes, microscopes, or telescopes. The data formats from these devices can vary, typically resulting in either planar or fisheye images.

Each input image undergoes a preprocessing step to correct any display effects. Common preprocessing tasks include compensating for exposure and colour, correcting radial distortion, reducing noise and vignetting effects, and segmenting the foreground from the background to extract moving objects (Bergen & Wittenberg, 2016).

After preprocessing, the next crucial step is image registration. In this step, it is vital to extract stable and consistent interest points or keypoints, as their reliability plays a significant role in the effectiveness and convergence of the matching and optimization algorithms used to align images. Incorrect matching of interest points can lead to significant stitching errors, therefore techniques like efficient approximate nearest neighbour search (Muja & Lowe, 2009) and the Random Sample Consensus (RANSAC) algorithm

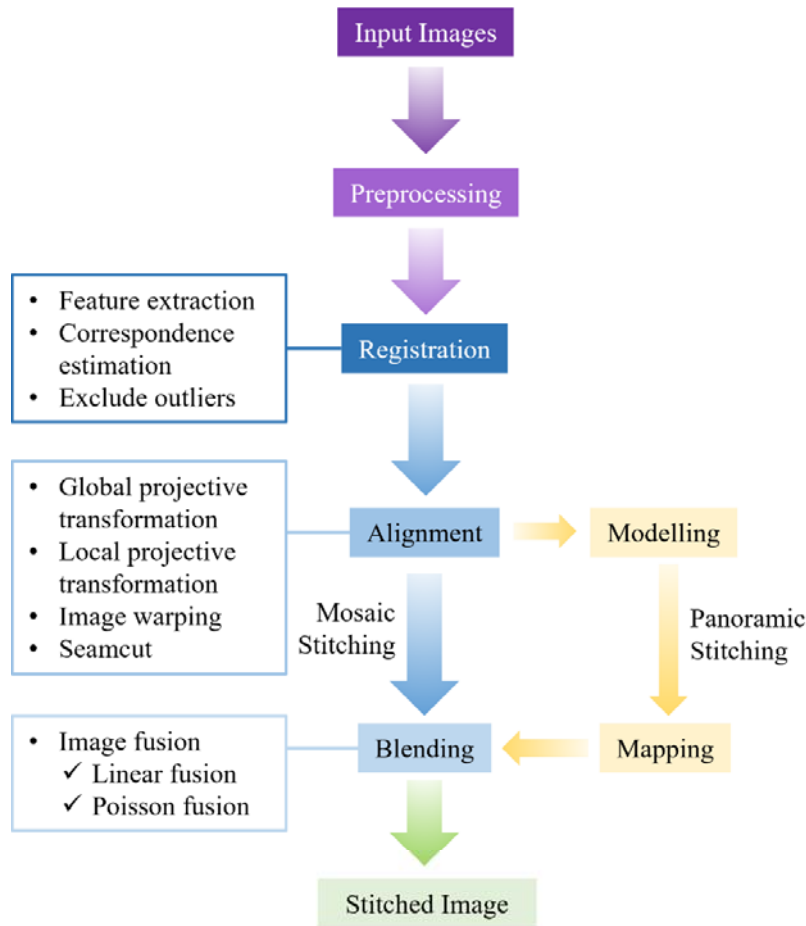


Figure 2.2: Common image stitching processing steps for mosaic stitching and panoramic stitching

(Fischler & Bolles, 1981) are generally employed to remove outliers and accurately calculate the geometric transformation between overlapping images.

In image alignment, several key algorithms play a crucial role, including global and local projective transformations, warping, and seam-cutting techniques. Image alignment involves determining the appropriate homography model that relates pixel coordinates between two images, as thoroughly explained by Szeliski (2007) in his early work. For

optimal image stitching performance, image alignment must be nearly perfect to effectively manage image deformation.

AutoStitch (Brown & Lowe, 2007) introduced one of the first global projective transformation methods. However, it struggles to address parallax issues because it assumes that the overlapping areas between images lie on the same plane and that the camera's optical centre remains fixed, allowing only rotational movement. To improve stitching outcomes, advanced transformation models that utilize multiple homography matrices have been developed. Examples include the dual-homography warping method by Gao et al. (2011) and the Smoothly Varying Affine (SVA) method proposed by Lin et al. (2011), both of which employ multiple affine transformations.

Local projective transformation algorithms, like the mesh-based method, further enhance image stitching capabilities affected by parallax. The As-Projective-As-Possible (APAP) method, introduced by Zaragoza et al. (2014), is the first to bring mesh-based image deformation and alignment strategies into the domain of image stitching. APAP breaks the image into uniform grids and calculates a local homography matrix for each grid. This process aligns the overlapping regions while preserving the structure of non-overlapping areas and minimizing distortion. Building on APAP, subsequent research has led to developments like Shape-Preserving Half Projective (SPHP) (Chang et al., 2014) and Adaptive as-Natural-as-Possible (ANAP) (Lin et al., 2015).

After estimating the homography for the geometric transformation between images, the next step in image stitching is typically warping. Warping involves mapping the pixels of an image according to the transformation defined by the homography, aligning it with the reference image before blending them in the final stitching process. Essentially, warping

adjusts the shape or perspective of the image. Once the images are warped and aligned, the seam-cutting step identifies the optimal location within the overlapping region by comparing pixel similarities. This process eliminates structural and intensity misalignments, ensuring a smooth transition between overlapping regions and minimizing visible seams or artifacts. The GraphCut method, introduced by Boykov et al. (1999), is considered a classic seam-cutting technique. Gao et al. (2013) later proposed a seam-driven method to optimize mesh deformation and improve final stitching results. Since then, various seam-cutting methods have been developed to better handle parallax issues, such as parallax-tolerant stitching that utilizes robust elastic warping (Li et al., 2018) and Seam-Guided Local Alignment (SEAGULL) (Lin et al., 2016).

Due to differences in equipment parameters, acquisition viewpoints, and lighting conditions, the final stitched image often exhibits inconsistent light intensity and colour after multiple images are projected onto the same plane through geometric transformation. This is where blending comes into play, combining and fusing the images according to specific fusion criteria. Blending is the final step in image stitching and typically involves image fusion. Image fusion can be generally categorized into linear fusion and Poisson fusion. Linear fusion combines images in overlapping regions based on the principle of linear superposition, with feather blending applying a linearly decreasing weight from the stitching seam to the edge of the overlapping area. In contrast, Poisson fusion is a multiband fusion technique that smoothly merges overlapping image regions, where the weights for each layer are based on their respective scaling, although these weights may not always be optimal.

Further reading on image stitching can be found in Yang et al. (2025), Mohammadi et al. (2024), Zhang & Xiu (2023), Gómez-Reyes et al. (2022), El Abbadi et al. (2021), Jagadeeswari et al. (2021), Paul & Pati (2021), and Wang & Yang (2020).

2.3 Feature Detectors

Over the past few years, image feature detectors and descriptors have become essential tools in numerous computer vision applications, including image retrieval, image classification, object recognition, object tracking, augmented reality, 3D reconstruction, as well as image registration and stitching. In the context of image stitching, feature detection is especially crucial because the success of the stitching process largely relies on the ability to identify stable and distinct features within the image. Feature detection focuses on detecting significant key structures and elements in images, such as points, curves, lines, and regions, that represent the semantic content of the image, such as corners, edges, ridges, and blobs (Li et al., 2015). These image structures can be categorized into corner features, blob features, line or edge features, and morphological region features (Ma et al., 2021). Typically, these features are represented as interest points, like corners and blobs, or keypoints that are usually associated with descriptors for the matching step of image stitching. The term 'feature points' is usually used to represent both interest points and keypoints.

The ultimate goal of a feature detector is to extract features from images with high stability in an effective and efficient manner. According to Awad & Hassaballah (2016) and Joshi & Patel (2020), an ideal feature detector should possess qualities such as robustness, repeatability, accuracy, generality, efficiency, and quantity. In other words, the algorithm

should consistently detect the same features regardless of different geometric transformations, be repeatable under varying viewing conditions, accurately locate image features at the same pixel locations, be applicable across different tasks, execute quickly enough for real-time applications, and detect most, if not all, relevant features in the image. While many feature detectors exist in the literature, there is currently no single algorithm that is ideal or appropriate for every application. Feature detectors can be generally categorized into single-scale detectors, multi-scale detectors, and affine invariant detectors (Awad & Hassaballah, 2016). Figure 2.3 illustrates the classification of feature detectors in a hierarchical form.

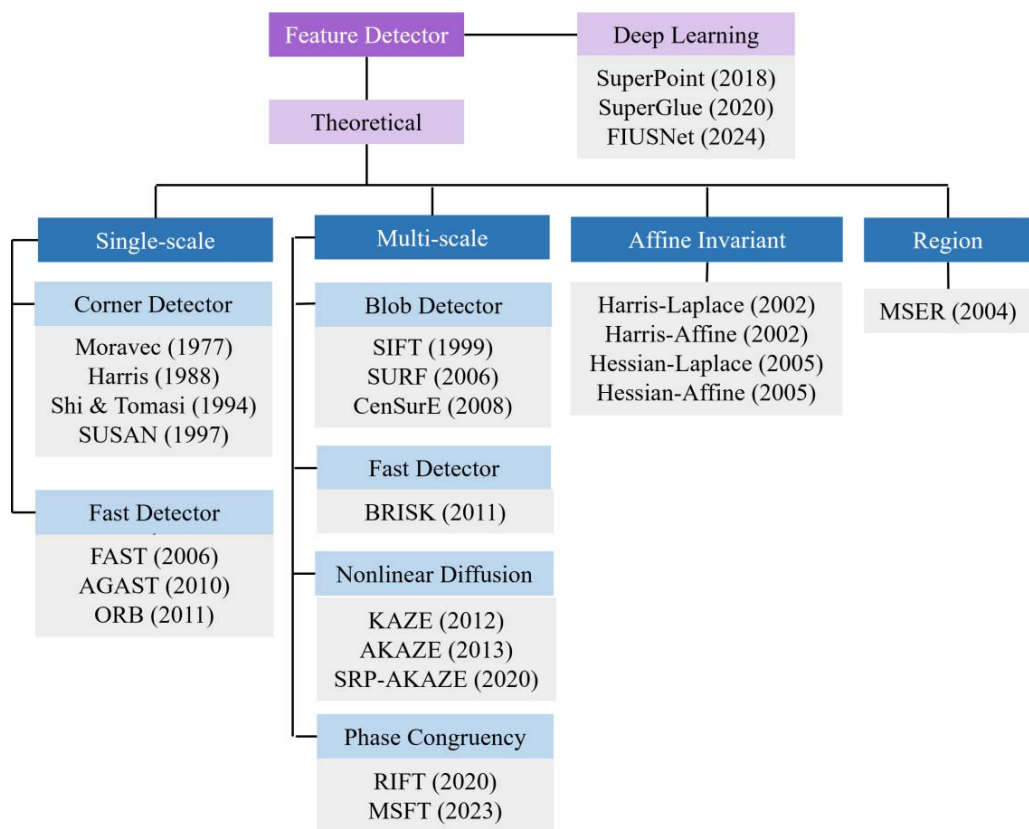


Figure 2.3: Classification of common feature detection techniques

The earliest study of feature detectors dates back to Moravec's work on single-scale corner feature detection (Moravec, 1977). Although Moravec's detector is straightforward and easy to implement, it does not handle rotation well. This limitation is addressed by the Harris detector, which combines corner and edge detection to tackle problems related to anisotropy and computational complexity (Harris & Stephens, 1988). However, the Harris detector does not exhibit scale invariance, reducing its effectiveness in matching images of varying sizes. Shi & Tomasi (1994) enhanced the Harris detector, achieving better tracking performance by ensuring more even feature distribution and improving localization accuracy.

Other notable single-scale corner detectors are the Smallest Univalue Segment Assimilating Nucleus (SUSAN) that aims to minimize image noise (Smith & Brady, 1997), and the Features from Accelerated Segment Test (FAST), which is tailored for real-time video processing (Rosten & Drummond, 2005, 2006). Additionally, the Adaptive and Generic Accelerated Segment Test (AGAST) builds on FAST by offering faster, adaptive, and more generic segment testing (Mair et al., 2010). The Oriented FAST and Rotated BRIEF (ORB) combines the FAST feature detector with the Binary Robust Independent Elementary Features (BRIEF) descriptor (Rublee et al., 2011). Although single-scale detectors are typically invariant to transformations like translation, rotation, noise and changes in illumination, they have difficulty handling scaling issues.

Consequently, multi-scale detectors have emerged to reliably extract distinctive features despite changes in scale. Techniques based on second-order partial derivatives, such as Laplacian of Gaussian (*LoG*), Difference of Gaussian (*DoG*), and Determinant of Hessian (*DoH*), are commonly employed in multi-scale feature detection to identify blob-like

structures at various scales within an image. These methods are prominently utilized in the well-known Scale-Invariant Feature Transform (SIFT) algorithm (Lowe, 1999, 2004). SIFT primarily employs *DoG* is derived from the difference of *LoG* filters to identify local extrema, while important features are selected using *DoH*. Since then, many researchers have been drawn to multi-scale feature-based methods, applying them in diverse areas of computer vision (CV), such as feature detection, texture analysis, image segmentation, image retrieval, and object recognition.

Several variants of SIFT have been crafted, such as Principal Components Analysis SIFT (PCA-SIFT) (Ke & Sukthankar, 2004), Colored SIFT (CSIFT) (Abdel-Hakim & Farag, 2006), Affine-SIFT (ASIFT) (Morel & Yu, 2009), Dense SIFT (Wang et al., 2010) and Bilateral Filter SIFT (BFSIFT) (Wang et al., 2012). The Speeded Up Robust Features (SURF) improves on SIFT by approximating the Hessian matrix-based detector through the use of Haar wavelet transforms and integral images, thereby simplifying the computation of second-order partial derivatives (Bay et al., 2006). Center Surround Extremas (CenSurE) employs simplified bi-level Gaussian approximation kernels as center-surround filters, allowing rapid real-time computation with integral images (Agrawal et al., 2008). Binary Robust Invariant Scalable Keypoints (BRISK) features an innovative scale-space detector based on FAST, paired with a binary descriptor string. It uses a customizable circular sampling pattern to compute intensity comparisons for the bit-string descriptor (Leutenegger et al., 2011).

While SIFT and SURF use Gaussian scale-spaces and are highly effective at detecting interest points, Gaussian smoothing has its limitations. It blurs both image details and noise, which can reduce the accuracy and distinctiveness of interest points by not

preserving object boundaries (Alcantarilla et al., 2012). To address this issue, nonlinear diffusion filtering can be used to create smooth scale-spaces while preserving natural boundaries of regions and objects (Perona & Malik, 1990). Methods like KAZE (Alcantarilla et al., 2012), Accelerated KAZE (AKAZE) (Alcantarilla et al., 2013), and Sparse Random Projection AKAZE (SRP-AKAZE) (Li et al., 2020) utilize nonlinear diffusion filtering to enhance the detection of local extrema across different nonlinear scale levels.

For significant affine transformations, scale-invariant detectors may fail because scale changes can differ in each direction rather than being uniform. To address this, advanced affine invariant detectors have been developed. Notably, Mikolajczyk & Schmid (2002, 2004) combined the Harris detector with the Laplacian kernel to create detectors for both scale and affine feature detection, such as the Harris-Laplace and Harris-Affine detectors. Their approach adjusts the position, scale, and shape of a point neighbourhood to attain affine invariance. This concept is similarly implemented in the Hessian-based detector, leading to the development of the Hessian-Laplace and Hessian-Affine detectors, as thoroughly analysed by Mikolajczyk & Schmid (2005).

Other approaches, such as Maximally Stable Extremal Regions (MSER), use extremal regions to extract covariant areas from images for wide-baseline matching (Matas et al., 2004). Recent multi-scale detectors designed for stitching multimodal images include the Radiation-variation Insensitive Feature Transform (RIFT) (Li et al., 2020) employs phase congruency for feature detection, and a maximum index map (MIM) for feature description. The Multi-scale Structural Feature Transform (MSFT) (Hu et al., 2023) is another example, utilizing a *DoG* image pyramid to identify scale-invariant feature points and address nonlinear radiation distortions.

In recent years, deep learning-based feature detectors have attracted significant research attention, especially with the advent of SuperPoint features (Detone et al., 2018) and SuperGlue (Sarlin et al., 2020) and FIUSNet (Ni et al., 2024). Zheng et al. (2018) explore the connection between SIFT and CNN-based methods. While many deep learning detectors have shown strong performance, the selection of a detector typically relies on specific computer vision tasks, considering the subtle differences between them. The relevance of theoretical methods versus deep learning approaches in image stitching presents an important and engaging topic for discussion. The following section focuses on SIFT and SURF as they form the foundation of interest point detection and employ non-maxima suppression, an approach adopted in the proposed methods.

2.3.1 Scale Invariant Feature Transform (SIFT) Detector

Introduced by Lowe (2004), SIFT is a widely recognized algorithm in computer vision for detecting stable keypoints in images using a cascade filtering approach. It is particularly effective for tasks like object recognition, image stitching, and 3D reconstruction due to its robustness against changes in scale, rotation, and illumination. SIFT identifies keypoints by detecting local extrema in a scale-space representation, a concept previously explored by Lindeberg (1993) and Witkin (1984) for detecting scale-invariant features.

The scale-space is created by applying Gaussian kernels, $G(x, y, \sigma)$, to an image, $I(x, y)$, at multiple scales. The Difference of Gaussian (*DoG*) images are subsequently derived by subtracting adjacent scales, where each scale is distinguished by a factor k , as indicated in the following expression.

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \quad \text{Equation 2.1}$$

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad \text{Equation 2.2}$$

where $D(x, y, \sigma)$ represents the *DoG* scale-spaces of an image, $L(x, y, \sigma)$ denotes the

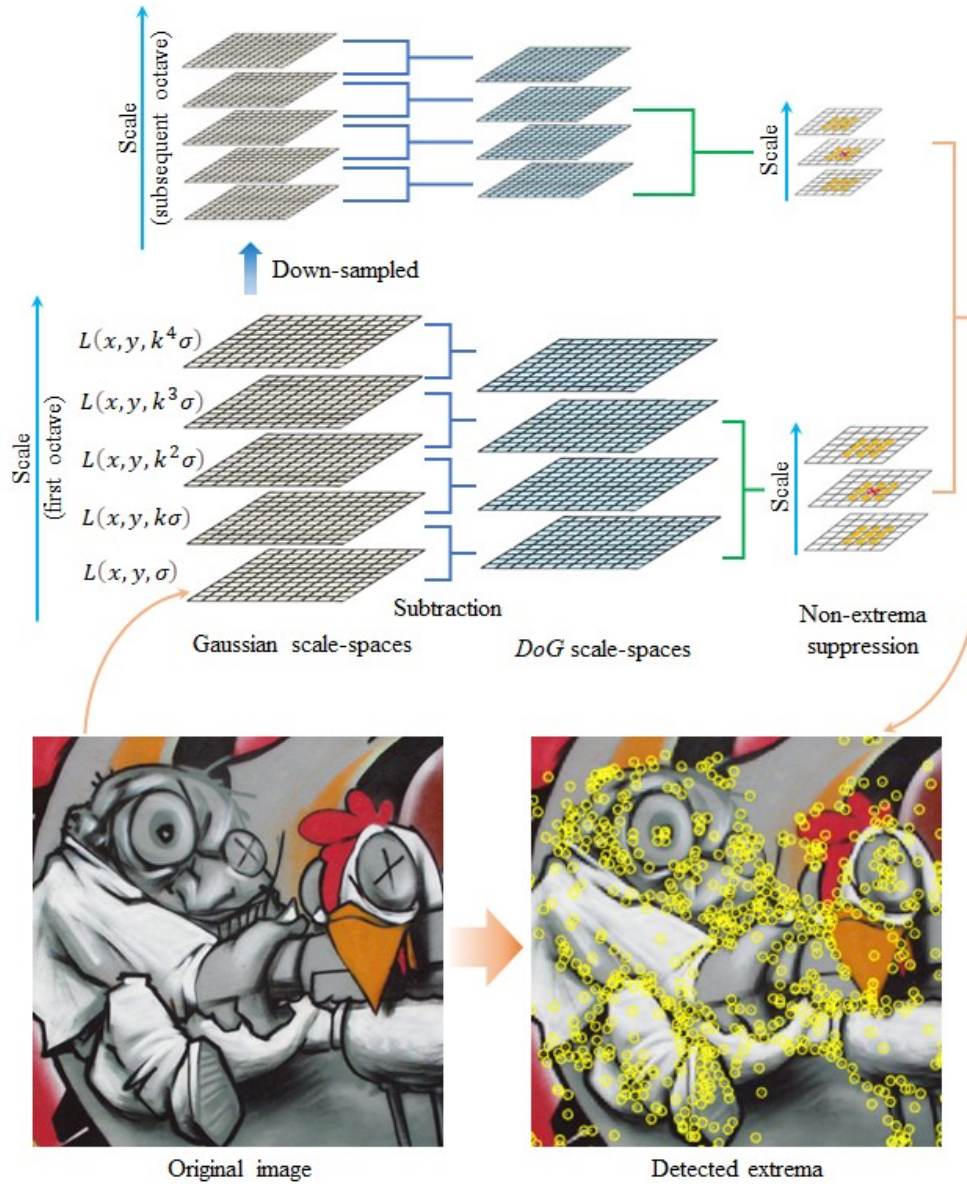


Figure 2.4: *DoG* representations for local extrema detection are generated by subtracting adjacent Gaussian images in each octave. Each subsequent octave involves down-sampling the Gaussian images by a factor of 2 (Lowe, 2004)

Gaussian scale-space of the image, and asterisk $*$ indicates the convolution operation in both the x and y directions. The standard deviation of the Gaussian, σ controls the scale-space by adjusting the amount of blurring applied to the image. In scale-space theory (Lindeberg, 1994), computation of Laplacian of Gaussian (LoG) is time-consuming. To accelerate this process, $D(x, y, \sigma)$ closely approximates the scale-normalized LoG , thereby enabling the efficient detection of stable keypoints. Figure 2.4 illustrates the detection of local extrema in the DoG scale-spaces of an image.

To detect local extrema in $D(x, y, \sigma)$, each pixel is compared against its 26 neighbouring pixels within 3×3 regions across the current and adjacent scales, as illustrated in Figure 2.4. This process, known as non-extrema suppression, identifies a candidate point only when it is either greater than or less than all its neighbouring points. After processing an entire octave, the Gaussian image is reduced in size by a factor of 2. Specifically, the Gaussian image is chosen based on a σ value that is twice the initial σ used in the current octave. Typically, this is the third image in a set of five within an octave (as each image's σ is progressively increased by a factor of $\sqrt{2}$). The process of detecting local extrema is then repeated for the new octave. While SIFT is invariant to scale, illumination changes, and rotation, it tends to be computationally intensive.

2.3.2 Speeded-Up Robust Features (SURF) Detector

SURF, crafted using scale-space theory, is a feature detector that is invariant to both scale and rotation. Instead of using Gaussian scale-spaces like SIFT, SURF approximates the Laplacian of Gaussians (LoG) using box-type convolution filters. It also employs a fast approximation of the Hessian matrix by utilizing an integral image, which allows for the

estimation of local maxima across various scale-spaces (Bay et al., 2008). This implementation allows SURF to be a faster alternative to SIFT. Given an input image, $I(x, y)$, the integral image is generated by summing all the pixels from a point, $p(x, y)$ back to the origin $(0,0)$, forming a rectangular region defined as follows:

$$I_{\Sigma}(p) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(x, y) \quad \text{Equation 2.3}$$

With an integral image, calculating the area of an upright rectangular region requires just three additional mathematical operations, regardless of the rectangle's size, as shown in Figure 2.5 (Evans, 2009). This allows for a simple and efficient computation of the sum of pixel intensities for any upright rectangular region using

$$\Sigma = A + D - (C + B) \quad \text{Equation 2.4}$$

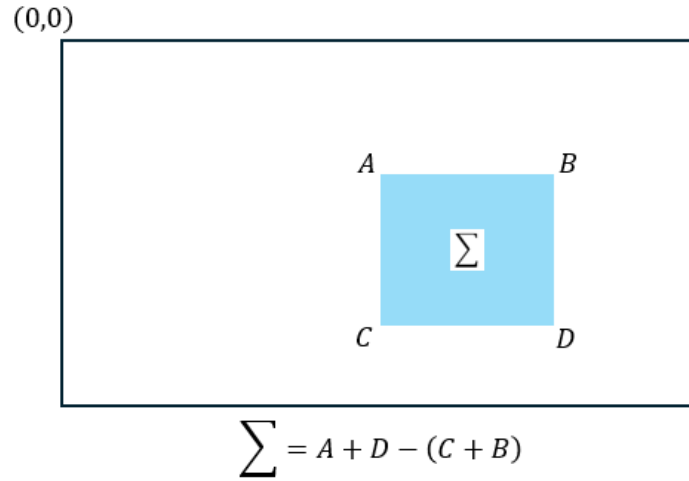


Figure 2.5: Calculating area using integral images (Evans, 2009)

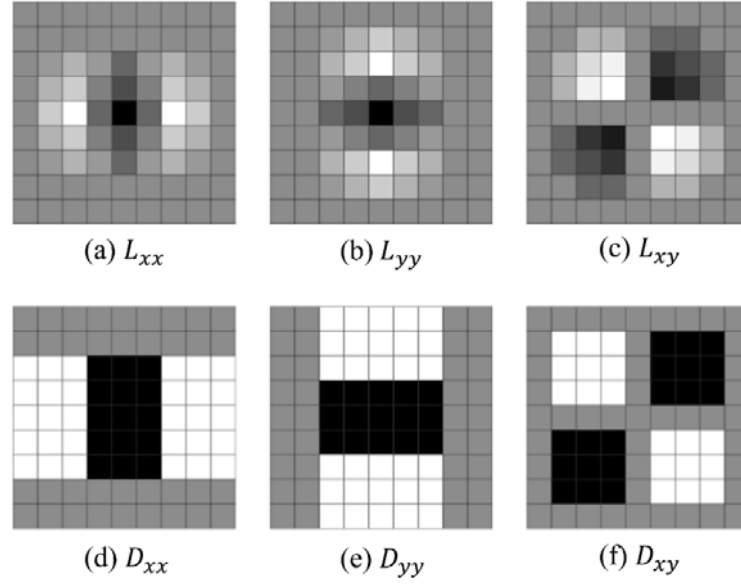


Figure 2.6: Approximation of the Laplacian of Gaussian. The discretised and cropped Laplacian of Gaussians: (a) L_{xx} , (b) L_{yy} and (c) L_{xy} . The corresponding weighted box filter approximations: (d) D_{xx} , (e) D_{yy} and (f) D_{xy} (Bay et al., 2008)

The Hessian matrix of the image $I(x, y)$ at any pixel point (x, y) with scale σ is defined as follows:

$$H(x, y, \sigma) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix} \quad \text{Equation 2.5}$$

where $L_{xx}(x, y, \sigma)$, $L_{yy}(x, y, \sigma)$ and $L_{xy}(x, y, \sigma)$ represent the convolution of the image I at point (x, y) with second-order Gaussian derivatives $\frac{\partial^2}{\partial x^2} G(x, y, \sigma)$, $\frac{\partial^2}{\partial y^2} G(x, y, \sigma)$ and $\frac{\partial^2}{\partial x \partial y} G(x, y, \sigma)$, respectively. These derivatives are known as the Laplacian of Gaussians.

While Gaussians are ideal for scale-space analysis, they must be discretized and cropped in practical applications, as shown in the top row of Figure 2.6, labelled as (a) L_{xx} , (b) L_{yy} , and (c) L_{xy} . To increase performance, the Laplacian of Gaussians is approximated

using simple weighted box filters, represented as (d) D_{xx} , (e) D_{yy} , and (f) D_{xy} . These approximations can be computed with minimal computational cost using integral images. Bay et al. (2008) uses these approximated Laplacians to compute the Hessian determinant using the following mathematical expression.

$$\det(H_{approx}) = D_{xx}D_{yy} - (0.9D_{xy})^2 \quad \text{Equation 2.6}$$

At the pixel point (x, y) in an image, the approximated Hessian determinant represents the blob response. This response is captured in a blob response map across different scales using a 9×9 pixels box filter (see Figure 2.6). This box filter corresponds to the smallest scale used for generating the blob response maps. Unlike SIFT, which reduces the image size iteratively, SURF scales the filter up at a constant cost using integral images, as illustrated in Figure 2.7. Subsequently, non-maxima suppression is performed within a $3 \times 3 \times 3$ neighbourhood. This is followed by interpolation of local maxima in the scale-space to identify interest points, along with their locations and scale values. However, SURF has limitations in handling viewpoint and illumination changes and is less sensitive to noise.

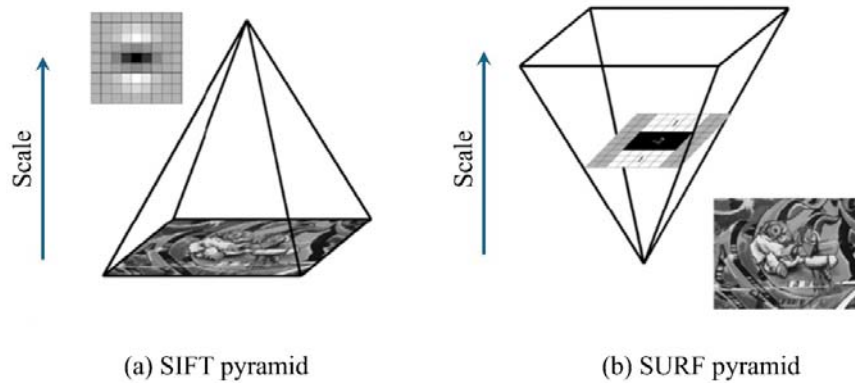


Figure 2.7: Scale-spaces pyramids: (a) SIFT varies the image size and applies Gaussian smoothing to the resulting layers. (b) SURF retains the image size and changes the filter size instead (Bay et al., 2008)

2.4 Feature Descriptors

Registering a pair of images is a critical step in the image stitching process. This involves using descriptors to provide the necessary information for establishing correspondences between interest points or regions, aligning the images to a common coordinate system. Essentially, this means transforming the images to match the spatial orientation of a reference image. According to Szeliski (2022), image registration algorithms can generally be categorized into two primary categories:

- i. Pixel-based approaches optimize a criterion calculated across all pixels in the overlapping areas of two images. Typical criteria include the sum of squared differences (SSD), normalized cross-correlation (NCC), and mutual information (MI) (Bergen & Wittenberg, 2016).
- ii. Feature-based approaches identify higher-level features that encode information about the local neighbourhood around interest points. These features are often represented as vectors and can be based on properties such as intensity values, gradient patterns, texture, colour, or other local characteristics.

Notable feature descriptors include SIFT (Lowe, 2004), SUFT (Bay et al., 2006), as well as Gradient Location-Orientation Histogram (GLOH) (Mikolajczyk & Schmid, 2005), an extension of SIFT. Some feature descriptors prioritize computational speed and minimize storage requirements by benefiting from the binary representations. Examples include Binary Robust Independent Elementary Features (BRIEF) (Calonder et al., 2010), BRISK (Leutenegger et al., 2011), Fast Retina Keypoint (FREAK) (Alahi et al., 2012), and Modified Local Difference Binary (M-LDB) (Alcantarilla et al., 2013). A recent approach, Maximum Index Map (MIM), has been developed for efficient multimodal image stitching (Li et al.,

2020). The following section will provide a detailed discussion of SIFT and SURF that form the basis for the methods proposed in this thesis.

2.4.1 SIFT Descriptor

Once interest points are identified from Difference of Gaussian (*DoG*) images, each point is assigned a dominant orientation to maintain rotation invariance (Lowe, 2004). First, the gradient magnitude, $m(x, y)$ and orientation, $\theta(x, y)$ are computed using pixel differences in each Gaussian image, $L(x, y)$ at the interest point's scale. Within a region around the interest point, a histogram of gradient orientations is created with 36 bins covering 360 degrees. The dominant orientation of each interest point is determined by identifying the highest peak in this histogram and any values within 80% of this peak.

The next step is to create the SIFT interest point descriptor. This process starts with sampling gradient magnitudes and orientations at each point within a 16×16 region around the interest point, as shown in Figure 2.8(a). The gradient magnitude of each sample point is weighted using a Gaussian window, depicted by the overlaid circle in Figure 2.8(a). To achieve orientation invariance, both the coordinates and gradient orientations of the descriptor are rotated to align with the dominant orientation of the interest point. The gradient magnitudes are then aggregated into orientation histograms, summarizing the information across 4×4 subregions, as illustrated in Figure 2.8(b). In these histograms, the length of each arrow represents the total gradient magnitudes in that direction within the respective region.

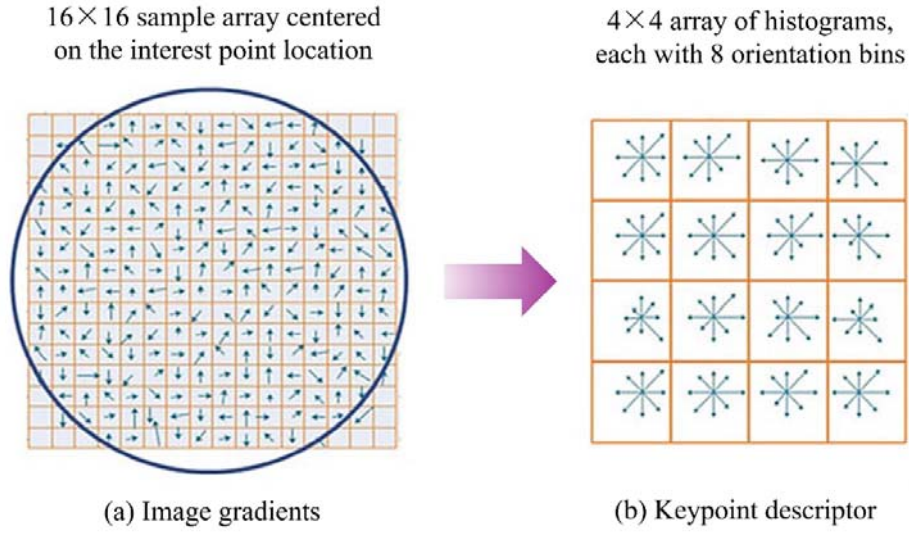


Figure 2.8: Construction of a 128-element SIFT feature vector derived from a 16×16 sample grid centered around the interest point position (Awad & Hassaballah, 2016)

The interest point descriptor is then formed as a vector that includes the values from all the entries of the orientation histogram (i.e., the length of the arrows in Figure 2.8(b)). Specifically, the feature vector for each interest point is constructed from a 4×4 array of histograms, with each histogram having 8 orientation bins, as shown in Figure 2.8(b). This results in a total of $4 \times 4 \times 8 = 128$ elements. The feature vector is then normalized to a unit length to ensure it remains invariant to affine variations in illumination. The process of constructing the SIFT feature vector is complex and results in increased computational time.

2.4.2 SURF Descriptor

The extraction of SURF descriptors involves two main steps: orientation assignment and descriptor construction. To create rotation-invariant descriptors, it is crucial to estimate the dominant orientation within a local neighbourhood surrounding the interest point. This

involves calculating the Haar wavelet responses, or first-order derivatives, d_x and d_y , in the x and y directions within a circular area of radius 6σ centred on the detected interest point, using a sampling step of size σ . Haar wavelets are simple filters used with integral images to efficiently and robustly compute gradients in the x and y directions. They are scale-dependent with a size of 4σ .

The wavelet responses, represented as vectors in a vector space, are weighted using a Gaussian function centred on the interest point. To determine the dominant orientation, the responses in the x and y directions are summed within a sliding circular sector that spans an angle of $\pi/3$. The combined horizontal and vertical responses produce a local orientation vector. The vector with the greatest length represents the dominant orientation. Figure 2.9 illustrates the process of orientation assignment.

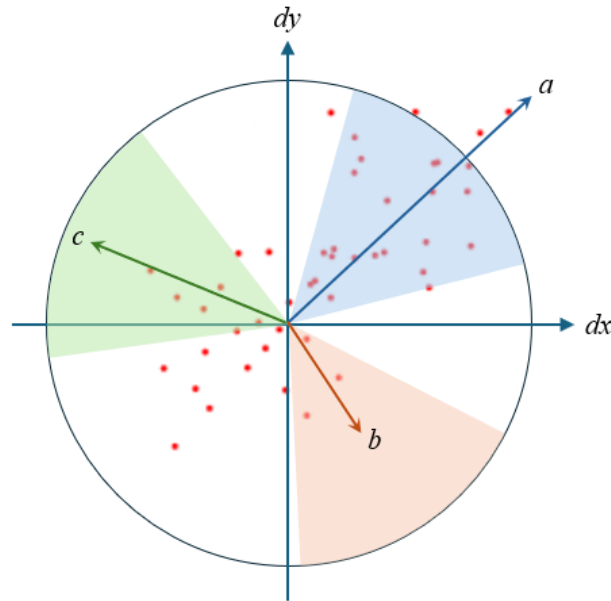


Figure 2.9: As the circular sector rotates around the origin, moving from blue to red to green areas, the Haar responses (marked as red dots) within each sector are summed to produce vectors a , b and c . The longest of these vectors represents the dominant orientation

Like SIFT, the initial step in constructing a SURF descriptor involves creating a square region centred on the interest point. This region, oriented according to the interest point's dominant orientation, has a size of 20σ , and all subsequent calculations are based on this orientation. The region contains pixels used to build the descriptor vector. To preserve important spatial information, the region is split into 4×4 subregions. Within each subregion, Haar wavelet responses are calculated at 5×5 uniformly spaced sample points within a size of 2σ , as shown in Figure 2.10. For simplicity, the Haar wavelet responses in both the x and y directions are represented as d_x and d_y , respectively. The wavelet responses, d_x and d_y , along with their absolute values, $|d_x|$ and $|d_y|$, are summed for each subregion to form the subregion descriptor vector, $v_{subregion}$ as

$$v_{subregion} = \left\{ \sum dx, \sum dy, \sum |dx|, \sum |dy| \right\} \quad \text{Equation 2.7}$$

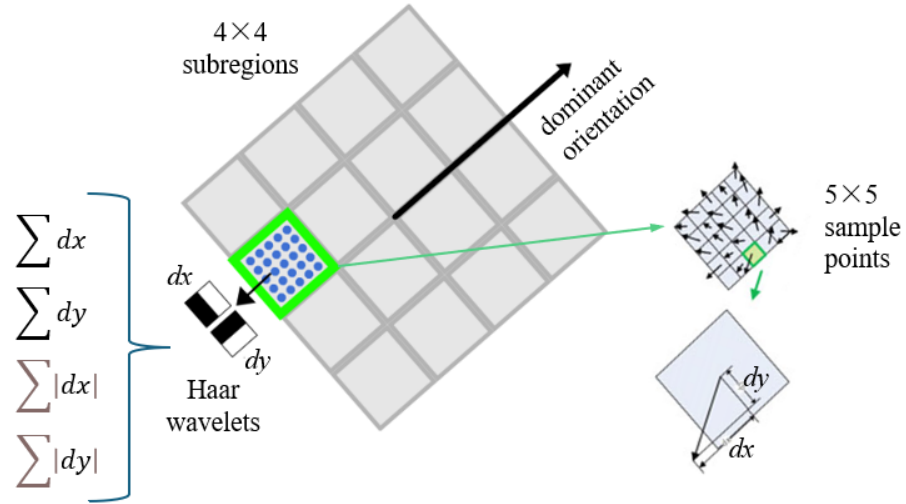


Figure 2.10: Construction of a 64-element SURF feature vector derived from a 4×4 grid of subregions (Awad & Hassaballah, 2016; Bay et al., 2008)

Each subregion provides four values to the descriptor vector. Given that there are 4×4 subregions, this results in an overall descriptor vector with $4 \times 4 \times 4 = 64$ elements. To achieve invariance to contrast, this descriptor vector is normalized to a unit length. Consequently, the resulting SURF descriptor remains invariant to changes in rotation, scale, brightness, and contrast.

2.5 Performance Evaluation

There is no universally accepted standard for assessing the performance of stitching algorithms (Bergen & Wittenberg, 2016). Additionally, there is a lack of publicly available databases, especially for near-uniform scenes, that provide ground truth for comparing different stitching approaches. Because creating ground truth data for stitching results is challenging, many researchers use their own methods for evaluation with non-standardized datasets. Some even present their results based solely on visual impressions of quality without quantitative measures (Bergen & Wittenberg, 2016). Generally, evaluating the performance of stitching algorithms involves considering several factors, including registration error, image structure similarity, seamless transitions between images, and processing speed (Bergen & Wittenberg, 2016).

Registration error can be assessed by comparing calculated target homographies to reference homographies derived from manually established correspondences. Alternatively, it can be measured by comparing the stitched image with a point grid printed on the target image, which acts as a reference phantom. The points can be automatically extracted and compared with the reference positions (Bergen & Wittenberg, 2016). Regarding structural similarity, the Structural Similarity Index (SSIM) introduced by Wang et al.(2004) is one of

the earliest metrics for assessing image quality, but it does not effectively capture stitching artifacts in stitched or panoramic images. According to Bergen & Wittenberg (2016), there is currently no established measure for quantifying the smoothness of seams between images; it is often assessed through visual inspection. Processing speed can be quantitatively evaluated to assess the real-time stitching capability.

Since stitched images are composed of several individual images, it is important to assess their quality, also known as Stitched Image Quality Assessment (SIQA), to optimize stitching algorithms and evaluate the visual quality of panoramic or stitched images. This task is more complex than traditional image quality assessment (IQA) due to the high resolution and distortion complexity inherent in stitched images. Over the past decade, several SIQA methods have been developed, including High-Frequency Information Structural Similarity (HFI-SSIM) (Qureshi et al., 2012), Stitched Image Quality Evaluator (SIQE) (Madhusudana & Soundararajan, 2019), Deep Learning based No-Reference Stitched Image Quality Assessment (DLNR-SIQA) (Ullah et al., 2020) and Patch Registration and Bidimensional Feature Aggregation (PRBFA) (Zhou et al., 2023). While these methods effectively address various distortion attributes, they often fall short in handling near-uniform scene images. Additionally, they fail to establish a connection between the attributes of feature points and IQA. Effective SIQA is crucial for evaluating various stitching methods and optimizing their design.

Common evaluation criteria for assessing the effectiveness of feature detectors and descriptors include repeatability (Mikolajczyk & Schmid, 2001; Schmid et al., 2000) as well as recall and precision (Mikolajczyk & Schmid, 2005). These metrics are widely used and accepted for evaluating advanced feature detectors and descriptors including SIFT, SURF,

BRISK, KAZE, and AKAZE. Feature detectors are typically evaluated based on repeatability, while feature descriptors are assessed using precision-recall graphs. Further details on these criteria and their limitations in evaluating image stitching quality are discussed in the following section.

2.5.1 Repeatability Metric

Repeatability measures a detector's ability to identify corresponding points across different images, regardless of variations in imaging conditions such as camera parameters, position relative to the scene, and illumination. For two images of a planar scene, an interest point, $\tilde{P}_i$ detected in image I_i should correspond to a similar interest point $\tilde{P}_j$ repeated in image I_j . Typically, the detected point $\tilde{P}_j$ will not be exactly at position $\tilde{P}_i$, but rather in its vicinity. Interest points are considered correspondences if the location error does not exceed $\epsilon (= 1.5)$ and the overlap error covered by the point region is under 40% (Mikolajczyk & Schmid, 2004). Consequently, the number of corresponding points, C is defined as

$$C = \{(\tilde{P}_i, \tilde{P}_j) | \text{dist}(H_{ij}\tilde{P}_i, \tilde{P}_j) < \epsilon\} \quad \text{Equation 2.8}$$

where H_{ij} is the homography between images I_i and I_j .

According to Schmid et al. (2000), the number of detected points may vary from one image to another, with only the fewest interest points can be reliably repeatable. Therefore, repeatability, R is defined as the ratio of the corresponding interest points identified between images I_i and I_j to the fewest number of detected interest points, expressed as follows:

$$R = \frac{C}{\min(n_i, n_j)} \quad \text{Equation 2.9}$$

where n_i and n_j are the number of interest points detected in the overlapping areas of images I_i and I_j respectively. The repeatability shall be ranged $0 \leq R \leq 1$.

2.5.2 Recall Metric

The effectiveness of an interest point's descriptor is judged by how accurately it represents a scene feature for precise matching. This assessment is determined by the count of correct matches or inliers, M , in relation to the total number of correspondences, C , obtained between a pair of images. This metric, known as recall, RC represents the ratio of inliers to the total number of corresponding points visible in the overlapping regions of the scene images. It is mathematically defined as

$$RC = \frac{M}{C} \quad \text{Equation 2.10}$$

In this thesis, a different approach is used to determine the number of inliers, M , by employing the M-estimator Sample Consensus (MSAC) algorithm (Torr & Zisserman, 2000), which is a variant of the Random Sample Consensus (RANSAC). The number of corresponding points, C is determined using an efficient approximate nearest neighbour search method developed by Muja & Lowe (2009). Consequently, recall, RC is defined as

$$RC = \frac{|\{(\tilde{P}_i, \tilde{P}_j) | \text{dist}(H_{ij}\tilde{P}_i, \tilde{P}_j) < \epsilon_m\}|}{|\{(D_i, D_j) | \text{Dist}(D_i, D_j) < \epsilon_c\}|} \quad \text{Equation 2.11}$$

where ϵ_m represent the upper limit allowable distance error between an interest point $\tilde{P}_i$ and its corresponding point $\tilde{P}_j$, with H_{ij} being the global geometric transformation between images I_i and I_j . The points $\tilde{P}_i$ and $\tilde{P}_j$ are considered a perfect match if their distance, $dist(H_{ij}\tilde{P}_i, \tilde{P}_j)$ is less than ϵ_m (typically 1.5 pixels). In addition, let ϵ_c be the matching threshold for the descriptor vectors D_i and D_j associated with $\tilde{P}_i$ and $\tilde{P}_j$, respectively. $\tilde{P}_i$ and $\tilde{P}_j$ are considered correspondences if the distance between their descriptors, $Dist(D_i, D_j)$ is less than ϵ_c . $Dist$ represents the Euclidean distance for non-binary feature matching, with a default value of $\epsilon_c = 10$, or the Humming distance for binary features matching, with a default value of $\epsilon_c = 100$. The notation $|\dots|$ represents the number of elements within a set.

2.5.3 Precision Metric

When comparing descriptors from reference images to those from target images, it is possible to identify not only correct matches but also false matches. According to Ke & Sukthankar (2004), a correct match occurs when two keypoints align with the same physical location, and can be confirmed either through ground truth in labelled images or by utilizing known transformations in synthetic image deformation tests. In contrast, a false match takes place when the interest points originate from different physical locations. Mikolajczyk & Schmid (2005) define the ratio of false matches to the total number of matches as being referred to $1 - Prec$.

$$Prec = 1 - \frac{FM}{CM + FM} \quad \text{Equation 2.12}$$

where $Prec$ is the precision, CM refers to the number of correct or true matches, and FM indicates the number of incorrect or false matches. Additionally, CM can be determined by

$$CM = C \cdot RC \quad \text{Equation 2.13}$$

whereas FM can be determined by

$$FM = \frac{C \cdot RC \cdot (1 - Prec)}{Prec} \quad \text{Equation 2.14}$$

The results are typically presented in graphs that compare recall, RC with $1 - Prec$. For any given precision, $Prec$, an ideal descriptor would reach a recall, RC value of 1. This indicates that the descriptor captures all the correct matches while minimizing the number of false matches.

2.5.4 Limitations

Although metrics like repeatability, recall, and precision are commonly used to evaluate feature detectors and descriptors, they fail to account for how detected inliers affect the overall quality of image stitching. Even with high repeatability and recall, there is no guarantee that the inliers will be sufficient in terms of both quantity and spatial distribution to yield high-quality image stitching results. In other words, good image stitching quality requires a sufficient number of stable, distinct inliers that are evenly distributed across the image. Without this, the overlapping scenes may not align perfectly, leading to visible artifacts in the stitched image. Furthermore, the lack of comprehensive public databases, particularly for near-uniform scenes that provide ground truth for comparing stitching quality, is a significant issue. Creating such ground truth data for near-uniform scenes is particularly difficult.

Structural Similarity Index (SSIM) is a useful and widely accepted metric for assessing image quality. However, it is not specifically designed for evaluating image stitching quality and may not be sensitive enough to detect stitching artifacts in stitched or panoramic images. Stitched Image Quality Assessment (SIQA) is another popular metric specifically designed for evaluating stitched images. However, like SSIM, SIQA may not be sensitive to all types of stitching artifacts, particularly those that are subtle and difficult for human perception to detect. These subtle artifacts are often found in near-uniform stitched images. Both SSIM and SIQA also do not directly relate the performance of feature detectors to the quality of image stitching. Thus, using existing image quality assessment metrics to capture stitching artifacts in near-uniform scenes may not be fully effective.

Therefore, in addition to visual assessments for comparing stitching quality, this thesis proposes new evaluation metrics designed to capture the relationship between the inliers detected by feature detectors and the quality of stitched images. These metrics include the spread-overlap, S_o metric and the ratio of recall-to-spread-overlap, RC/S_o metric. Details of these metrics will be provided in Section 3.6.

2.6 Theoretical versus Deep Learning

In recent years, deep learning (DL) has become the dominant approach in computer vision (CV), including the field of image stitching (Fu et al., 2023; Huang et al., 2024; Xu et al., 2024; Yang et al., 2025). While many classic CV techniques developed over the past two decades may seem less relevant due to the rise of DL, it is important to understand that DL is not a universal solution for all CV challenges (O’Mahony et al., 2020). In fact, DL introduces its own set of challenges in image stitching and other CV applications (O’Mahony

et al., 2020; Yang et al., 2025). Although the latest DL methods can achieve significantly higher accuracy, they demand substantial computing power, including costly hardware for training and AI-accelerated platforms for development (O'Mahony et al., 2020).

Additionally, DL heavily relies on large datasets, often requiring millions of data records (O'Mahony et al., 2020). For more specific tasks, such as those beyond general image classification, the data requirements might be lower, typically ranging from hundreds to thousands of records (Yang et al., 2025). Training a deep convolutional neural network (CNN) can be time-consuming, taking hours or even days, depending on the availability of computing resources (O'Mahony et al., 2020). In situations where large datasets or advanced computing facilities are not available, theoretical methods remain valuable alternatives.

According to Xu et al. (2024) and Yang et al. (2025), DL has already been applied in various aspects of image stitching, including feature extraction and matching, homography estimation, and image reconstruction. It has also proven effective in specialized areas like fisheye camera image stitching and video stitching algorithms (O'Mahony et al., 2020). Since the introduction of deep learning-based SuperPoint by Detone et al. (2018), using representation learning for effective feature point extraction and matching, and SuperGlue by Sarlin et al. (2020), a neural network for matching local features, deep learning approaches for feature extraction have become a major research focus. Fu et al. (2023) suggest that future developments may focus on optimizing the joint processing of multiple images and using end-to-end neural networks for homography matrix regression, aiming to improve image registration and alignment.

Despite these advancements, DL is not yet fully established across all CV areas (Yang et al., 2025). Many challenging CV problems, such as low-resolution or texture-scarce

stitching, virtual reality, and 3D reconstruction, remain difficult for DL to address effectively on its own (O'Mahony et al., 2020; Yang et al., 2025). While DL has achieved significant advancements in CV, it is not a universal solution for every challenge. In some cases, theoretical algorithms are better suited, as discussed by O'Mahony et al. (2020). For instance, theoretical feature-based algorithms like SIFT, commonly used for image stitching, do not rely on specific classes and perform consistently across a wide range of image types. In contrast, DL-based features are often tailored to the training dataset and can lead to poor performance on images that differ from the training data (Huang et al., 2024). Although training with large datasets can mitigate this issue, it requires substantial research effort and is often impractical for specialized applications. Yang et al. (2025) highlight that only a limited number of standard datasets currently exist for deep-learning-based image and video stitching algorithms, with some being entirely synthetic. They emphasize the importance of developing representative real-world datasets to advance research in stitching techniques.

Moreover, a limited training dataset can hinder the CNN model's ability to generalize effectively. Tuning the parameters of a CNN model can be challenging due to the millions of interconnected variables, making it more complex than adjusting parameters in theoretical CV methods, which are generally easier to optimize across a broader range of images (O'Mahony et al., 2020). To overcome these challenges, Huang et al., (2024) and Yang et al., (2025) propose combining handcrafted features with neural network-based methods. These hybrid approaches integrate theoretical CV algorithms with DL, offer improved interpretability and robustness in image stitching. These methods have demonstrated significant advantages in solving difficult CV problems, including image stitching, where DL alone may fall short (Fu et al., 2023; Ma et al., 2021; O'Mahony et al., 2020).

2.7 Research Gap

Creating an effective image stitching algorithm is nearly impossible without precise image registration, a crucial requirement for accurate alignment. Although advanced blending techniques have been developed can help address registration errors, the importance of accurate registration in image stitching cannot be overstated (Nie et al., 2020; Zaragoza et al., 2014). Significant progress has been made in refining alignment and composition techniques to minimize or hide artifacts (Nie et al., 2020; Zaragoza et al., 2014). However, handling severe misalignment or distortion remains a considerable challenge (Xiang et al., 2018).

In any feature-based stitching method, a fundamental requirement is the presence of a sufficient quantity of stable pairs of interest points that are uniformly distributed across the overlapping areas of the images. Effective feature extraction is critical to meeting this requirement. It is essential that the extracted features are distinctive, evenly distributed throughout the image, easily detectable, resilient to various geometric and photometric changes, and have a high repeatability rate (Ghosh & Kaabouch, 2016).

However, detecting the reliable feature points in real-world scenes is often challenging due to factors such as wide baselines, significant parallax, changes in lighting and contrast, moving objects, low-texture areas, and occlusion (Lyu et al., 2019; Pandey & Pati, 2019). As a result, there is a growing need for researchers to focus more on image stitching in complex, real-world scenarios (Fu et al., 2023; Lyu et al., 2019). To the best of the author's knowledge, there has been little focus on addressing the specific challenges posed by near-uniform or low-texture scenes, despite their potential significance. Table 2.1 summarizes the datasets used in previous studies to evaluate feature detector performance.

The findings reveal that most state-of-the-art detectors are assessed primarily on feature-rich scenes, rather than on near-uniform or texture-sparse scenarios, underscoring a gap in current research.

Table 2.1: Summary of some state-of-the-art feature detectors and the datasets used for their performance evaluation

Detector	Feature Detection Method	Datasets Used
MSER (Matas et al., 2004)	Extracts covariant extremal regions under affine transformations (wide-baseline)	6 feature-rich image sets including <i>Bookshelf</i> , <i>Vallone</i> , <i>Wash</i> , <i>Kampa</i> , and <i>Cylindrical box</i> .
SIFT (Lowe, 2004)	Difference of Gaussian for detecting scale-invariant blobs	9 feature-rich images.
SURF (Bay et al., 2006)	Fast approximation of Hessian matrix for blob detection	8 standard datasets (each with 6 images) with geometric and photometric transformations (<i>Affine Covariant Features</i> , 2007).
BRISK (Leutenegger et al., 2011)	Builds on FAST and uses a customized circular sampling pattern to compute a binary (bit-string) descriptor for corner detection	7 standard datasets (each with 6 images) with geometric and photometric transformations (<i>Affine Covariant Features</i> , 2007).
KAZE (Alcantarilla et al., 2012)	Nonlinear diffusion-based scale-space for blob-like structures	5 standard datasets (each with 6 images) with geometric and photometric transformations (<i>Affine Covariant Features</i> , 2007); and 6 newly generated <i>Iguazu</i> images with added Gaussian noise.

Table 2.1 continued

AKAZE (Alcantarilla et al., 2013)	Accelerated KAZE using Fast Explicit Diffusion (FED) for nonlinear scale-space blob detection	8 standard datasets (each with 6 images) with geometric and photometric transformations (<i>Affine Covariant Features</i> , 2007); 1 real-world image for deformable surface matching; and 6 <i>Iguazu</i> images with Gaussian noise.
AGAST (Mair et al., 2010)	Adaptive and accelerated segment test corner detection using optimal binary decision trees	5 images from various scenes including <i>Lab</i> , <i>Outdoor</i> , <i>Indoor</i> , <i>Aerial</i> , and <i>Medical</i> .
ORB (Rublee et al., 2011)	Combination of FAST keypoint detector and rotated BRIEF descriptor (corner-based)	2 types of datasets: one with synthetic in-plane rotation and Gaussian noise; the other a real-world set of textured planar images from multiple viewpoints.
RIFT (Li et al., 2020)	Radiation-variation insensitive detection of corners and edges using phase congruency	6 types of multi-modal satellite/UAV datasets, each containing 10 image pairs.

This thesis specifically focuses on addressing the challenges associated with stitching images of near-uniform or low-texture scenes. The primary issues stem from two main factors: a lack of distinctive corresponding points and the "clustering" of feature points within a limited portion of the overlapping region, especially for near-uniform scene images. To overcome these challenges, this thesis proposes new image stitching methods based on scale-space theory. These methods involve creating and utilizing multiple nonlinear scale-

spaces to enable efficient non-maxima suppression, thereby enhancing the detection of the desired corresponding interest points.

The proposed methods, namely Lorentz-modulated nonlinear diffusion and Log-Gabor feature transform have demonstrated superior performance compared to current state-of-the-art feature detectors. Details of these methods and their performance are thoroughly discussed in subsequent chapters. Additionally, this thesis addresses the limitations of existing evaluation metrics by developing new metrics that more effectively capture the relationship between inliers and the quality of stitched images. The new evaluation metrics, namely spread-overlap, S_o and the ratio of recall-to-spread-overlap (RC/S_o), are used to evaluate the effectiveness of the newly developed feature-based image stitching methods.

2.8 Chapter Summary

Image stitching involves several key processing steps, including image registration, alignment, and blending (Fu et al., 2023). Among these, image registration is particularly crucial, as the success of stitching process largely depends on accurately matching the overlapping images. At this stage, feature detection plays a vital role by identifying distinctive interest points across image pairs (Awad & Hassaballah, 2016). This chapter reviews the evolution of stitching methods, with particular emphasis on multi-scale, feature-based techniques that are considered more effective in handling the challenges of near-uniform scenes.

In addition, this chapter highlights the classification of feature detection and description methods. It provides a concise overview of various existing algorithms for

identifying interest points, organized by their classification categories. It explains the fundamental notations and mathematical principles behind several widely used multi-scale feature detection and description methods, including SIFT (Lowe, 2004) and SUFT (Bay et al., 2006) serve as the foundation for the methods proposed in this thesis.

This chapter also explores various approaches for assessing the effectiveness of image stitching methods and discusses their limitations. It delves into the mathematical concepts underlying standard metrics commonly used for feature detectors and descriptors. Moreover, the chapter reviews the relevance of theoretical techniques in the modern era of deep learning in computer vision, highlighting promising hybrid approaches (Yang et al., 2025).

Finally, this chapter identifies critical research gaps in image stitching. First, existing feature detectors often struggle to extract reliable interest points in near-uniform scenes (Yang et al., 2025). Second, current evaluation metrics do not adequately capture the relationship between inliers and the overall stitching quality. These limitations highlight the need for more effective feature detection techniques and improved evaluation metrics, especially for near-uniform image stitching.

CHAPTER 3

METHODOLOGY

3.1 Introduction

While the fundamentals of image stitching have been extensively studied, accurately matching and aligning low-texture or near-uniform images remains a challenging and unresolved issue (Fu et al., 2023; Lyu et al., 2019; Yang et al., 2025). Image stitching algorithms typically begin with feature detection, where keypoints are identified across multiple overlapping images (Awad & Hassaballah, 2016; Huang et al., 2024). Indeed, feature detector plays a critical role in accurately identifying reliable inliers, which directly influence the quality of the final stitched image, particularly in reducing visual artifacts such as seams, ghosting, and misalignment.

To address the limitations of existing feature detection techniques in handling near-uniform or low-texture scenes, this chapter introduces two novel methods aimed at increasing the quantity and distribution of reliable inliers across overlapping regions. The first method, known as the enhanced Partial Differential Equation (ePDE), applies a modified conductivity function derived from the Lorentz factor that is inspired by Einstein’s theory of special relativity (Bambi, 2018). This modification enables the construction of multi-scale, enhanced edge-preserving nonlinear scale-spaces that improve the robustness of interest point detection in near-uniform region.

The second method, known as the Log-Gabor Feature Transform (LGFT), utilizes Log-Gabor filters to overcome the limitations of intensity-based feature detection techniques (Li et al., 2020). This method generates multiple oriented and scaled Log-Gabor scale-spaces

that enhance texture granularity, providing richer structural information for improved detection of interest point. These advancements support more precise estimation of the global homography transformation model, resulting in better alignment and reduced geometric distortions in the stitched images.

In addition to the proposed detection methods, this chapter introduces two new evaluation metrics, the spread-overlap, S_o and RC/S_o score, that address a critical gap in current image stitching evaluation practices. Unlike existing metrics that often overlook the spatial distribution of inliers, the proposed metrics are specifically designed to capture both the quantity and distribution of detected inliers in relation to the visual quality of the stitched images (Mikolajczyk et al., 2005; Mikolajczyk & Schmid, 2001; Wang et al., 2004).

This chapter is structured into seven sections. The core contributions of this thesis are detailed in Sections 3.2 and 3.3, describing the proposed feature detection methods. Section 3.4 discusses the feature description technique employed, followed by the implementation of the image stitching algorithm in Section 3.5. Section 3.6 introduces the proposed evaluation metrics, and Section 3.7 concludes with a summary of the chapter.

3.2 Proposed Method 1: Enhanced Partial Differential Equation (ePDE)

Scale-space theory is crucial for feature detection, as it uses Partial Differential Equations (PDEs) to generate image representations at multiple scales. PDEs decompose an image into progressively smoother versions across different scales or time units, known as scale-space. These scale-space representations are capable of identifying potential interest points and can be applied in diverse areas, including image registration, object recognition,

and image stitching. Several scale-space filtering approaches have been developed and widely used over the years, including linear methods (Lindeberg, 1994) and nonlinear methods such as isotropic models (Rudin et al., 1992) and anisotropic diffusion models (Weickert, 1996). As these methods aim to smooth images across various scales, nonlinear diffusion models are especially effective at preserving edges, outperforming the linear Gaussian filter in this regard.

The earliest formulation of linear scale-space is axiomatically established by Taizo Iijima (Weickert et al., 1999). However, the theories of linear scale-space introduced by Witkin (1984) and Koenderink (1984) are more commonly referenced in the research community. Witkin (1984) presented the Gaussian scale-space representation by applying a Gaussian kernel to the original image. With reasonable assumptions, Koenderink (1984) and Lindeberg (1994) demonstrated that the Gaussian function and its derivatives are the only logical linear scale-space kernels. In contrast to Gaussian, Duits et al. (2004) explored the Poisson scale-space as a viable alternative. Recent findings suggest that multi-scale Poisson kernel yields considerable stable features in scale-space (Bhatia et al., 2021). Generally, the Gaussian kernel is defined as follows:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad \text{Equation 3.1}$$

where σ represents the scale level, and the Cartesian coordinates within the image plane are indicated as x and y . Basically, the Gaussian scale-space of an image $L(x, y, \sigma)$ is fundamentally generated by convolving a variable-scale Gaussian kernel $G(x, y, \sigma)$ with an input image $I(x, y)$ as follows:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad \text{Equation 3.2}$$

where $*$ signifies the operation of convolution between x and y . Equation 3.2 typically yields simpler or smoother image representations using a Gaussian kernel at a larger scale level.

3.2.1 Nonlinear Diffusion Filtering

Although Gaussian smoothing is effective in reducing noise and emphasizing prominent structures at larger scales, its primary limitation is that it fails to preserve object boundaries, leading to an increased loss of localized structural details as the scale increases. This constraint can be mitigated through the nonlinear diffusion method introduced by Perona & Malik (1990) for image restoration and edge detection. Nonlinear diffusion is expressed as a Partial Differential Equation (PDE) that governs prior information about image features via the diffusion coefficient during the filtering processing. Nonlinear scale-space remains relatively stable in noisy conditions while preserving details or edges effectively. For a given digital image, I , nonlinear diffusion can be mathematically expressed as follows:

$$\frac{\partial I}{\partial t} = \text{div}(g(x, y, t) \cdot \nabla I) \quad \text{Equation 3.3}$$

where $g(x, y, t)$ is the conductivity function that determines the diffusion weight, div is the divergence operator, and ∇ is the spatial gradient operator. In the function $g(x, y, t)$, the variable t act as the ‘time’ scale parameter, which is used to enumerate iteration ‘time’ steps that guide the preceding image into simpler image representations in a discrete manner. Therefore, the function $g(x, y, t)$ regulates the diffusion process, adapting it to the local differential structure of each pixel in the image.

In general, there are three distinct formulations of conductivity functions. The following conductivity functions are proposed in the work of Perona & Malik (1990).

$$g_1 = \exp\left(-\frac{|\nabla I(x, y, t)|^2}{k^2}\right) \quad \text{Equation 3.4}$$

$$g_2 = \frac{1}{1 + \frac{|\nabla I(x, y, t)|^2}{k^2}} \quad \text{Equation 3.5}$$

The parameter k serves as the contrast factor, influencing the magnitude of diffusion weights relative to the image's spatial gradient, and in turn affect the sharpness of boundaries. According to Alcantarilla et al. (2012) and Perona & Malik (1990), the value of k can either be set manually as a constant or determined automatically from the image gradient's histogram. In contrast, Weickert introduced an alternative version of the conductivity function g_3 , which is designed to decrease diffusivity rapidly in areas where smoothing is more pronounced on either side of an edge compared to across it (Weickert, 2001; Weickert et al., 1998).

$$g_3 = \begin{cases} 1 & |\nabla I|^2 = 0 \\ 1 - \exp\left(-\frac{3.315}{(|\nabla I(x, y, t)|/k)^8}\right) & |\nabla I|^2 > 0 \end{cases} \quad \text{Equation 3.6}$$

These three forms of conductivity functions generate nonlinear scale-spaces with notable differences: g_1 exposes high-contrast edges, g_2 prioritizes larger areas rather than smaller ones, and g_3 focuses on intraregional smoothing rather than interregional blurring. Alcantarilla et al. (2012) suggest that g_1 and g_3 are better suited for corner detection, while g_2 excels at identifying blob-like features.

3.2.2 Lorentz Factor

Einstein's special theory of relativity defines time dilation phenomenon as the difference in elapsed time between two events, observed by two clocks that are either moving relative to one another or positioned at varying gravitational potentials (Bambi, 2018). Time dilation can generally be mathematically represented as follows:

$$\Delta t = \gamma \Delta t' \quad \text{Equation 3.7}$$

where Δt denotes the time interval measured by the moving clock, while $\Delta t'$ indicates the time interval measured by the stationary clock. The Lorentz factor, γ is a scaling factor that measures the extent to which time is stretched or contracted relative to an observer. It is derived from the Lorentz transformations (Bambi, 2018) and is defined as

$$\gamma = \frac{1}{\sqrt{1 - \frac{v^2}{c^2}}} \quad \text{Equation 3.8}$$

where v represents the velocity of the moving object and c stands for the speed of light. Since γ is greater than 1, the elapsed time Δt measured by the moving clock is greater than the elapsed time $\Delta t'$ recorded by the stationary clock. This effect is referred to as time dilation. In simple terms, as an object moves faster through space, the slower it experiences the passage of time.

3.2.3 Improved Conductivity Function

Inspired by the concept of time dilation, which is explicitly quantified by the Lorentz function, the formulation in Equation 3.8 can be adapted and generalized to derive a newly optimized conductivity function. This improved function regulates diffusion weighting in

the diffusion equation and is used to generate novel and distinct scale-spaces. These scale-spaces help enhance the detection of interest points, particularly in near-uniform image regions.

Given an input image, let the spatial gradient at each pixel be represented as $\nabla I(x, y, t)$. As stated in Equation 3.3, achieving simpler scale-space representation of the image involves applying a diffusion weight, g to preserve object boundaries (regions with high image gradients) while smoothing homogenous regions (regions with low image gradients). Essentially, regions with higher spatial gradients, such as edges, tend to diffuse more slowly over time scale (Alcantarilla et al., 2012). This behaviour is analogous to the characteristics of the Lorentz function (Bambi, 2018). Consequently, by substituting the variable v and constant c in Equation 3.8 with the spatial gradient $|\nabla I(x, y, t)|$ and a contrast parameter k , respectively, a revised conductivity function can be formulated as

$$g_4 = \frac{1}{\sqrt{\left|1 - \frac{|\nabla I(x, y, t)|^2}{k^2}\right|}} = \frac{1}{\sqrt{\alpha}} \quad \text{Equation 3.9}$$

Considering that the magnitude of an image's spatial gradient $|\nabla I(x, y, t)|$ is typically scaled to a range of 1 to 255 in digital images (excluding flat regions with zero gradient), the absolute value of α is used to prevent the introduction of complex numbers in the square root term of g_4 . Based upon the experiments conducted in this thesis, a parameter k is empirically chosen within the range of 0.1 to 0.9, as this interval typically yields stable diffusivity outputs. Since g_4 focuses on larger areas and is more effective at detecting blob-like features, Figure 3.1 presents a comparison specifically with the standard g_2 , which generates a comparable nonlinear scale-space.

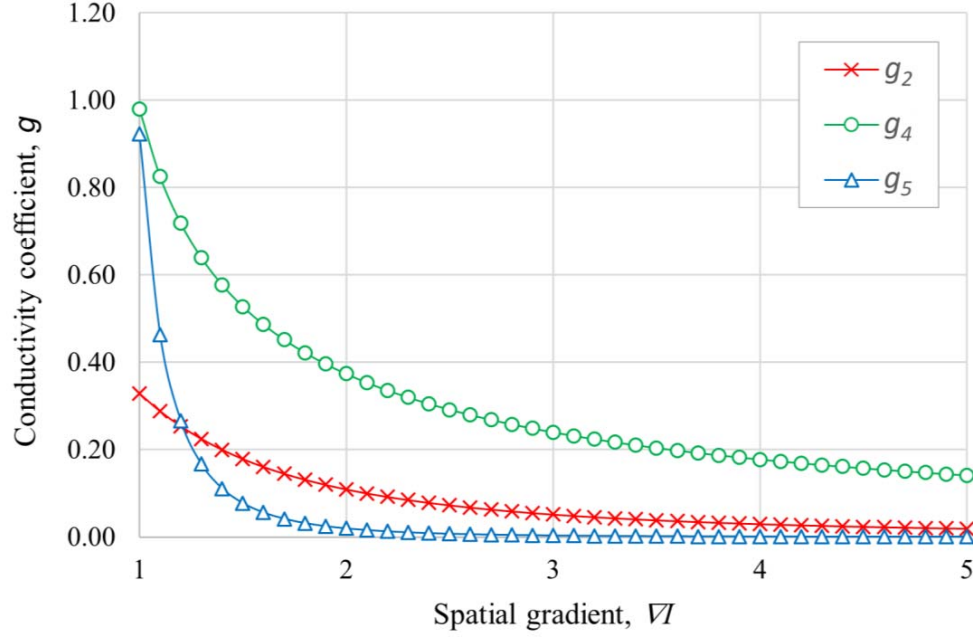


Figure 3.1: The relationship between conductivity coefficients g_2 , g_4 and g_5 and image spatial gradients $|\nabla I|$ with a fixed parameter $k = 0.7$

Figure 3.1 illustrates the influence of various conductivity coefficients on image spatial gradients $|\nabla I(x, y, t)|$ by keeping the parameter k constant. The graph is truncated at $|\nabla I(x, y, t)| = 5$, beyond which the conductivity values approach zero for larger spatial gradients. As depicted in the figure, it is evident that the conductivity coefficient g_4 has a more pronounced effect on small image gradients (i.e., $|\nabla I(x, y, t)| = 1$ or homogenous regions) compared to the standard g_2 . However, when $|\nabla I(x, y, t)| > 1$, the g_4 coefficient is significantly higher, potentially leading to a blurred scale-space at larger scales. To reduce this blurring effect, the g_4 function is adjusted by raising it to the fourth power, resulting in a new function, g_5 as

$$g_5 = \frac{1}{\left(1 - \frac{|\nabla I(x, y, t)|^2}{k^2}\right)^2} \quad \text{Equation 3.10}$$

As illustrated in Figure 3.1, the modified g_5 as well exhibits a stronger coefficient for $|\nabla I(x, y, t)| = 1$. The strong coefficient is anticipated to rapidly smooth out homogenous regions. As well, the g_5 coefficient approaches zero when $|\nabla I(x, y, t)| > 1$, indicating that high image gradients along object boundaries tend to degrade at a slower rate over time, thereby effectively preserving object boundaries at larger scales. The enhancement of the g_4 and g_5 functions generate distinct scale-spaces that significantly improve the detection of interest points in near-uniform regions and enhance image stitching quality, as discussed further in Chapter 4.

3.2.4 Lorentz Modulated Nonlinear Scale-spaces

Concerning digital images, Weickert's modified semi-implicit method, known as the Additive Operator Splitting (AOS) scheme, is employed to create nonlinear diffusion scale-spaces (Weickert, 2001; Weickert et al., 1998). The AOS method provides a numerical approximation of the nonlinear partial differential equation (PDE) in a discretized form. Within the framework of the AOS scheme, Equation 3.3 can be expressed in vector-matrix notation as follows (Weickert, 2001; Weickert et al., 1998):

$$L^{t+1} = \frac{1}{m} \sum_{l=1}^m (Id - m\tau A_l(L^t))^{-1} L^t \quad \text{Equation 3.11}$$

where m is the number of dimensions (with $m = 2$), τ denotes the time step size, $A_l(L^t)$ is the discretized second-order derivative of L^t along the l -th coordinate axis, and L^t represents

the nonlinear scale-spaces at evolution time t . The identity matrix is denoted as Id . As pixels are indexed sequentially along the direction l , the term $(Id - m\tau A_l(L^t))$ represents one-dimensional diffusive interactions along the corresponding axes, forming diagonally dominant tridiagonal matrices within a linear system of equations. When applied to L^t , the term $(Id - m\tau A_l(L^t))$ models anisotropic or directional diffusion along the l -th axis at evolution time t . Equation 3.11 can be efficiently solved using the Thomas algorithm or the Tri-Diagonal Matrix Algorithm (TDMA) (Ralli, 2011; Weickert et al., 1998). In the AOS approach, all coordinate axes are treated uniformly for each step size τ , allowing the construction of discrete nonlinear scale-spaces.

Before constructing the scale-spaces, the first step is to compute a series of evolution times t_i , from which the step size τ is determined as $t_{i+1} - t_i$ for used in Equation 3.11. The scale-spaces are organized into sequential discrete octaves, o and sub-levels, s . These are subsequently analysed by scaling up SURF's box filter to approximate second-order Gaussian derivatives (Bay et al., 2006; Evans, 2009). Each pair of octave-sublevel is associated with a corresponding box filter size defined as:

$$f_i = 3((2^o \times s) + 1), \quad i = \{0 \dots N\} \quad \text{Equation 3.12}$$

where the initial filter size $f_o = 9$ corresponds to a 9×9 pixels box filter, approximating the Gaussian derivatives with an initial standard deviation $\sigma_o = 1.6$ in the proposed method. As the filter size grows, the corresponding Gaussian scale likewise increases that can be easily computed because the ratio of the filter layout stays constant. Since nonlinear diffusion operates in time units, the discrete scale sigma σ_i can be associated with their corresponding time units t_i using

$$t_i = \frac{1}{2} \sigma_i^2, \quad i = \{0 \dots N\} \quad \text{Equation 3.13}$$

where N represents the total number of two-dimensional arrays of scale-spaces. In the proposed method, a total of 12 scale-spaces is created, organized into 5 octaves, each containing 4 sublevels. The first octave consists of 4 consecutive scale-spaces, whereas the subsequent octaves include the last 2 scale-spaces from the preceding octave, together with 2 newly generated successive scale-spaces.

3.2.5 ePDE Feature Detection

The generated scale-space images are essential for identifying scale-invariant interest points. For feature detection, the Hessian matrix is employed by transforming all nonlinear scale-spaces into integral images, allowing for rapid calculations using box filters (Bay et al., 2006; Evans, 2009). For each integral image, the scale-normalized Hessian determinant matrix is calculated using SURF's box filters to approximate the Laplacian of Gaussian (Bay et al., 2006). This determinant serves as an indicator of blob responses, subsequently processed using non-maxima suppression. Hessian determinant responses that fall below a predetermined threshold are eliminated, while those above the threshold are retained to control detection efficacy.

After applying the thresholding, local maxima are identified from the response maps (i.e., the blob responses from the scale-spaces). This is achieved using non-maxima suppression approach, where each candidate point is compared to its 26 neighbouring pixels within a 3×3 pixel region across the current and adjacent scale-spaces as shown in

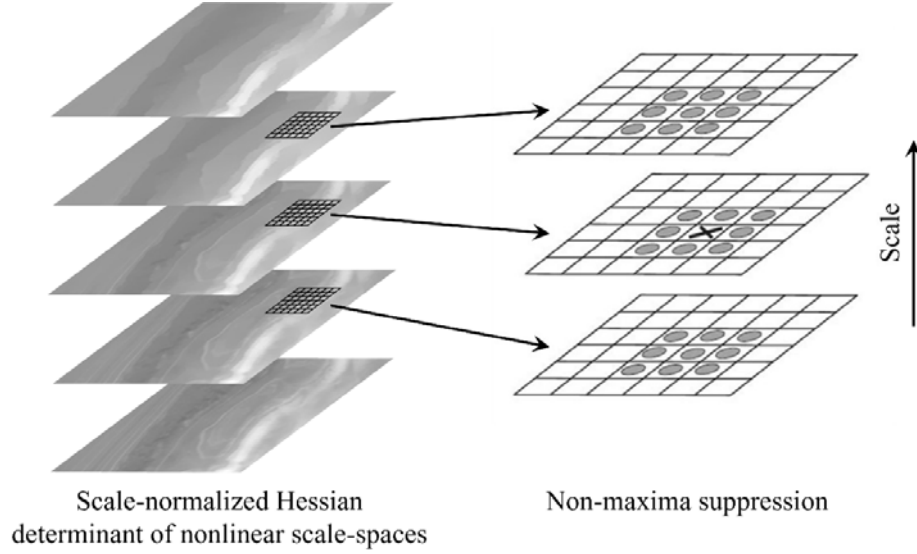


Figure 3.2: Non-maxima suppression. The pixel marked ‘×’ is selected as interest point if it is greater than neighboring pixels in its interval and adjacent intervals (Awad & Hassaballah, 2016)

Figure 3.2 (Lowe, 2004). A point is considered an interest point if it exceeds all its neighbours.

Finally, the positions of the interest points is interpolated by fitting a three-dimensional quadratic function to their neighbouring data points, achieving sub-pixel accuracy and stable localization (Lowe, 2004). This process eliminates unstable candidates that have low contrast (i.e., those sensitive to noise) and poor edge localization. Interpolated maxima locations significantly enhance matching accuracy and stability. For image matching purposes, each identified interest point must have a distinctive feature descriptor, as will be further explained in subsequent Section 3.4. Table 3.1 summarizes the proposed Lorentz-modulated nonlinear scale-spaces construction and the ePDE feature detection algorithm.

Table 3.1: Summary of the proposed enhanced Partial Differential Equation (ePDE) feature detector algorithm

Algorithm I: enhanced Partial Differential Equation (ePDE) feature detector
Input: Images I_n, Gaussian sigma $\sigma = 0.5$, initial sigma $\sigma_o = 1.6$, number of octaves n_o, number of sub-levels n_s and contrast factor k Output: A set of detected interest points $\tilde{P}_n$ function ePDE for $o = 1 \cdots n_o$ for $s = 1 \cdots n_s$ i. Calculate a set of filter size f_i ii. Calculate corresponding discrete scale sigma σ_i iii. Calculate evolution times t_i and step size $\tau_i = t_{i+1} - t_i$ end for end for for $n = 1, 2$ i. Denoise I_n with Gaussian kernel $\sigma \rightarrow L_n$ ii. Compute N nonlinear scale-spaces L_i using the loop outlined below: for $i = 1 \cdots N$ a. Compute diffusion weight g_4 or g_5 b. Compute diffusivity matrix $A_l(L_n)$ c. Solve the vector-matrix using AOS and TDMA algorithms d. Transform nonlinear scale-spaces to integral images L_i end for iii. Compute Hessian determinant and Laplacian sign response map from each L_i iv. Apply non-maxima suppression for interest points search

Table 3.1 continued

v. Interpolate extrema locations to reject unstable candidate points
end for

3.3 Proposed Method 2: Log-Gabor Feature Transform (LGFT)

This method introduces an alternative feature detection technique based on the Log-Gabor feature transform, specifically designed to overcome the limitations of intensity-based feature detection methods. Gabor filter is widely recognized as an ideal function for signal processing because of its important mathematical properties (Fischer et al., 2007). Essentially, a Gabor function combines a Gaussian function with a complex exponential, with the Gaussian function shifted away from the origin in the Fourier domain. Similarly, the 2D Gabor filter effectively represents image information, achieving optimal localization in both frequency and spatial domains. Studies on visual cortical neurons have shown that the two-dimensional receptive field profiles of simple cells in the Primary Visual Cortex (V1) of mammal are effectively characterized by images filtered with a 2D Gabor function (Daugman, 1985; Field, 1987). However, Gabor filters present several drawbacks (Arrospide & Salgado, 2013; Field, 1987; Fischer et al., 2007):

- i. Gabor functions act as bandpass filters, exhibiting strong responses at lower frequencies. However, the combination of this response with a $1/f$ fall-off in image amplitude leads to redundant filter information. Additionally, these filters fail to capture the high-frequency tail of the image.

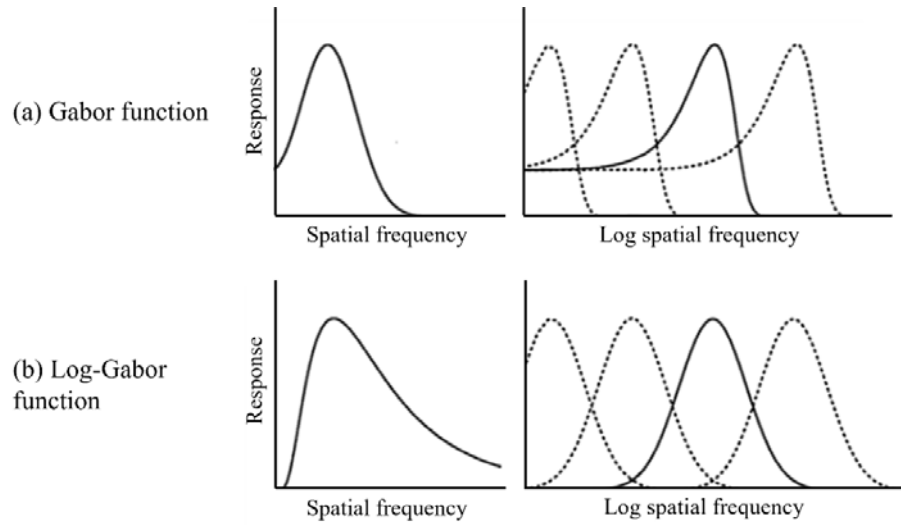


Figure 3.3: Comparison between (a) Gabor and (b) Log-Gabor functions (Field, 1987)

- ii. Gabor filters possess a bandwidth of about one octave and are not optimal for wide spectral coverage. As a result, a substantial number of filters is needed to encompass extensive spectral information.
- iii. Gabor filters exhibit a lack of uniformity in the mid-frequency range.

To overcome above issues, the Log-Gabor function is employed. The function is regarded as a better option for capturing information from natural images compared to the Gabor function (Field, 1987).

3.3.1 2D Log-Gabor Filtering and Scale-spaces

The Log-Gabor function is fundamentally crafted as a Gaussian response that spreads from the origin along the logarithmic frequency axis. Figure 3.3 illustrates a comparison between the Gabor and Log-Gabor functions. The key advantages of the Log-Gabor filter include:

- i. Log-Gabor filters exhibit symmetrical responses in the Fourier domain, allowing the same filters to be used for both direct and inverse transforms in image analysis and synthesis.
- ii. Log-Gabor filters reduce redundancy in the lower frequency range by minimizing overlap between filters and providing more even spectral coverage.
- iii. Log-Gabor filters feature extended tails at higher linear frequencies, improving the frequency fall-off observed in natural images.

Mathematically, the Log-Gabor function consists of two main components: the radial component that controls the frequency band, and the angular component that defines the filter's orientation. Initially proposed by Field (1987), the radial component of Log-Gabor function in the frequency domain f is expressed as follows:

$$G_s(f) = \exp\left(-\frac{(\log(f/f_s))^2}{2(\log(\sigma_{f_s}/f_s))^2}\right) \quad \text{Equation 3.14}$$

where $f_s = 1/\lambda$ represents the central radial frequency of the filter with λ represents the wavelength of the Log-Gabor function, and σ_{f_s} denotes the radial bandwidth of B_f in octaves, as defined by (Gao et al., 2007)

$$B_f = 2 \sqrt{\frac{2}{\log(2)}} \left| \log\left(\frac{\sigma_{f_s}}{f_s}\right) \right| \quad \text{Equation 3.15}$$

The angular component of Log-Gabor function in the orientation angle θ is expressed as follows (Arrospide & Salgado, 2013; Li et al., 2020; Matas et al., 2004; Nava et al., 2012; Gao et al., 2007):

$$G_o(\theta) = \exp\left(-\frac{(\theta - \theta_o)^2}{2\sigma_{\theta_o}^2}\right) \quad \text{Equation 3.16}$$

where θ_o represents the orientation angle of the filter, and σ_{θ_o} denotes the angular bandwidth of B_θ in octaves, defined as (Gao et al., 2007)

$$B_\theta = 2\sigma_\theta\sqrt{2\log(2)} \quad \text{Equation 3.17}$$

To construct 2D Log-Gabor filter, the radial component $G_s(f)$ is multiplied by the angular component $G_o(\theta)$ around a central frequency (f_s, θ_o) as follows:

$$G_{s,o}(f, \theta) = G_s(f) \cdot G_o(\theta) \quad \text{Equation 3.18}$$

$$G_{s,o}(f, \theta) = \begin{cases} \exp\left(-\frac{(\log(f/f_s))^2}{2(\log(\sigma_{f_s}/f_s))^2}\right) \exp\left(-\frac{(\theta - \theta_o)^2}{2\sigma_{\theta_o}^2}\right), & f \neq 0 \\ 0, & f = 0 \end{cases} \quad \text{Equation 3.19}$$

where $s \in \{1, \dots, n_s\}$ and $o \in \{1, \dots, n_o\}$ represent the 2D Log-Gabor filter's scale and orientation, respectively. In this context, n_s and n_o denote the number of scales and orientations of the filters, respectively. On the other hand, Fischer et al. (2007) proposed a different expression for the 2D Log-Gabor filter in the log-polar coordinates of the Fourier domain, which is given as

$$G_{s,o}(\rho, \theta) = \exp\left(-\frac{(\rho - \rho_s)^2}{2\sigma_\rho^2}\right) \cdot G_o(\theta) \quad \text{Equation 3.20}$$

where (ρ, θ) denotes the log-polar coordinates. Here, $\rho = \log f$, $\rho_s = \log f_s$ and $\sigma_\rho = \log(\sigma_{f_s}/f_s)$ for $f \neq 0$.

Since the Log-Gabor function operates in the frequency domain, the input image $I(x, y)$ must first be transformed to this domain using a 2D Fourier transform, resulting in the Fourier-transformed image $\mathcal{F}(I(x, y))$. The representation of 2D Log-Gabor feature in the frequency domain is subsequently achieved by applying a convolution between $\mathcal{F}(I(x, y))$ and a 2D Log-Gabor filter (Fischer et al., 2007). This convolution produces complex-valued results that can be expressed as a response vector, as follows:

$$[E_{s,o}^* + iO_{s,o}^*] = \mathcal{F}(I(x, y)) * G_{s,o}(f, \theta) \quad \text{Equation 3.21}$$

where i denotes the imaginary unit, while $E_{s,o}^*$ and $O_{s,o}^*$ represents the even-symmetric and the odd-symmetric components of the convolution responses, respectively, in the frequency domain at scale s and orientation o . The corresponding responses in the spatial domain can be derived by applying the inverse Fourier transform to the convolution responses (Fischer et al., 2007; Li et al., 2020; Ye et al., 2017), as follows:

$$[E_{s,o} + iO_{s,o}] = \mathcal{F}^{-1}[E_{s,o}^* + iO_{s,o}^*] \quad \text{Equation 3.22}$$

where $E_{s,o}$ denotes the even-symmetric component, and $O_{s,o}$ signifies the odd-symmetric components in the spatial domain at scale s and orientation o . Finally, the amplitude component $A_{s,o}$ is computed using

$$A_{s,o} = \sqrt{E_{s,o}^2 + O_{s,o}^2} \quad \text{Equation 3.23}$$

The proposed method defines $n_s = 4$ and $n_o \in \{1, 2, 3, 4, 5, 6\}$ to create a series of multi-scale and multi-orientation Log-Gabor convolution responses, known as Log-Gabor scale-spaces throughout this paper. For instance, with $n_s = 4$ and $n_o = 3$, a total of 12 Log-Gabor scale-spaces are generated, comprising of 4 scale levels for each orientation.

These scale-spaces are employed to effectively identifying interest points by examining maxima across all scales.

3.3.2 LGFT Feature Detection

The generated Log-Gabor scale-spaces are then arranged into a sequence of discrete octaves and sub-levels. Each is indexed by o (octave) and s (sub-level), representing the orientation and scale properties of the 2D Log-Gabor filter, respectively. In each Log-Gabor scale-space, the Laplacian of Gaussian (LoG) is approximated using a box filter that represents the Gaussian derivative kernel, applied across the x , y and xy directions (Bay et al., 2008; Evans, 2009). In this proposed method, the size of the box filter for each sublevel scale-space is determined based on the corresponding wavelength assigned to each Log-Gabor scale-space. As a result, each sublevel scale-space is associated with a specific box-type convolution filter size $\emptyset$, defined as follows:

$$\emptyset = \lceil w\lambda \rceil + n = \lceil w\lambda_{min}C^{s-1} \rceil + n \quad \text{Equation 3.24}$$

where w serves as a weight parameter that regulates the size of the box filter, λ_{min} represents the smallest wavelength scale of the 2D Log-Gabor filter, and C denotes the scaling factor between consecutive 2D Log-Gabor filters. The notation $\lceil \dots \rceil$ signifies the nearest integer, and n is defined as follows: $n = 0$ when $\lceil w\lambda \rceil$ returns an odd value, and $n = 1$ when $\lceil w\lambda \rceil$ returns an even value.

Similar to the ePDE feature detection, the process of identifying scale-invariant blob-like interest points from the Log-Gabor scale-spaces begins by converting these scale-spaces into a series of integral scale-spaces. This is followed by determining the blob response maps

(Bay et al., 2008; Evans, 2009), they are then further analysed through non-maxima suppression (Lowe, 2004). Table 3.2 summarizes the proposed Log-Gabor Feature Transform (LGFT) algorithm for interest points detection across different layers of Log-Gabor scale-spaces.

Table 3.2: Summary of the proposed Log-Gabor Feature Transform (LGFT) feature detector algorithm

Algorithm II: Log-Gabor Feature Transform (LGFT) feature detector
Input: Images I_n, Gaussian sigma $\sigma = 0.5$, initial sigma $\sigma_o = 1.6$, number of scales n_s, number of orientations n_o, radial bandwidth σ_{f_s}, smallest wavelength filter $\lambda_{min} = 3$, scaling factor C and parameter w Output: A set of detected interest points $\tilde{P}_n$ function LGFT for $s = 1 \cdots n_s$ i. Calculate <i>LoG</i> box filter sizes $\emptyset$ ii. Calculate corresponding discrete scale sigma σ_i iii. Compute radial component of the Log-Gabor function $G_s(f)$ end for for $o = 1 \cdots n_o$ Compute angular component of the Log-Gabor function $G_o(\theta)$ for $s = 1 \cdots n_s$ Multiplying $G_s(f)$ and $G_o(\theta)$ to construct 2D Log-Gabor filters end for end for for $n = 1, 2$

Table 3.2 continued

i.	Denoise I_n with Gaussian kernel $\sigma \rightarrow L_n$
ii.	2D Fourier transform I_n to yield $\mathcal{F}(I_n)$
iii.	Convolve $\mathcal{F}(I_n)$ with $G_s(f) \cdot G_o(\theta)$, return $n_s \times n_o$ layers of $[E_{s,o}^* + iO_{s,o}^*]$
iv.	Inverse Fourier transform each layer of $[E_{s,o}^* + iO_{s,o}^*]$, return $[E_{s,o} + iO_{s,o}]$
v.	Compute absolute value of each layer of $[E_{s,o} + iO_{s,o}]$, return $A_{s,o}$
vi.	Compute Hessian determinant and Laplacian sign response map from each L_i
vii.	Apply non-maxima suppression for interest points search
viii.	Interpolate extrema locations to reject unstable candidate points
end for	

3.4 Feature Description

For effective image matching, it is essential to define a distinctive feature description for each identified interest point from the ePDE and LGFT methods. This feature description is derived from a square region of the original input image, aligned with the primary orientation at each key interest point. The process primarily involves two crucial steps: determining the dominant orientation and defining the descriptor vectors.

To determine the dominant orientation, Haar wavelet responses with a window size of $4\alpha_i$ are first computed for a set pixels within a circular region of radius $6\alpha_i$ centred on the detected interest point (Bay et al., 2006). These responses are then weighted using a Gaussian function with a standard deviation of $2.5\alpha_i$ (Evans, 2009). In this context, α_i refers to either the discrete scale σ_i that corresponds to the time units t_i in the ePDE method, or the scale-dependent sampling step size s_i associated with the interest point in the LGFT method. The horizontal and vertical Haar wavelet responses that approximate intensity gradients in the x and y directions are treated as vectors, and summed within a rotating circular segment of

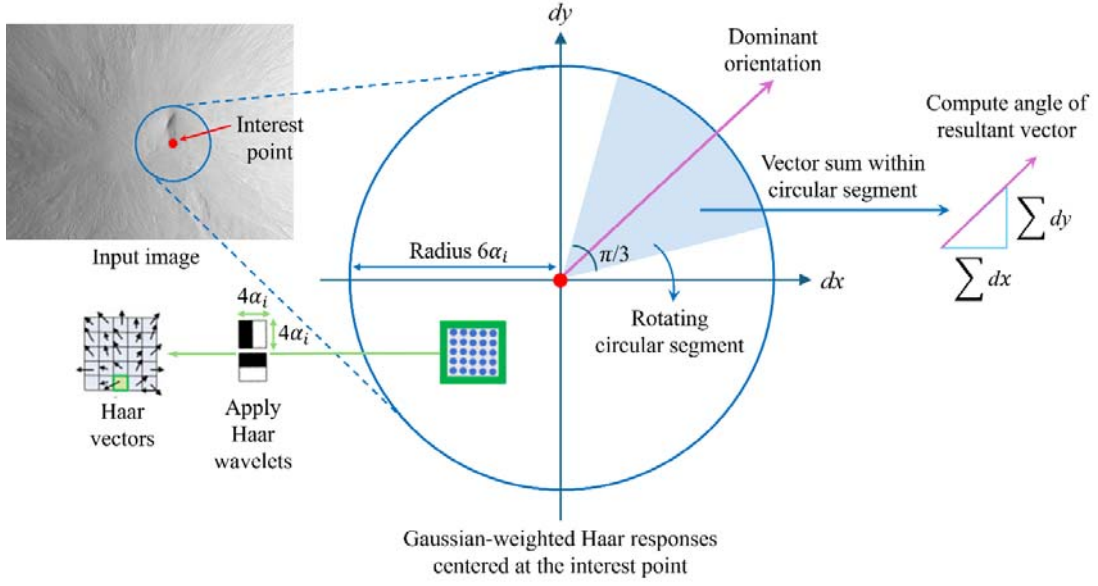


Figure 3.4: Orientation assignment process. Haar wavelet responses in the x and y directions are summed within a rotating circular segment. The segment with the largest resultant vector indicates the dominant orientation of the interest point

angle $\pi/3$ centred on the interest point. The vector sum within each segment yields a local orientation vector, and the angle of this vector (in radians or degrees) is computed from the summed x and y responses. The segment with the longest resultant vector determines the dominant orientation of the interest point. This process is illustrated in Figure 3.4.

To construct the descriptor vector for each interest point in the image, the M-SURF descriptor (Alcantarilla et al., 2012) is applied by calculating Haar-wavelet responses of size α_i in the horizontal and vertical directions relative to the dominant orientation of the detected interest point, denoted as d_x and d_y . The responses are computed over a square region grid of size $24\alpha_i \times 24\alpha_i$, which is split into smaller 4×4 square subregions of size $9\alpha_i \times 9\alpha_i$ with an overlapping zone of $2\alpha_i$ width. In each subregion, wavelet responses are weighted using a subregion-centred Gaussian (standard deviation = $2.5\alpha_i$) and aggregated into a four-

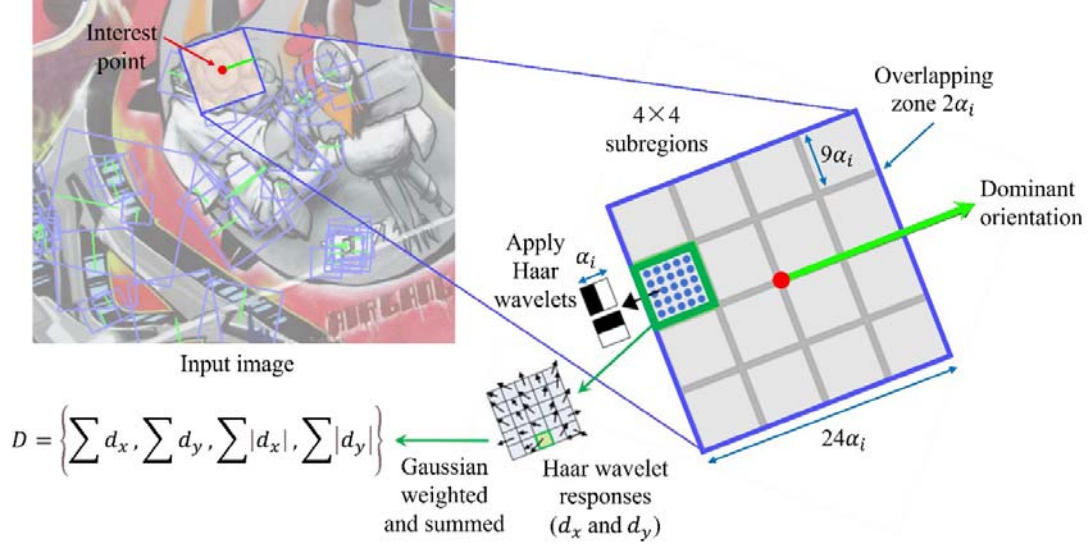


Figure 3.5: Descriptor construction process. The blue window (sized at $24\alpha_i$) is centered on the interest point (red dot) and aligned with the dominant orientation (green arrow). The green square is one of 16 subregions; blue dots indicate pixels used to compute Haar wavelet responses for constructing the $4 \times 4 \times 4$ -element descriptor

dimensional descriptor vector, denoted as $D = \{\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|\}$. This results in an overall length of 64 ($= 4 \times 4 \times 4$) elements feature vector for each interest point. Each subregion vector is then weighted using a Gaussian (standard deviation $= 1.5\alpha_i$) over a 4×4 square mask centred at the interest point. Finally, each descriptor vector is normalized to a unit vector to achieve contrast invariance. This process is illustrated in Figure 3.5.

3.5 Image Stitching Algorithm

Accurate matching of a significant number of interest points between overlapping images is essential for effective image stitching. This study utilizes a matching algorithm that employs the approximate nearest neighbour search from the VLFeat open-source library (Vedaldi & Fulkerson, 2010) to gather a set of indexed corresponding interest points along

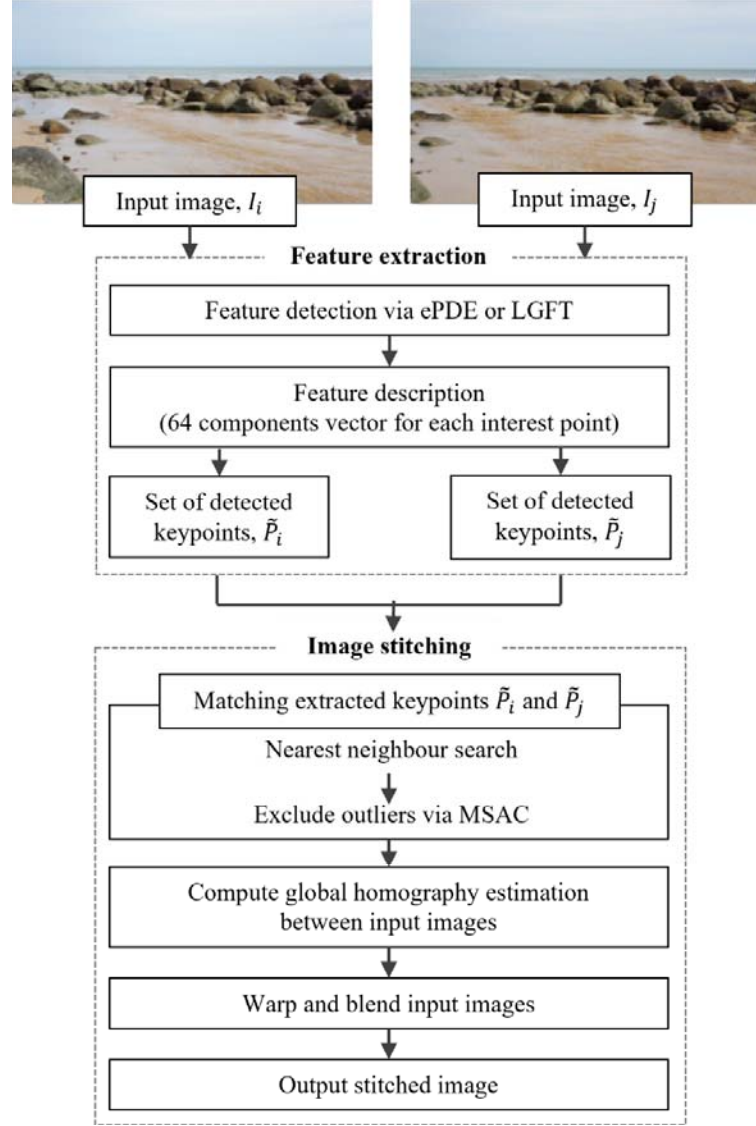


Figure 3.6: Flowchart outlining the entire process of image stitching

with their squared Euclidean distances. To better estimate inliers and effectively reject outliers, this study employs a variant of the Random Sample Consensus (RANSAC) algorithm called the M-estimator Sample Consensus (MSAC) algorithm (Torr & Zisserman, 2000). This approach enhances the matching of correct matches, or inliers, between the images. Given the inherent randomness of the MSAC algorithm, the number of inliers may

fluctuate slightly with each execution, although such variations are typically negligible. To maximize the number of inliers, the MSAC algorithm is executed iteratively over 20 trials. This number is empirically determined to provide an optimal balance between performance consistency and computational efficiency. Finally, the remaining inliers are used to estimate the global geometric transformation model that aligns, warps, and blends the overlapping images, producing a satisfactory stitched image representation. Figure 3.6 provides an overview of the entire image stitching process.

3.6 Proposed Performance Evaluation Metrics

Mikolajczyk et al. (2005) introduced a widely recognized framework for evaluating the performance of feature detectors and descriptors. Metrics such as repeatability, recall and precision are commonly used to measure their effectiveness (Mikolajczyk et al., 2005; Mikolajczyk & Schmid, 2005). However, while these metrics are useful in assessing detection accuracy and consistency, they do not reflect how well the identified inliers contribute to the overall quality of stitched images in the context of image stitching. For instance, a feature detector with a high repeatability score does not necessarily imply that the extracted inliers are sufficient in terms of both quantity and spatial distribution across the overlapping areas to achieve effective and satisfactory image stitching. Effective image stitching often relies on having an adequate number of stable inliers that are well-distributed across the overlapping area between images. To address this limitation, this study introduces new evaluation metrics: the spread-overlap, S_o and the ratio of recall-to-spread-overlap denoted as RC/S_o . These metrics aim to provide a more comprehensive assessment of how feature detectors impact image stitching performance.

3.6.1 Spread-Overlap Measure

The spread-overlap, S_o metric, inspired by Marmol et al. (2017), is developed to analyse the spatial distribution of interest points within the overlapping areas of images. In their work, Marmol et al. (2017) utilized a uniform 10×10 grid cell mask within the eyepiece's field of view of an arthroscope to determine the proportion of grid cells that contain at least one interest point.

Rather than using a fixed 10×10 grid, the overlapping area of the images is divided into grid cells mask, where each square cell represents 0.25% of the overlapping area. This approach establishes a uniform grid of 400 square cells, covering the overlapping area, irrespective of its size. To ensure that each cell can hold several interest points, it is necessary for the sample images or their overlapping area to have dimensions of at least 100×100 pixels. The spread-overlap, S_o metric is derived as the ratio of the number of grid cells within the overlapping area containing at least one inlier N_i to the total number of grid cells within the overlapping area N_g . It can be mathematically expressed as follows:

$$S_o = \frac{N_i}{N_g} = \frac{|\{(\tilde{p}_i^g, \tilde{p}_j^g) | num(\tilde{p}_i^g, \tilde{p}_j^g) \geq 1\}|}{N_g} \quad \text{Equation 3.25}$$

where num indicates the number of pairs between the interest points $\tilde{p}_i^g$ from images I_i and their corresponding projective points $\tilde{p}_j^g$ from images I_j , within each grid cell. Figure 3.7 shows an example of S_o calculation, where 50 out of 400 grid cells in a 100×100 pixel overlapping area contain at least one inlier.

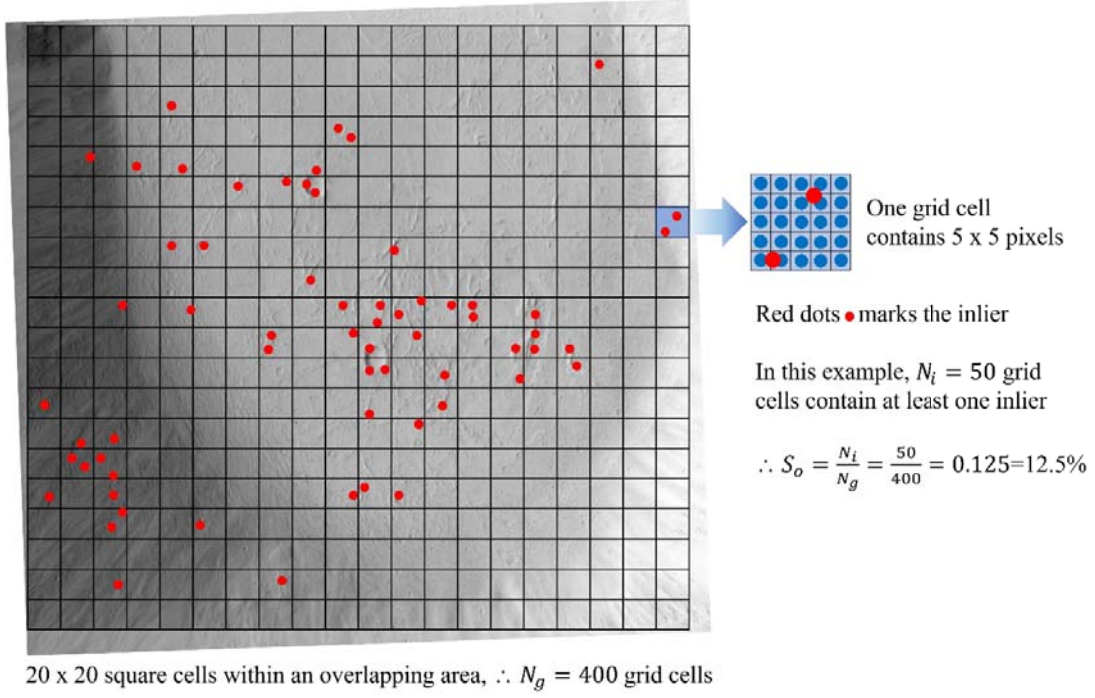
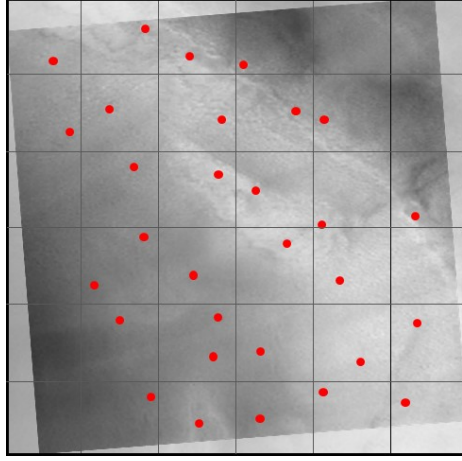


Figure 3.7: Example of spread-overlap, S_o calculation: 50 out of 400 grid cells within a 100×100 pixel overlapping area contain at least one inlier

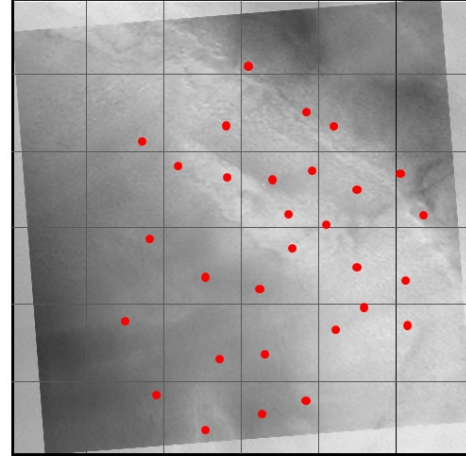
3.6.2 RC/S_o Score

The RC/S_o score indicates whether the distribution of inliers across the overlapping area is adequate for accurately estimating the global geometric transformation model, thereby ensuring images are stitched with minimal misalignment errors. Essentially, this score provides a means to assess the probability of success or failure in image stitching without needing to visually inspect the final stitched image. In mathematical terms, the RC/S_o metric is simply defined as the ratio of recall, RC to the spread-overlap, S_o metric.

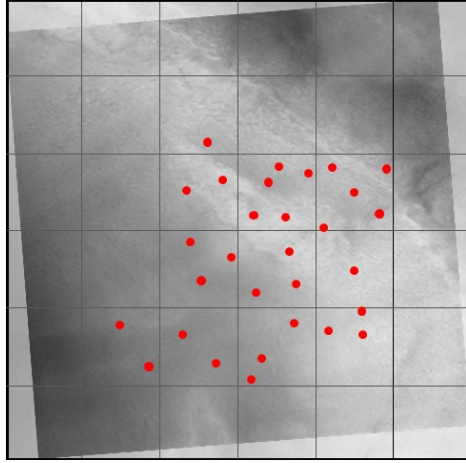
To better visualise this concept, imagine a simple 6×6 uniform square grid mask that overlays the overlapping area between two images, as depicted in Figure 3.8. In this setting, suppose there are 30 inliers (marked with red dots) distributed across the overlapping area



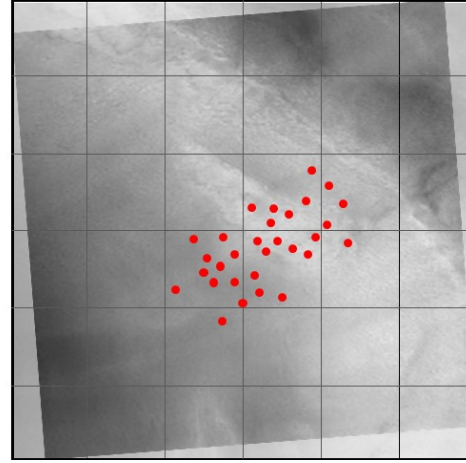
(a) $S_o = 0.778, RC/S_o = 0.918$



(b) $S_o = 0.611, RC/S_o = 1.169$



(c) $S_o = 0.306, RC/S_o = 2.333$



(d) $S_o = 0.167, RC/S_o = 4.275$

Figure 3.8: Visualization of RC/S_o concept with 30 inliers (marked with red dots) across a 6×6 grid in the overlapping region. RC is fixed at 0.714

under 4 different distribution conditions, with an RC measure of 0.714 (or 71.4%) held across all samples. A higher S_o measure occurs when inliers are widely distributed across the overlapping area (as shown in Figure 3.8(a) and Figure 3.8(b)), leading the RC/S_o scores to typically approach a value of one.

On the other hand, when inliers are clustered in a small area of the overlapping region (see Figure 3.8(c) and Figure 3.8(d)), the RC/S_o scores increase noticeably. Implicitly, a higher RC/S_o score indicates an increased risk of noticeable misalignment, distortion, and potential failure in image stitching, especially with more extreme RC/S_o scores. This issue arises from the limited spread of inliers that fails to provide sufficient data for accurately estimating the global geometric transformation model between the overlapping images. In the worst-case scenario, image stitching is more likely to fail with fewer inliers that clustered in small area.

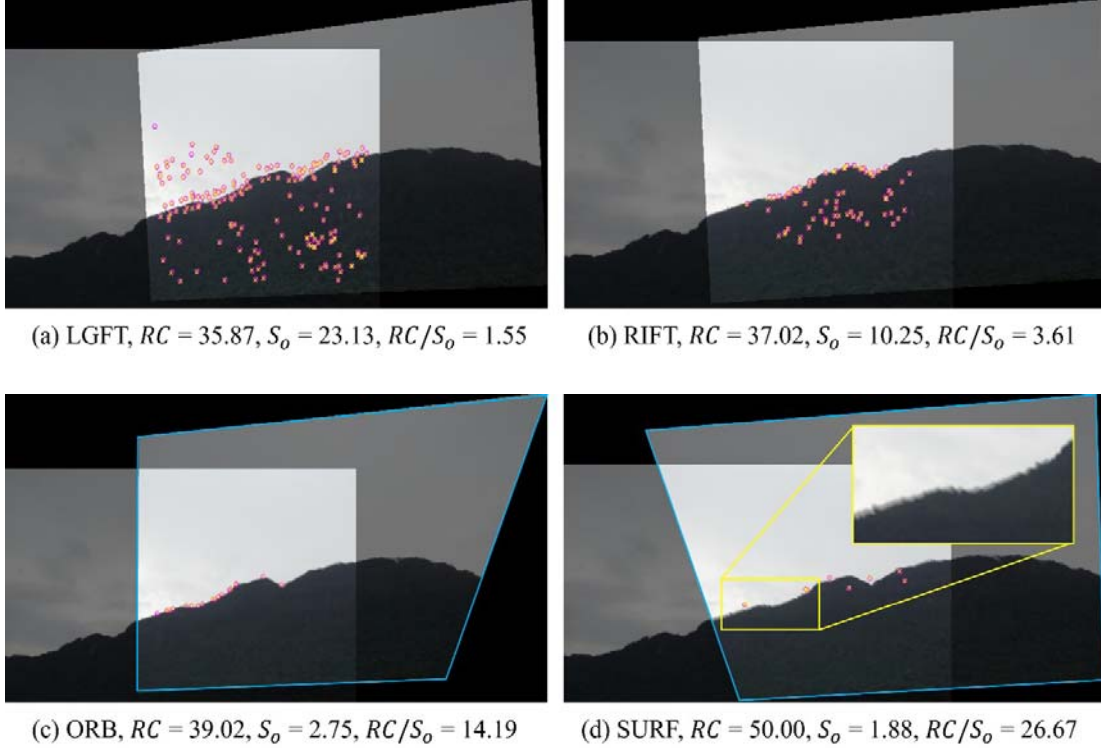


Figure 3.9: Examples of the proposed evaluation metrics, S_o and RC/S_o , for the LGFT, RIFT, ORB, and SURF. Yellow crosses (reference image) and magenta circles (target image) mark inlier locations. The yellow zoom-in highlights the misalignment, and the blue trapezoid shows distortion in the stitched images

To further validate the proposed evaluation metrics, Figure 3.9 illustrates how S_o and RC/S_o affect the final stitched images produced by LGFT, RIFT, ORB, and SURF. In Figure 3.9(a) and Figure 3.9(b), the RC/S_o values of 1.55 and 3.61 for LGFT and RIFT, respectively, indicate that the distribution of inliers across the overlapping area is adequate for accurately estimating the global geometric transformation, producing well-aligned stitched images with no misalignment errors. In contrast, ORB and SURF exhibit higher RC/S_o values of 14.19 and 26.67 due to clustering of inliers along the mountain edges, leading to distorted stitched images, as shown by the blue trapezoid shapes in Figure 3.9(c) and Figure 3.9(d). A zoomed-in view of the yellow rectangular area in Figure 3.9(d) clearly highlights the misalignment in the stitched image.

Figure 3.10 further demonstrates the effect of S_o and RC/S_o on the final stitched outputs from LGFT, KAZE, SIFT, and AGAST. Figure 3.10(a) and Figure 3.10(b) demonstrate that both LGFT and KAZE show a high number and broad distribution of inliers, corresponding to their low RC/S_o values of 0.43 and 2.45, respectively, resulting in well-aligned stitching. However, as the RC/S_o values rise due to a significantly lower number of inliers across the overlapping region, misalignment becomes evident, as highlighted by the yellow zoom-in areas in Figure 3.10(c) for the SIFT method, which has a high RC/S_o value of 28.57. In the worst-case scenario, stitching fails entirely, as illustrated in Figure 3.10(d) for the AGAST method, which has a very high RC/S_o value of 106.67, primarily due to insufficient data to accurately estimate the global geometric transformation between overlapping images.

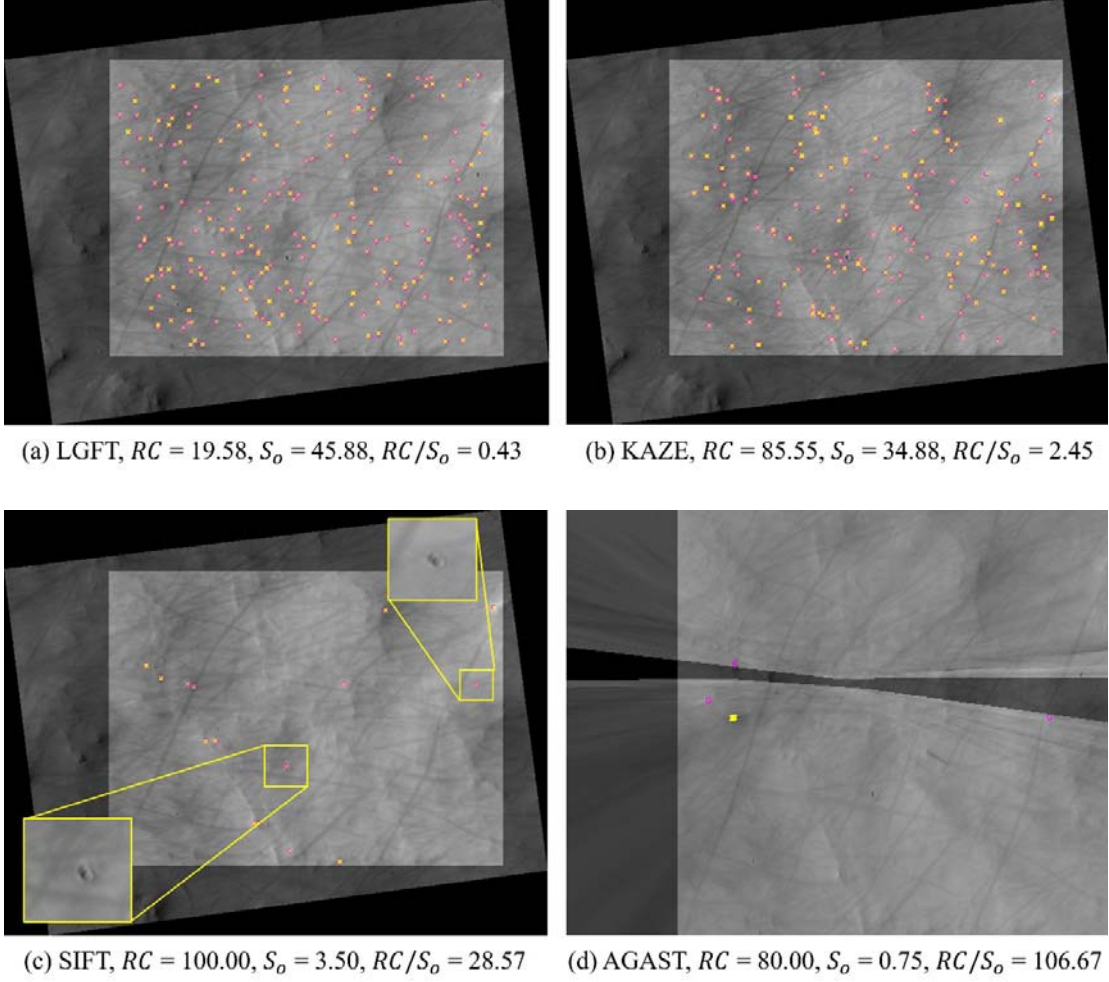
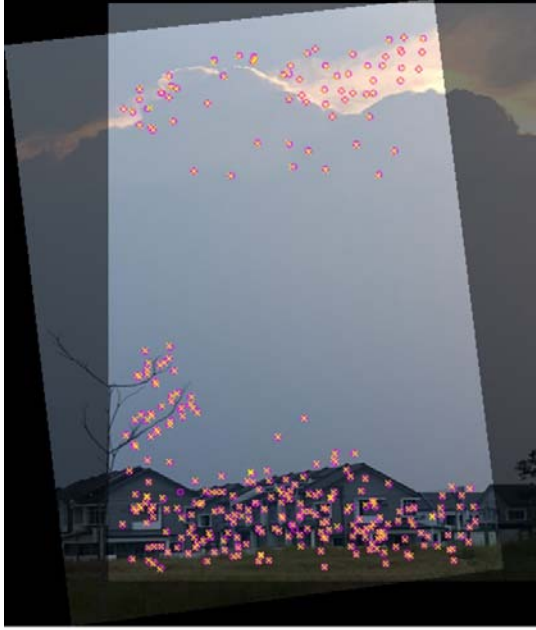


Figure 3.10: Examples of the proposed evaluation metrics, S_o and RC/S_o , for the LGFT, KAZE, SIFT and AGAST. Yellow crosses (reference image) and magenta circles (target image) mark inlier locations. The yellow zoom-in highlights the misalignment

Figure 3.11(a) provides another example of the ePDE method with a low RC/S_o value of 1.78. The images are perfectly stitched due to a sufficient number of inliers are widely spread across the upper and lower areas, with 32.13% covering the overlapping region, despite a large unfilled area in the centre. As the number of inliers decreases and their distribution becomes more clustered, the RC/S_o values increase, as seen from Figure 3.11(b) to Figure 3.11(d). This leads to greater misalignment errors, highlighted in



(a) ePDE, $RC = 57.14$, $S_o = 32.13$, $RC/S_o = 1.78$



(b) SIFT, $RC = 82.78$, $S_o = 15.38$, $RC/S_o = 5.49$



(c) BRISK, $RC = 81.25$, $S_o = 4.25$, $RC/S_o = 19.12$



(d) MSER, $RC = 100.00$, $S_o = 3.88$, $RC/S_o = 25.81$

Figure 3.11: Examples of the proposed evaluation metrics, S_o and RC/S_o , for the ePDE, SIFT, BRISK and MSER. Yellow crosses (reference image) and magenta circles (target image) mark inlier locations. The yellow zoom-in highlights the misalignment

the yellow zoom-in areas. Therefore, the results presented in Figure 3.9 to Figure 3.11 suggest that when inliers are few and clustered in smaller areas, reducing the spread-overlap, S_o , higher RC/S_o values significantly increase the chances of misalignment and even stitching failure, due to insufficient data for accurately estimating the global geometric transformation between the images.

3.7 Chapter Summary

This chapter addresses the shortcomings of existing state-of-the-art image stitching methods, particularly their limitations in handling near-uniform scenes. To overcome these challenges, two new feature-based approaches are introduced: the Enhanced Partial Differential Equation (ePDE) and the Log-Gabor Feature Transform (LGFT).

The ePDE method enhances intensity-based feature detection by modifying the Lorentz factor, derived from Einstein’s theory of special relativity (Bambi, 2018), and applying it to define the novel conductivity function in the nonlinear diffusion equation. To ensure stable diffusivity outputs, the contrast control parameter k is empirically selected within the range of 0.1 to 0.9. This method generates 12 nonlinear scale-spaces, structured into 5 octaves with 4 sublevels each, producing multi-scale, edge-preserving representations that improve feature detection in low-texture regions. The LGFT method overcomes the limitations of intensity-based detection by applying two-dimensional Log-Gabor filters in the Fourier domain (Li et al., 2020). With $n_o \in \{1,2,3,4,5,6\}$ orientations and $n_s = 4$ sublevels, it creates Log-Gabor scale-spaces that capture fine granule-like structural details at various orientations and scales. Both proposed methods utilize a common feature descriptor, generating a 64-element vector for each detected interest point.

To evaluate image stitching effectiveness, two new performance metrics are introduced: the spread-overlap, S_o and the recall-to-spread-overlap ratio, RC/S_o . The S_o metric quantifies the spatial distribution of inliers across a uniform 400 grid cells covering the overlapping image area, with each cell representing 0.25% of the total overlapping area. RC/S_o combines recall with spread-overlap, providing a more comprehensive evaluation of feature detector performance in the context of image stitching.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter presents a comprehensive evaluation of the proposed feature-based image stitching methods in comparison with several state-of-the-art techniques. The experiments are essential for verifying the robustness and effectiveness of the proposed methods, particularly in challenging near-uniform image scenes where stitching performance can vary significantly across different algorithms. The comparison includes established methods such as MSER (Zhang et al., 2015), SIFT (Lowe, 2004), SURF (Bay et al., 2006), BRISK (Leutenegger et al., 2011), KAZE (Alcantarilla et al., 2012), AKAZE (Alcantarilla et al., 2013), AGAST (Mair et al., 2010), ORB (Rublee et al., 2011), and the upright variant of RIFT (U-RIFT) (Li et al., 2020). These methods are selected due to the availability of their MATLAB implementations, ensuring consistency in testing and enabling fair comparison using standardized, pre-processed input images and default parameter settings.

Prior to performance evaluation, parameter tuning is conducted for the proposed methods to ensure optimal results. A visual analysis is also carried out to examine the differences in texture representation across the newly constructed nonlinear scale-spaces, providing a better understanding of how these differences influence inlier detection performance. The subsequent evaluation includes quantitative experiments using standard metrics such as recall, RC (Mikolajczyk & Schmid, 2005), along with the proposed spread-overlap, S_o and the recall-to-spread-overlap ratio, RC/S_o to provide an insightful performance measure. Qualitative experiments enable visual comparison of stitched outputs

to observe visual artifacts and misalignments. Finally, efficiency experiments assess the computational cost of each method based on processing time.

This chapter is divided into six sections. Section 4.2 presents the experimental tools and datasets employed for the implementation of the proposed methods. Section 4.3 describes the pre-analysis procedures and the parameter settings optimized for each method. In Section 4.4, a visual analysis of the nonlinear scale-spaces generated by the enhanced Partial Differential Equation (ePDE) and the Log-Gabor Feature Transform (LGFT) techniques is provided. Section 4.5 validates the robustness and effectiveness of the proposed methods by applying them to a set of image pairs and evaluating the resulting inliers using the newly proposed evaluation metrics. Finally, Section 4.6 provides a summary of the chapter’s key findings.

4.2 Experimental Tools and Datasets

The proposed methods and associated algorithms are implemented using the MATLAB Computer Vision System Toolbox, coupled with the mexOpenCV interface, supporting both image stitching and evaluation of performance. Additionally, algorithms provided from the Ralli diffusion code (Ralli, 2011), the RIFT source code (Li et al., 2020), the OpenSURF library (Evans, 2009), and the VLFeat open-source library (Vedaldi & Fulkerson, 2010) are adapted and employed to execute the proposed algorithms. The experiments are carried out on a Windows 10 64-bit system, powered by an Intel core i5-6300U CPU running at 2.40GHz with 8GB of RAM, and without GPU acceleration.

For the experimental datasets, the following sets of image pairs are used in the subsequent evaluation experiments:

i. 25 pairs of benchmark images

The nonuniform or feature-rich scene benchmark images are obtained from the Affine Covariant Features (2007) and Alcantarilla et al. (2012) collections. These images, shown in Figure 4.1, include the ‘bikes’ and ‘trees’ series for image blur, ‘leuven’ for illumination changes, ‘ubc’ for JPEG compression, and ‘iguazu’ for Gaussian noise effects. Each series comprises 6 images, with transformations applied incrementally.

ii. 70 pairs of general images

The general images are randomly selected from datasets provided by previous researchers (Li et al., 2020; Zaragoza et al., 2014), as well as online resource such as Photojournal: NASA’s Image Access Home Page (n.d.), Adobe Open Source Data Sets (n.d.), OpenCV Extra Data (n.d.), along with real images taken by the author. These image pairs represent real-world scenes chosen for their partially near-uniform regions, where certain areas are either homogeneous or exhibit low-texture. Such near-uniform scenes often result in diminish the accuracy and sensitivity of advanced feature detection methods. Some examples are illustrated in Figure 4.2.

iii. 123 pairs of HiRISE images

The High Resolution Imaging Science Experiment (HiRISE) images are obtained from the HiRISE | Image Catalog (n.d.), with dates ranging from April 23 to May 3, 2023. These images, capture by the HiRISE camera on the Mars Reconnaissance Orbiter (which was launched in 2005 and reached Mars

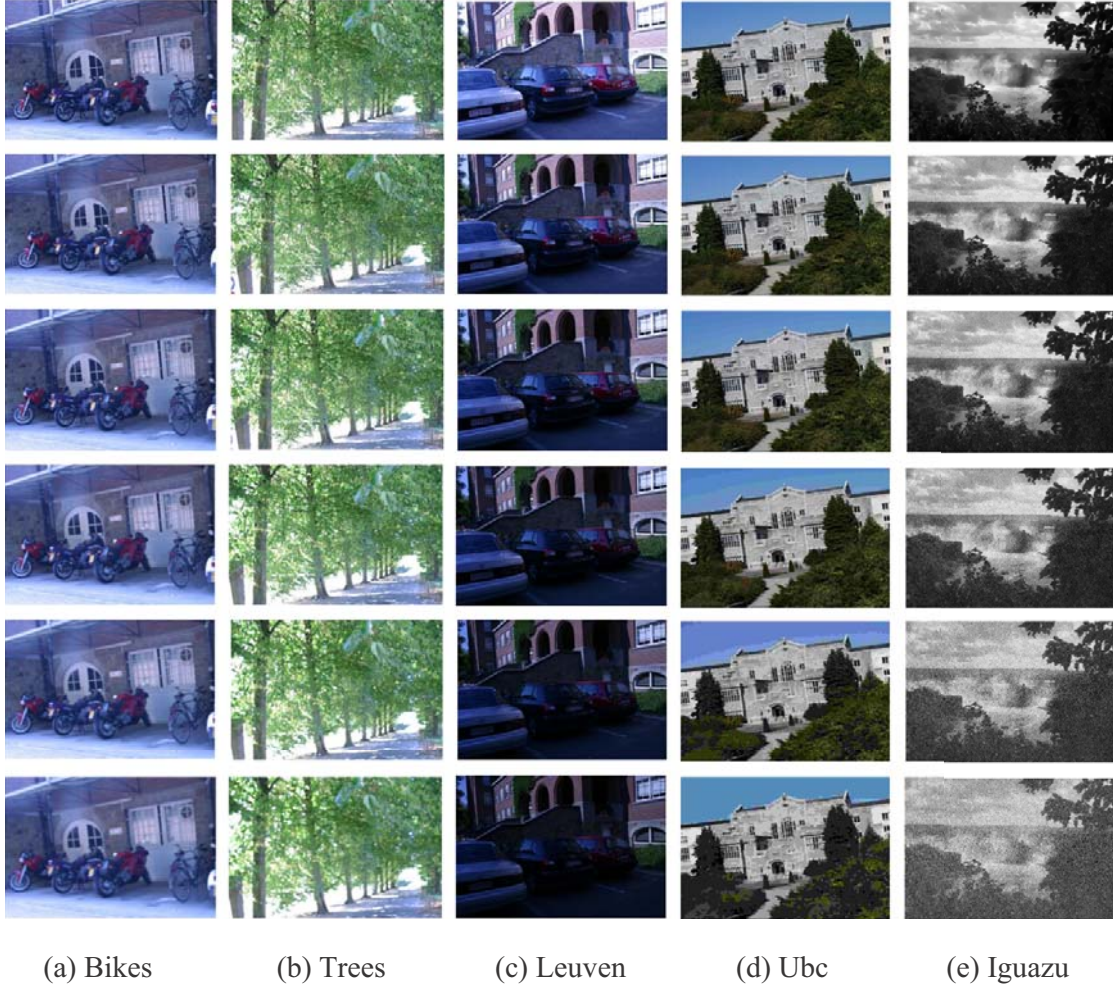


Figure 4.1: The benchmark image pairs include: (a)-(b) for blurred images, (c) for light variation, (d) for JPEG compression, and (e) for Gaussian noise. Each series contains 6 sequences of progressively transformed images (from top to bottom)

in 2006), provide high-resolution views of the Martian surface, with a resolution of up to 30 centimetres per pixel. The selected image pairs feature primarily near-uniform scenes in grayscale, available in both map projected and non-map projected formats and are provided in JPEG format. Some examples are shown in Figure 4.3. In these figures, the HiRISE image pairs are first cropped to regions that exhibit the most near-uniform scenes before proceeding to the processes of feature detection and image stitching.

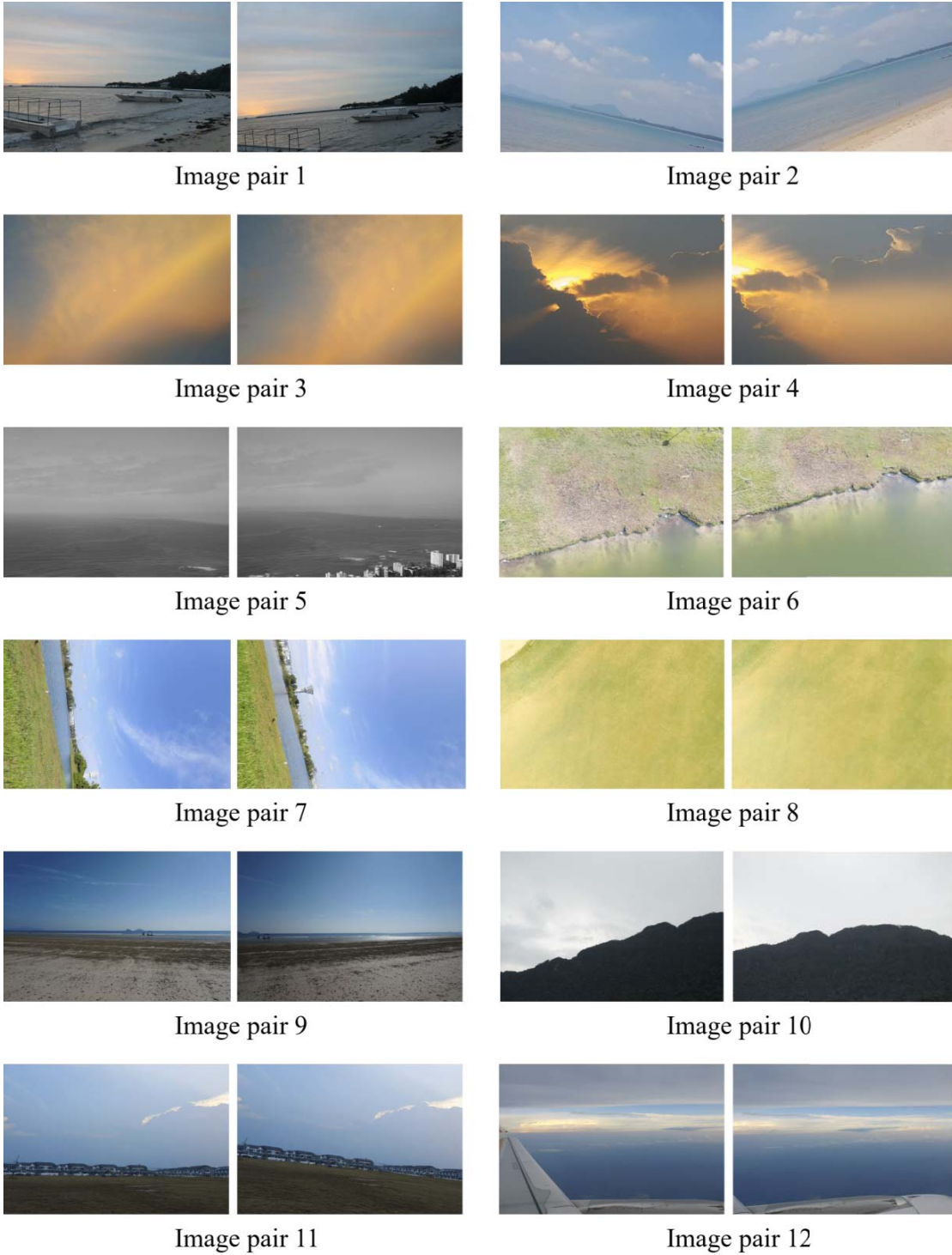


Figure 4.2: Examples of general image pairs illustrating real-world scenes with partially near-uniform regions

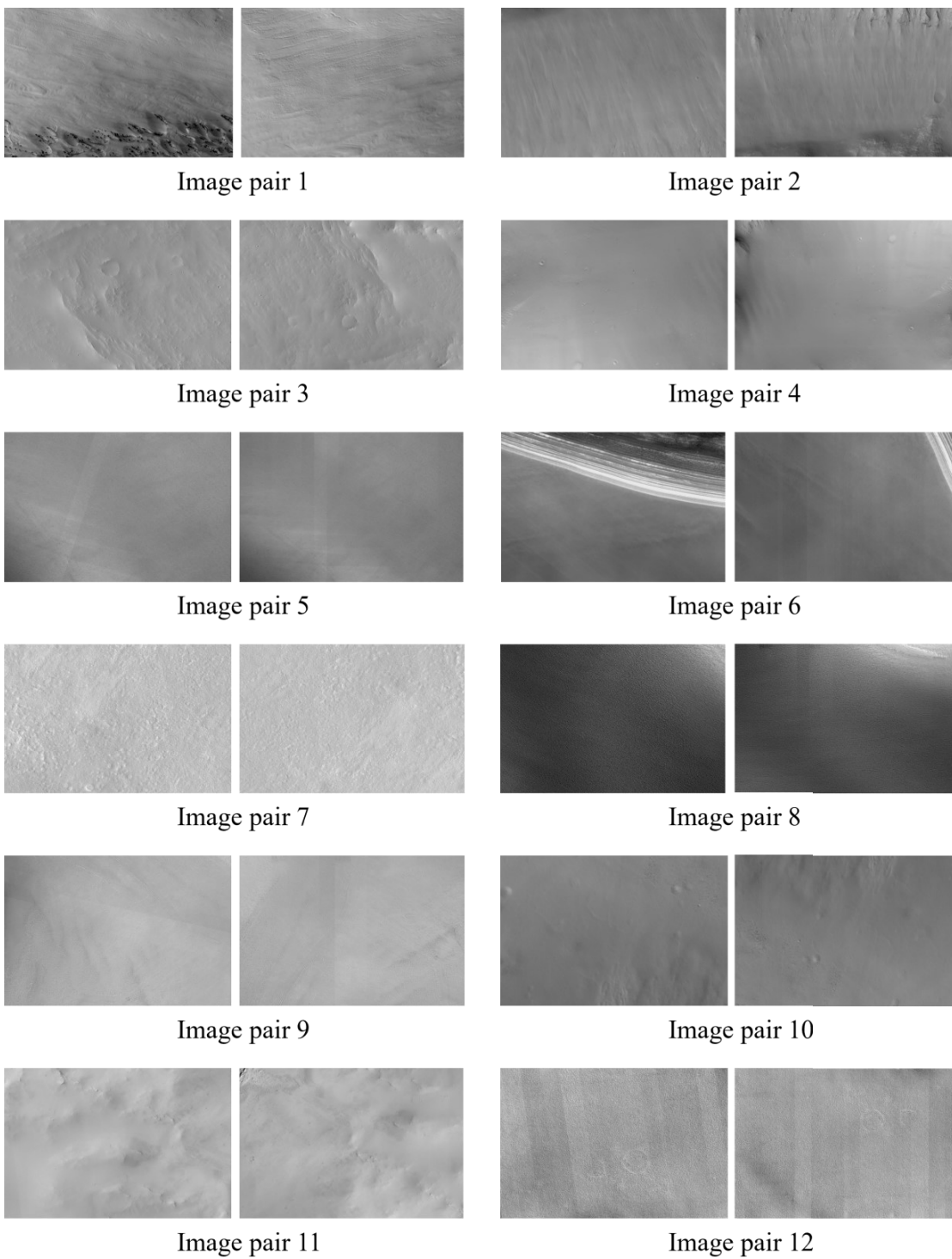


Figure 4.3: Examples of cropped HiRISE image pairs featuring predominantly near-uniform regions in grayscale

Before executing feature detection and image stitching, each input image is pre-processed to ensure a fair and consistent performance comparison. The pre-processing steps are as follows:

- i. All colour images are converted to grayscale to reduce computational complexity and focus on intensity features.
- ii. Both benchmark and general images are resized to an average of approximately 200,000 pixels to standardize input and ensure consistent processing.
- iii. The HiRISE images are resized to an average of approximately 300,000 pixels to standardize input and maintain consistency in processing. The larger near-uniform image size ensures sufficient detail for effective feature detection.
- iv. All images are denoised using a Gaussian kernel with a standard deviation of 0.5 to reduce noise and enhance feature detection accuracy.

The effectiveness and robustness of the proposed feature detection methods are evaluated in comparison to various advanced techniques, such as MSER, SIFT, SURF, BRISK, KAZE, AKAZE, AGAST, ORB, and U-RIFT. Table 4.1 provides a summary of these methods, highlighting their feature detectors, corresponding descriptors, targeted features, data formats, associated tools, key functions, and parameter settings applied during experimental implementation. The experiments employ the default configurations for each method. For clarity, the term "feature detectors" is used to refer to both the feature detector and its associated descriptor throughout this thesis. Note that the stitched images in the thesis intentionally retain the seam, allowing the reader to see more clearly how they are stitched together.

Table 4.1: Summary of feature detectors and descriptors used for validation, including feature types, data formats, associated tools, key functions, and parameter settings

Detector	Descriptor	Feature Type	Data Format	Tool / Function	Settings
MSER	SURF	Region	Non-binary, $\epsilon_c = 10$	MATLAB / detectMSERFeatures, extractFeatures	Default
SIFT	SIFT	Blob	Non-binary, $\epsilon_c = 10$	mexOpenCV / cv.SIFT, sift.detectAndCompute	Default
SURF	SURF	Blob	Non-binary, $\epsilon_c = 10$	MATLAB / detectSURFFeatures, extractFeatures	Default
BRISK	BRISK	Corner	Binary, $\epsilon_c = 100$	MATLAB / detectBRISKFeatures, extractFeatures	Default
KAZE	M-SURF	Blob	Non-binary, $\epsilon_c = 10$	MATLAB / detectKAZEFeatures, extractFeatures	Default
AKAZE	M-LDB	Blob	Binary, $\epsilon_c = 100$	mexOpenCV / cv.AKAZE, akaze.detectAndCompute; MATLAB / binaryFeatures	cv.AKAZE with 'Threshold' = 0.0001
AGAST	FREAK	Corner	Non-binary, $\epsilon_c = 10$	mexOpenCV / cv.AGAST; MATLAB / extractFeatures	Default

Table 4.1 continued

ORB	BRIEF	Corner	Non-binary, $\epsilon_c = 10$	mexOpenCV / cv.ORB, orb.detectAndCompute	cv.ORB with 'MaxFeatures' = 10000
U-RIFT	MIM	Corner, Edge	Non-binary, $\epsilon_c = 10$	RIFT source code; MATLAB / detectFASTFeatures (extracts corner & edge points from minimum & maximum moment maps)	Set 3: $\sigma_{f_s} = 0.75$, C = 1.6, $w = 2.5$ (from Table 4.2); detectFASTFeatures with 'MinContrast' = 0.05

4.3 Pre-analysis and Parameter Settings

Understanding how variation in certain parameters influence the methods' performance often provides valuable insights into their robustness and reliability. Therefore, parameter fitting is important for optimizing the algorithms to attain the best possible performance. This section provides a preliminary analysis to determine and set the values of certain parameters for the proposed methods before begin experimentation. These parameters are as follows:

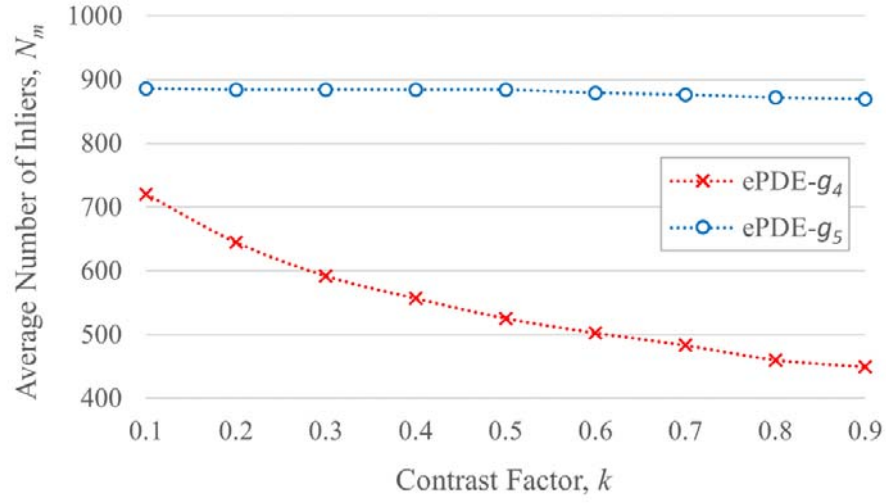
- i. For the enhanced Partial Differential Equation (ePDE) method: contrast factor k .
- ii. For the Log-Gabor Feature Transform (LGFT) method: number of scales n_s , number of orientations n_o , radial bandwidth σ_{f_s} , filter scaling factor C , and parameter w .

To simplify referencing in the subsequent discussion, the following acronyms will be used to represent the proposed feature detectors.

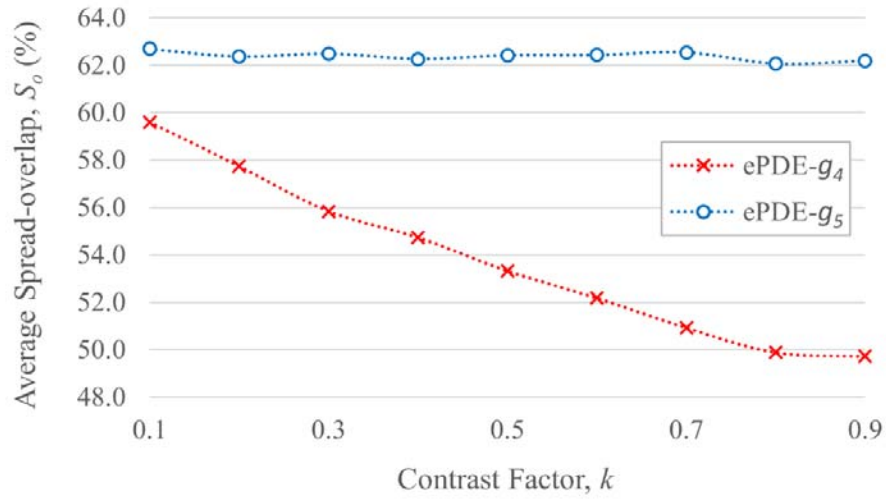
- i. For the Partial Differential Equations (PDE) with the new conductivity function of g_4 and g_5 , it is simply referred to as ePDE- g_4 and ePDE- g_5 , respectively.
- ii. For the Log-Gabor Feature Transform (LGFT), with the number of scales, n_s , number of orientations, n_o , and parameter set number, Set#, it is denoted as LGFT- $n_o.n_s.Set\#$. For example, LGFT-2.4.3 refers to the LGFT method with $n_o = 2$, $n_s = 4$ and the parameters in Set 3 as listed in Table 4.2.

4.3.1 Proposed Method 1: ePDE Method

The methods outlined by Alcantarilla et al. (2012) and Perona & Malik (1990) allow the contrast factor k in Equation 3.4 and Equation 3.5 to be set either by manually assigning a fixed value or by calculating it automatically from the histogram of the image gradient. Conversely, the choice of contrast factor k that regulate the diffusion weight g_4 and g_5 in Equation 3.9 and Equation 3.10 respectively, is based on an analysis of both conventional and proposed evaluation metrics for each feature detector tested on benchmark images. Figure 4.4 and Figure 4.5 illustrate the performance differences of ePDE- g_4 and ePDE- g_5 in relation to varying values of the contrast factor k across the 25 benchmark image pairs. For each k value, the performance evaluation results for each feature detector are calculated and averaged over all benchmark images.



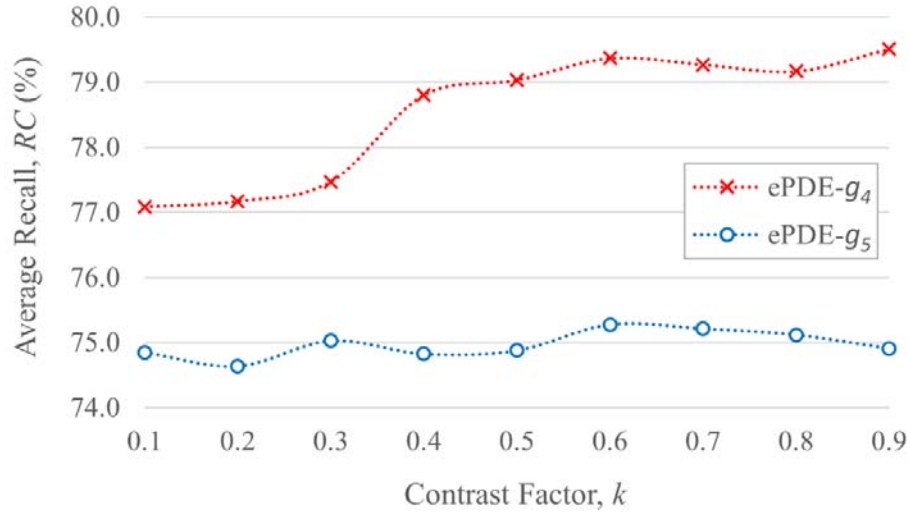
(a) N_m vs k



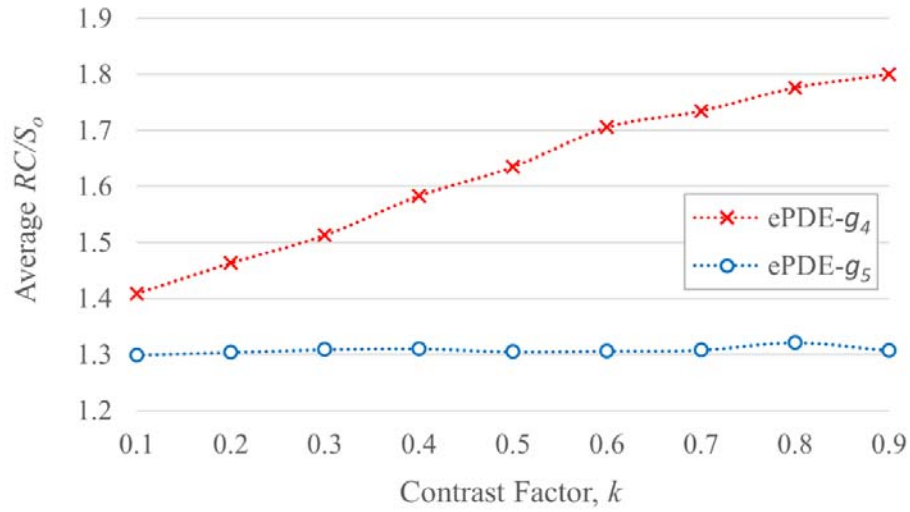
(b) S_o vs k

Figure 4.4: Performance comparison of ePDE- g_4 and ePDE- g_5 based on the number of inliers, N_m and spread-overlap, S_o relative to different contrast factors k . The metrics are averaged over 25 benchmark image pairs

In Figure 4.4(a) and Figure 4.4(b), a declining trend is observed in both the average number of inliers and spread-overlap, S_o for the ePDE- g_4 algorithm as the contrast factor k increases. This outcome is anticipated, as the nonlinear scale-spaces produced by ePDE- g_4



(a) RC vs k



(b) RC/S_o vs k

Figure 4.5: Performance comparison of ePDE- g_4 and ePDE- g_5 based on the recall, RC and RC/S_o relative to different contract factors k . The metrics are averaged over 25 benchmark image pairs

tend to show diminished local structural detail at elevated contrast factor k levels (see Figure 4.10). Conversely, the performance of ePDE- g_5 remains relatively consistent, showing only a minor decline across different contrast factors k values. The inliers detected by ePDE- g_5

are more numerous and spread wider within the overlapping area than those detected by ePDE- g_4 . This stability is attributed to the well-preserved dominant structure in its nonlinear scale-space produced by ePDE- g_5 (see Figure 4.10), maintained by its stable diffusivity as regulated by Equation 3.10.

Figure 4.5(a) shows that ePDE- g_4 retain slightly more inliers than ePDE- g_5 across various contrast factors k , suggesting that ePDE- g_4 outperforms ePDE- g_5 , as measured by the recall, RC metric. In Figure 4.5(b), the average RC/S_o result for ePDE- g_5 shows greater consistency than that of ePDE- g_4 . This consistency implies that ePDE- g_5 is likely to produce less distortion in the resulting stitched image. In other words, ePDE- g_4 is less robust at higher contrast factors k , and is likely to result in misaligned image stitching. To balance the trade-off between higher computational complexity at lower contrast factors k (due to more inliers) and reduced robustness at higher contrast factors k , as illustrated in Figure 4.4 and Figure 4.5, a contrast factor of $k = 0.5$ is selected for the proposed ePDE- g_4 and ePDE- g_5 methods in the subsequent experiments.

4.3.2 Proposed Method 2: LGFT Method

Kovesi (2003) suggests that to achieve uniform coverage of the Log-Gabor spectrum, the radial bandwidth, σ_{f_s} in Equation 3.18, and the filter scaling factor, C in Equation 3.24 should be set to the values specified in Table 4.2. As well, the parameter w in Equation 3.24 plays a crucial role for the feature detection algorithm, as it significantly influences the responses of Hessian determinant matrix by controlling the size of the box filter that approximates the Laplacian of Gaussian. The parameter values for w presented in Table 4.2 are derived from Figure 4.6 and Figure 4.7, based upon the number of inliers identified

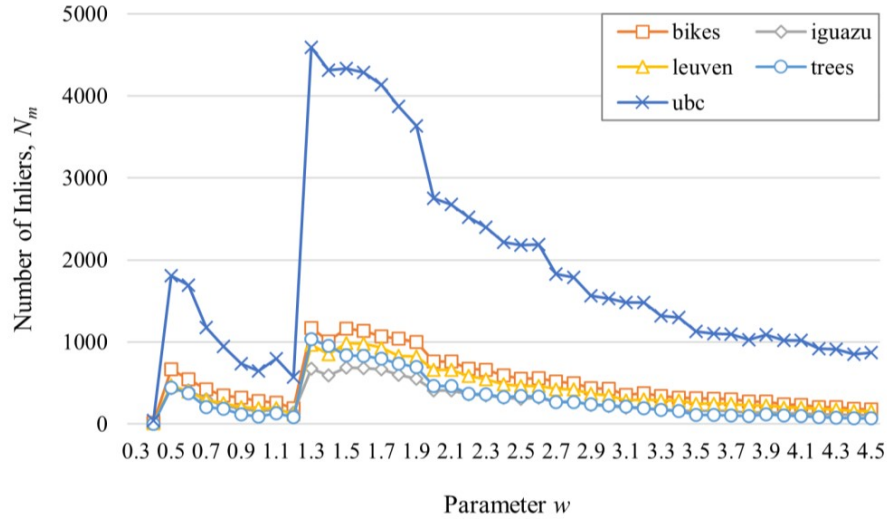
Table 4.2: Parameter settings for the Log-Gabor filter and the box filter approximation of the Laplacian of Gaussian. The radial bandwidth σ_{f_s} and scaling factor C are based on Kovési (2003), while the parameter w is derived from the results shown in Figures 4.6 and 4.7

Parameter Settings	Radial Bandwidth, σ_{f_s} Kovési (2003)	Filter Scaling Factor, C Kovési (2003)	Weight Parameter, w
Set 1	0.55	3.0	1.3
Set 2	0.65	2.1	1.9
Set 3	0.75	1.6	2.5
Set 4	0.85	1.3	3.0

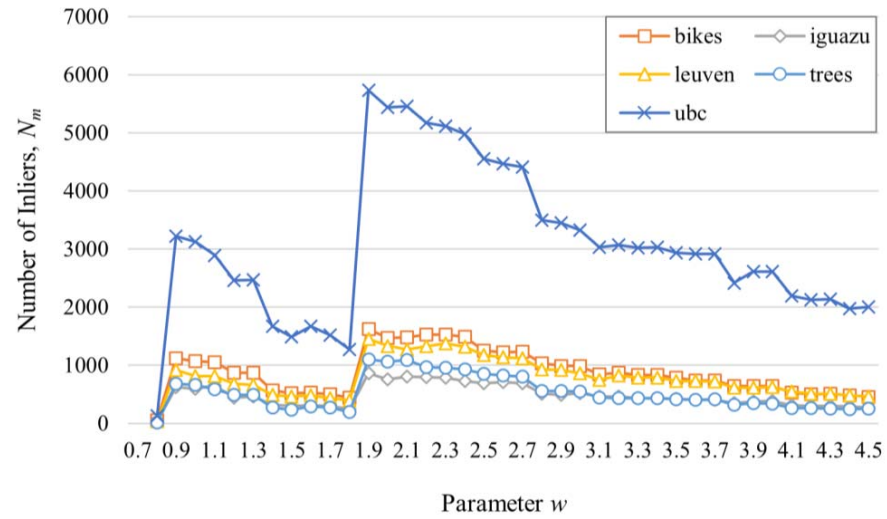
in nonuniform scene benchmark images for various radial bandwidth σ_{f_s} and filter scaling factor C .

Figure 4.6(a) and Figure 4.6(b) present the results for the first pair of images from each series in the benchmark datasets using parameter Set 1 and Set 2, respectively. Similarly, Figure 4.7(a) and Figure 4.7(b) present the results using parameter Set 3 and Set 4. In all cases, the number of scales n_s is set to 4, and the number of orientations n_o is set to 2. The results in Figure 4.6 and Figure 4.7 are based on the number of inliers, N_m that vary with different values of parameter w . Notice that the parameter w has significant impact on the number of inliers, N_m detected for different radial bandwidths σ_{f_s} and filter scaling factor C . Detection of a large number of inliers tends to favour specific parameter w for different sets of σ_{f_s} and C . Therefore, the value of parameter w as shown in Table 4.2, that yields the highest number of inliers, N_m for each combination of σ_{f_s} and C , is designated as the optimal weight for determining the box filter size in Equation 3.24. Besides, the

findings also indicate that the proposed method performs equally well in nonuniform scene images, as confirmed by visual inspection of the stitched images.

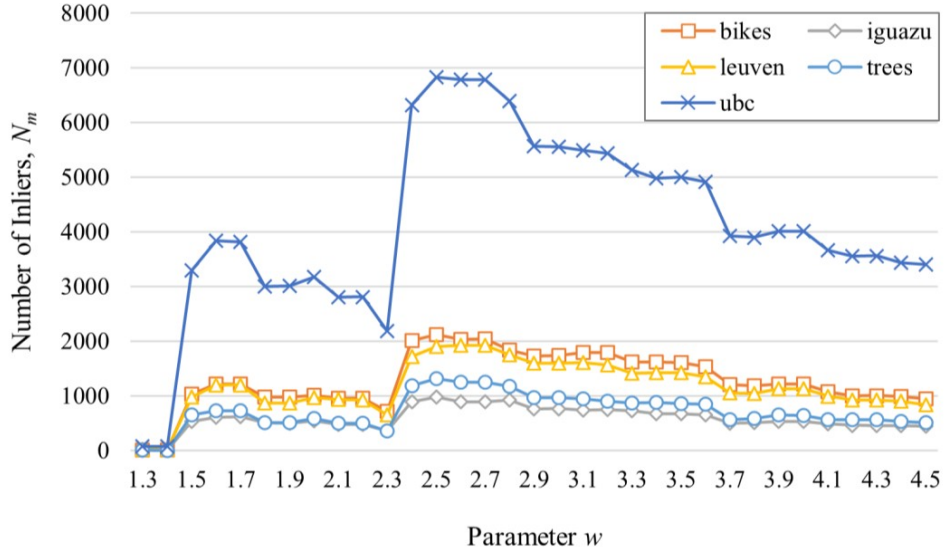


(a) $n_o = 2, n_s = 4$ & parameter Set 1: $\sigma_{f_s} = 0.55, C = 3.0$

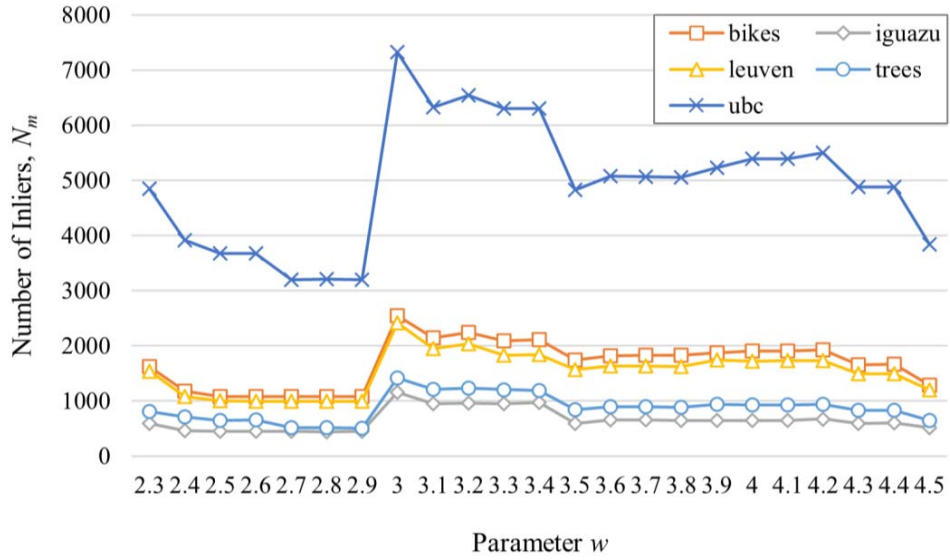


(b) $n_o = 2, n_s = 4$ & parameter Set 2: $\sigma_{f_s} = 0.65, C = 2.1$

Figure 4.6: Analysis of parameter w using the first image pair from each benchmark dataset, with parameter sets, Set 1 and Set 2 (see Table 4.2)



(a) $n_o = 2$, $n_s = 4$ & parameter Set 3: $\sigma_{f_s} = 0.75$, $C = 1.6$



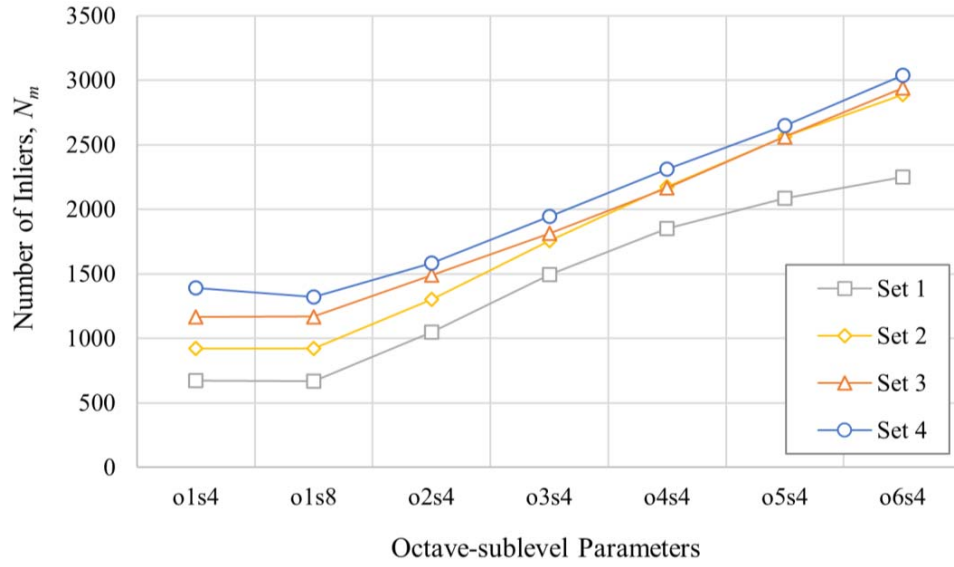
(b) $n_o = 2$, $n_s = 4$ & parameter Set 4: $\sigma_{f_s} = 0.85$, $C = 1.3$

Figure 4.7: Analysis of parameter w using the first image pair from each benchmark dataset, with parameter sets, Set 3 and Set 4 (see Table 4.2)

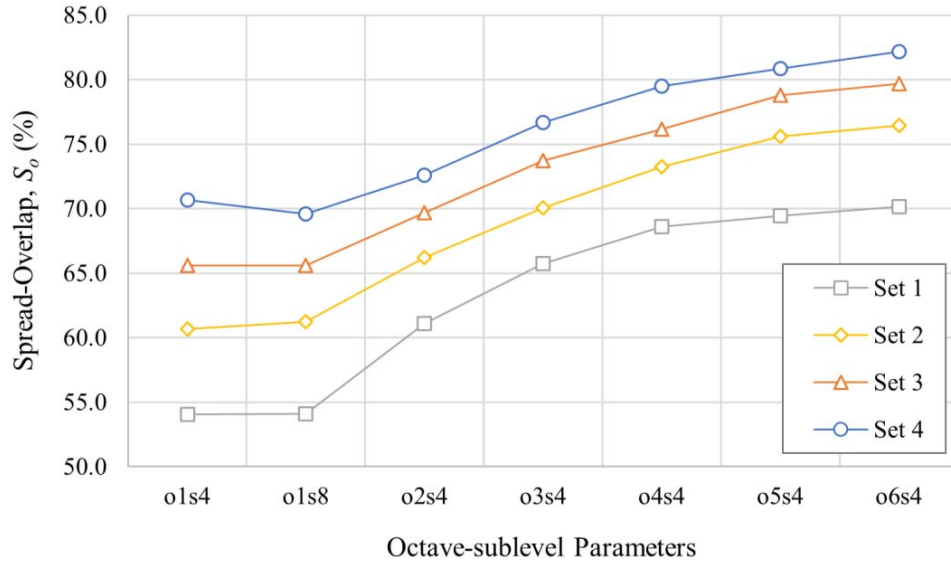
The performance of matching and the computational complexity of the proposed LGFT method are significantly influenced by the number of scales, n_s and orientations, n_o

used in the 2D Log-Gabor filter. In general, increasing the number of orientations enhances the richness of the scale-space's localization structure. However, this increase also intensifies computational complexity, potentially slowing down the process. With a reduced number of orientations, the resulting scale-spaces may not provide adequate local structural information, which can lead to deficiency of inliers. Therefore, a parameter study is conducted to examine their effects on matching performance, considering $1 \leq n_o \leq 6$. Since scale-spaces tend to lose structural detail at coarser scales (with a higher number of scale levels), n_s is fixed at 4 for all n_o in subsequent experiments, mirroring the scale levels per octave used in SURF. Additionally, n_s is set to 8 for $n_o = 1$ to compare its performance against the configuration where n_s is set to 4 for $n_o = 1$. Nevertheless, the experimental findings reveal that both configurations of n_s yield similar outcomes, suggesting that an increase in n_s is unnecessary.

Figure 4.8 and Figure 4.9 illustrate the performance results for different parameter sets outlined in Table 4.2, assessed based on various combinations of n_o and n_s . For simplicity, each combination is referred to as $o\#s\#$, indicating the number of octaves, $o\#$ and number of sublevels, $s\#$. Each result point in Figure 4.8 and Figure 4.9 represents the average responses derived from all benchmark datasets presented in Figure 4.1. The results in Figure 4.8 suggest that an increased number of orientations, n_o , associated with $o\#$, yields more detailed information in the generated scale-spaces. This leads to a greater number of inliers (see Figure 4.8(a)) and broader spatial distribution (see Figure 4.8(b)). However, a greater number of orientations, n_o substantially raises the algorithm's computational complexity, indicating longer processing time for interest point detection (see Figure 4.9(b)).

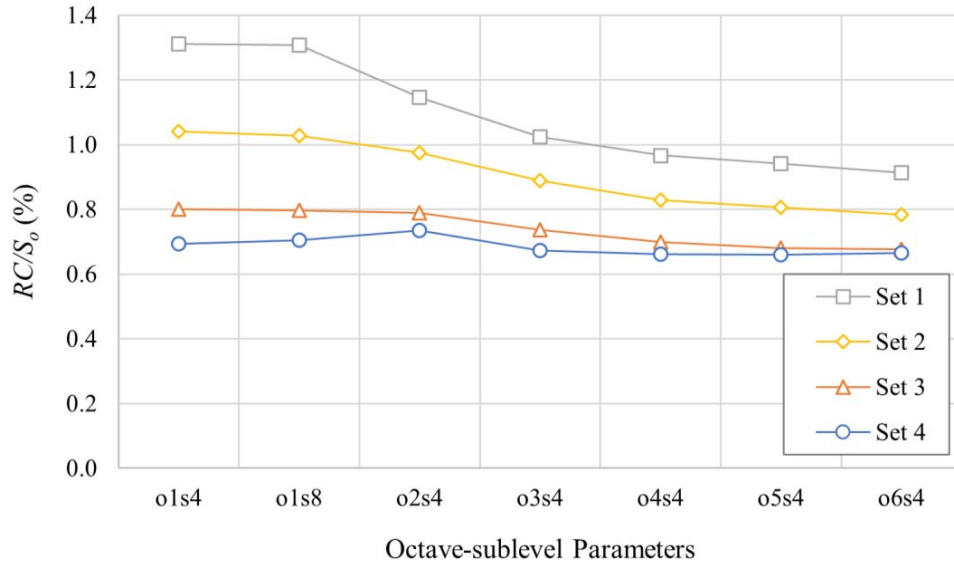


(a) N_m vs different groups of n_o and n_s

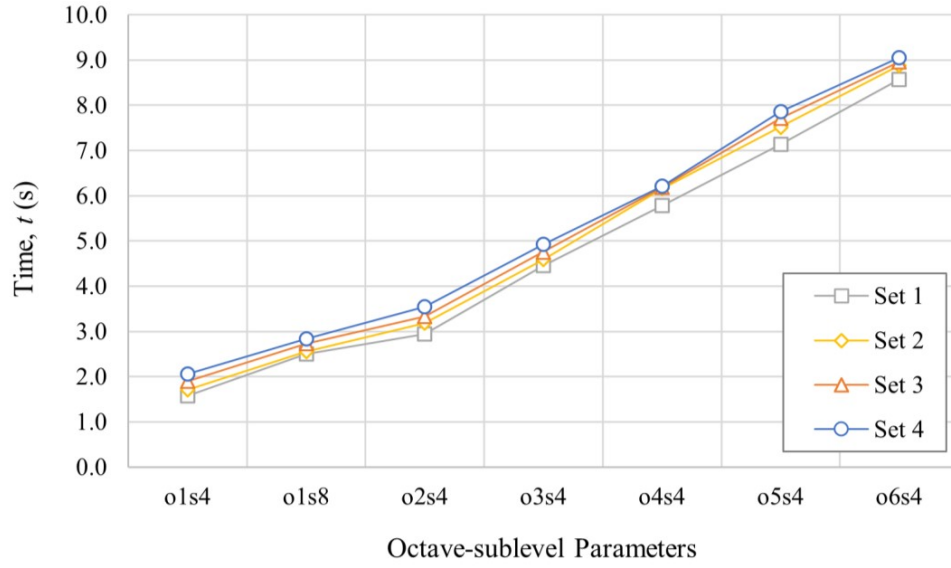


(b) S_o vs different groups of n_o and n_s

Figure 4.8: Evaluation of octave-sublevel parameters for the number of inliers, N_m and spread-overlap, S_o , using parameter sets from Table 4.2, applied across all benchmark datasets



(a) RC/S_o vs different groups of n_o and n_s



(b) Execution time t for single image vs different groups of n_o and n_s

Figure 4.9: Evaluation of octave-sublevel parameters for RC/S_o and the execution time t for single image, using parameter sets from Table 4.2, applied across all benchmark datasets

Increasing the number of scales to $n_s = 8$ does not improve performance when compared to $n_s = 4$, as seen by comparing the responses of *ols4* and *ols8* in Figure 4.8 and Figure 4.9. This implies that scale levels greater than $n_s = 4$ have a minimal effect on feature detection performance. Notably, the performance results improved from parameter Set1 to Set4, as demonstrated in Figure 4.8 and Figure 4.9(a). However, this improvement also results in increased computation time due to the increase in the number of orientations, n_o and scales, n_s (see Figure 4.9(b)). Considering the performance of these parameters and the importance of processing time, the settings are therefore set to Set3, using $n_o = \{1, 2, 4\}$ and $n_s = 4$, for the subsequent experiments. This includes three LGFT configurations as follows: LGFT-1.4.3, LGFT-2.4.3, and LGFT-4.4.3.

4.4 Scale-space Comparisons

In this study, constructing nonlinear scale-spaces is an essential step to improve feature detection algorithms. This step allows for a structured analysis of images across various nonlinear scale-spaces, thereby enhancing robustness and improving the accuracy of detected inliers via non-maxima suppression. The new nonlinear scale-spaces generated by the proposed ePDE and LGFT methods are particularly effective at detecting reliable interest points in near-uniform scene images, a task that poses significant challenges for many advanced feature detection techniques. This section will analyse the nonlinear scale-spaces generated using the proposed conductivity functions, g_4 and g_5 , and the Log-Gabor feature transform, against those conventionally generated using Gaussian kernel and difference of Gaussian (*DoG*). The comparison will highlight their differences and the implications in feature detection performance.

4.4.1 Proposed Method 1: ePDE Method

In Figure 4.10, the nonlinear scale-space structures of an input image are illustrated for comparison. These scale-spaces are computed using conductivity functions: (a) g_2 as defined in Equation 3.5, (b) g_4 from Equation 3.9, and (c) g_5 from Equation 3.10, applied at different contrast factors value k . In Figure 4.10, each resulting scale-space image has been cropped into a square shape, presenting only the 12th layer of the scale-spaces corresponding to each conductivity function across various contrast factor values k . It is clearly shown that these conductivity functions operate at different diffusivity rates, resulting in scale-space images that exhibit varying degrees of smoothness and blurriness.

Nonlinear diffusion filtering addresses a significant constraint of the standard Gaussian smoothing by preserving object boundaries while smoothing non-boundary regions. This is clearly shown in Figure 4.10(a) and Figure 4.10(c), presenting examples of nonlinear scale-spaces generated by the conductivity functions, g_2 and g_5 , respectively. On the other hand, the proposed g_4 (see Figure 4.10(b)) smooths the input images more rapidly, leading to accelerated blurring of structures in the resulting scale-space. The proposed g_5 (see Figure 4.10(c)) smooths the input image at a notably slower rate compared to g_2 and g_4 , preserving most prominent structures and keeps strong image edges largely intact, even when higher evolution time units are applied. In addition, at higher contrast factors k , the scale-spaces generated by g_2 and g_4 show noticeable blurriness and rapidly lose prominent structures, while scale-spaces generated by g_5 keep preserving nearly all structural information, including strong image edges. The differences observed in the scale-spaces generated by g_2 , g_4 and g_5 , as depicted in Figure 4.10, will in fact influence the detection of local maxima at various scale levels. This, in turn, shapes the feature detection performance discussed in the subsequent Section 4.5.

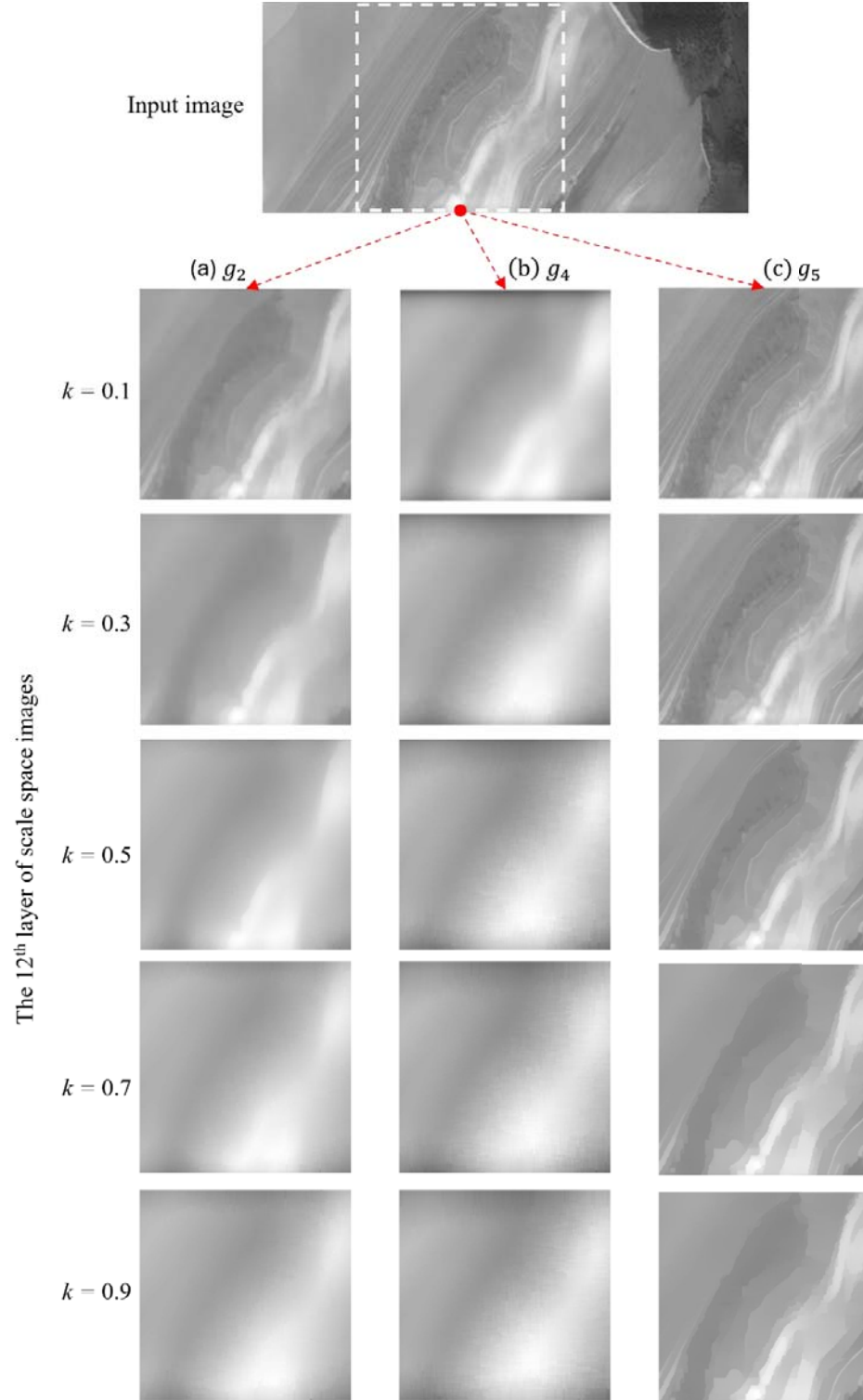


Figure 4.10: An example of nonlinear scale-spaces generated with the conductivity functions g_2 , g_4 and g_5 at different contrast factors k . Each image, cropped into a square shape, represents the 12th layer of the scale-space

4.4.2 Proposed Method 2: LGFT Method

Figure 4.11 illustrates a comparison of scale-spaces generated by various methods: the Gaussian kernel, L , the different of Gaussian, DoG , the nonlinear diffusion filter, L^t and the Log-Gabor feature transform filter, $A_{s,o}$. These results are computed using an image of Mars's surface, sourced from Photojournal: NASA's Image Access Home Page (n.d.). This image captures the central peak of Moreux Crater on the right side and a relatively uniform surface texture on the left. As illustrated in Figure 4.11(a) and Figure 4.11(b), the Gaussian kernel, L and DoG filters result in noticeably greater blurring at larger scales.

Figure 4.11(c) demonstrates that, as expected, the nonlinear diffusion function, L^t tends to maintain object boundaries while progressively blurring other regions as the evolution time increases (Alcantarilla et al., 2012). Conversely, the Log-Gabor function, $A_{s,o}$ excels at enhancing image texture, creating granule-like features at coarser scales (see Figure 4.11(d)). Therefore, considering the texture characteristics of the resulting scale-spaces, the Log-Gabor-based LGFT is anticipated to detect a greater number of interest points than SIFT that relies on Gaussian and DoG scale-spaces, and KAZE that uses nonlinear diffusion scale-spaces.

The lower part of Figure 4.11 presents a comparison of matched interest points, or inliers, detected by SIFT, KAZE and LGFT, alongside the resulting stitched images. Clearly, LGFT detected a higher number of inliers, with a wider spatial distribution across the overlapping areas than both SIFT and KAZE. As a result, the final stitched image is precisely aligned when using the LGFT method, whereas the SIFT and KAZE exhibit misalignments in object placement, as shown in the zoomed views.

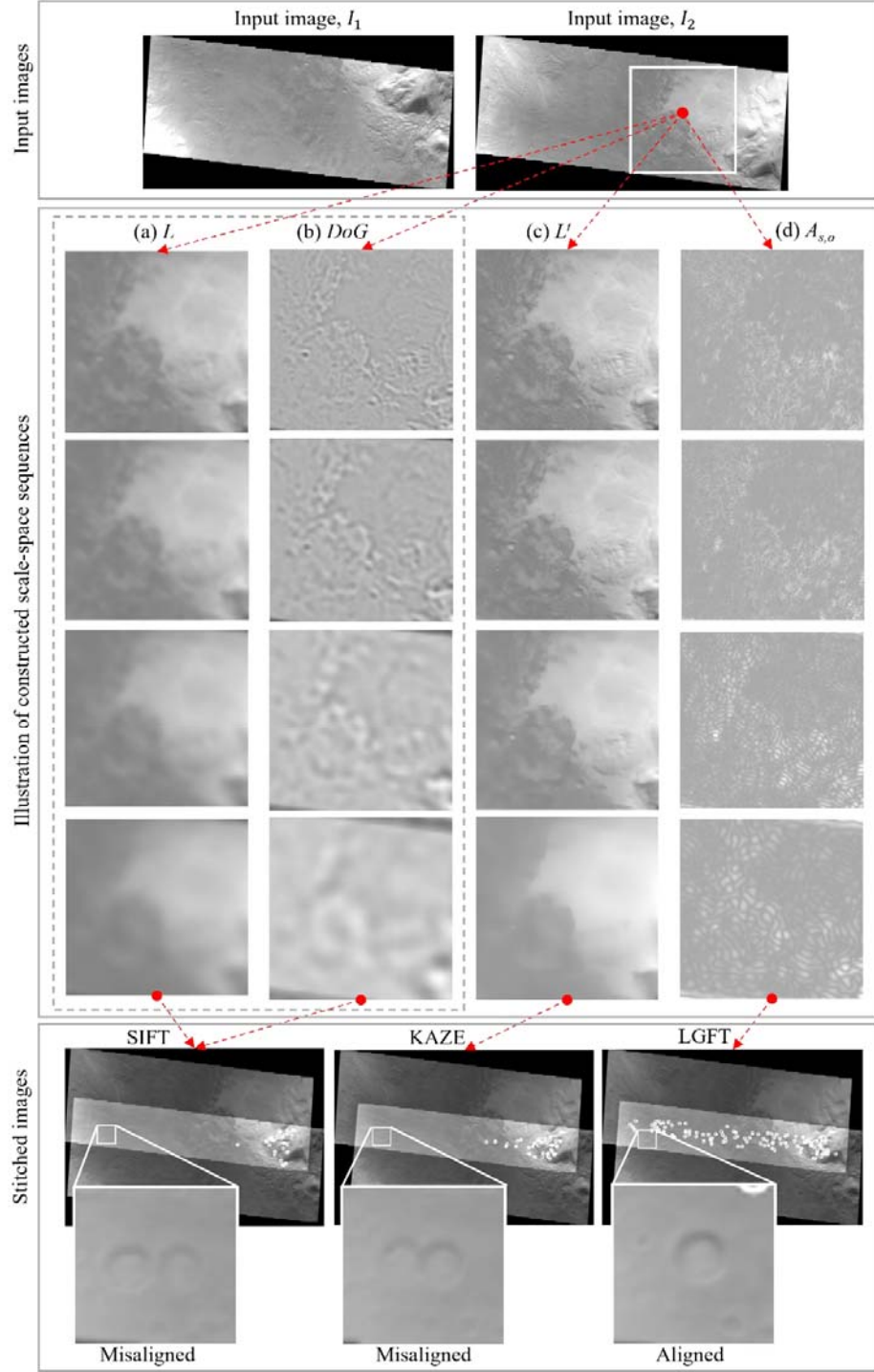


Figure 4.11: Comparison of various scale-space generation methods. Top row: Input image pair. Middle row: Cropped images showing (a) Gaussian, L ; (b) Different of Gaussian, DoG ; (c) nonlinear diffusion, L^t ; and (d) Log-Gabor, $A_{s,o}$ scale-spaces. Bottom row: Matching and stitching effects produced by SIFT, KAZE, and LGFT

4.5 Performance Comparisons

The robustness and effectiveness of the proposed methods are evaluated through a series of experimental comparisons with several leading feature detection techniques, with particular emphasis on image stitching in near-uniform scenes. These comparisons are structured into three main categories. First, a quantitative analysis is conducted to numerically evaluate the performance of each feature detector, focusing on their success rate in stitching near-uniform images. This includes analysis based on the number of inliers, recall, spread-overlap, and RC/S_o metrics. Second, a qualitative comparison involves visual inspection to identify the strengths and weaknesses of each feature detector in producing high-quality stitched images. Finally, execution efficiency is evaluated by measuring and comparing the execution time required by each feature detector.

4.5.1 Quantitative Analysis

This section presents a numerical evaluation of the proposed ePDE and LGFT to justify their superiority over several advanced feature extraction techniques in the context of image stitching. The evaluation utilizes a dataset comprising 25 pairs of benchmark images, 70 pairs of general images, and 123 pairs of HiRISE images, all selected to support the analysis of feature detection and image stitching. The benchmark images are notable for their photometry transformation characteristics such as blurriness, light changes, JPEG compression, and Gaussian noise transformation (see Figure 4.1). In contrast, the general images depict real-world scenes partially with near-uniform regions (see Figure 4.2). The HiRISE images predominantly feature near-uniform scenes (see Figure 4.3).

In the context of this study, not all feature detection methods are capable of extracting sufficient and reliable inliers from images that exhibit homogeneous or featureless regions, an essential requirement for successful image stitching. To assess the image stitching success rate for each feature detector, the alignment of each stitched image pair is visually inspected to ensure the final stitched image is free of severe distortion effects. Besides images that fail to stitch, those with significant distortion effects that are difficult to repair or fine-tune using warping, blending, and seam-cutting methods (Miao et al., 2023; Zaragoza et al., 2014) are excluded from the count. The success rate of image stitching is expressed as the percentage of accurately stitched images relative to the total number of sample image pairs.

Figure 4.12 presents the image stitching success rates for the proposed ePDE and LGFT methods in comparison to advanced feature detection techniques across all benchmark

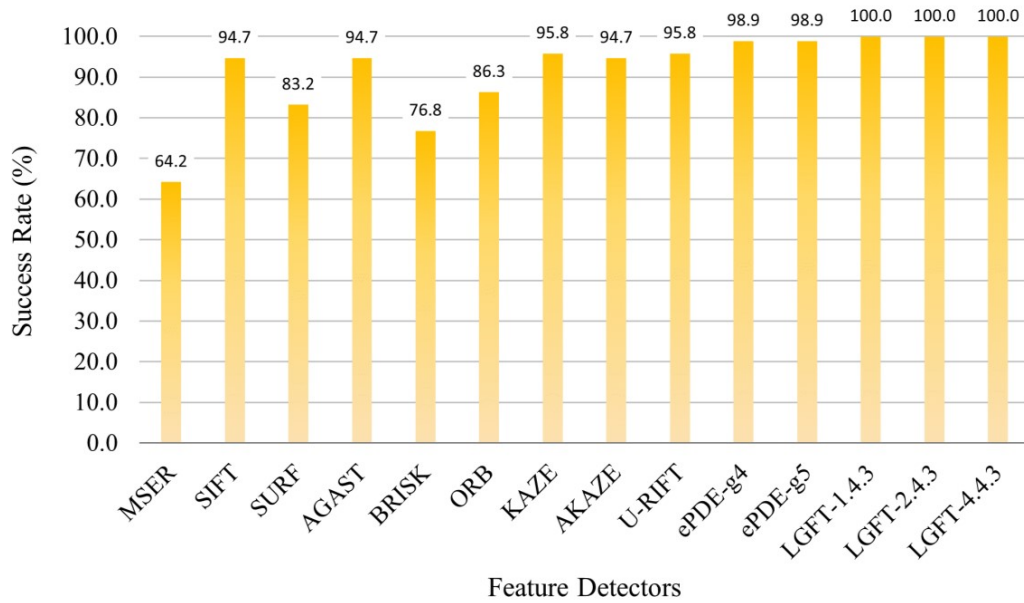


Figure 4.12: Success rate of different feature-based image stitching across 95 pairs of benchmark and general images

and general image pairs. The 25 benchmark image pairs are typically rich in features, but they gradually exhibit blurriness, reduced brightness, loss of detail, and increased noise. These characteristics allow for easy inliers detection using all feature detectors, resulting in satisfying image stitching outcomes. In contrast, the remaining 70 pairs of partially near-uniform scene images significantly challenge the performance of feature detectors. The LGFT method particularly excels in stitching near-uniform images, attaining a perfect success rate of 100% and surpassing other feature detectors. This indicates that LGFT is highly effective and robust to these types of images, generating sufficient and well-distributed inliers in the overlapping areas of partially near-uniform scenes, as further demonstrated in the subsequent results illustrated in Figure 4.13 to Figure 4.16.

Following LGFT, the proposed ePDE method ranks second, hindered only by false inliers detected in one image pair with a narrow overlapping region. Despite this, ePDE proves to be more robust than other advanced feature detectors when stitching near-uniform images. Feature detectors like SIFT, AGAST, KAZE, AKAZE, and U-RIFT typically reach a success rate approximately $\pm 95\%$. In contrast, the remaining feature detectors fall below 87% success rate, with MSER performing the worst at 64.2%. Most poorly stitched images often arise from a lack of sufficient and densely clustered inliers within a limited area of the overlapping region.

Due to the low performance of MSER, a total of 60 pairs of sample images that demonstrate successful image stitching using all feature detectors are brought together for the subsequent analyses. This includes 25 pairs of benchmark images and 35 pairs from a total of 70 pairs of general images. The analyses utilize the evaluation metrics, including the number of inliers, N_m , the spread-overlap, S_o , the recall, RC , and the RC/S_o scores. The

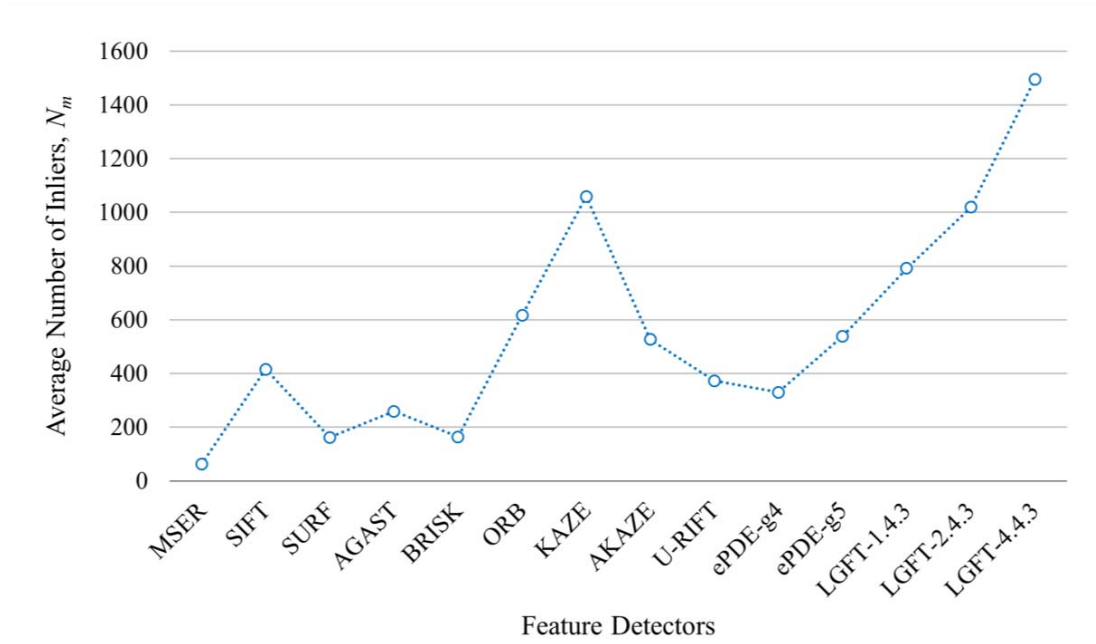


Figure 4.13: The average number of inliers, N_m varies among different feature detectors for 60 sample image pairs

evaluation response for each feature detector is expressed as the average values derived from 60 sample image pairs. For easier comparison, the following analyses use figures that display one evaluation metric alongside the number of inliers, shown on the left and right vertical axes respectively, for all feature detectors.

Figure 4.13 demonstrates that the LGFT-4.4.3 variant detects significantly more inliers than other feature detectors. This result is expected because the LGFT-4.4.3 generally generates more scale-spaces compared to the LGFT-1.4.3 and LGFT-2.4.3 variants. It also employs additional oriented Log-Gabor scale-spaces used for interest point detection through non-maxima suppression. KAZE achieves results comparable to LGFT-2.4.3 due to its robustness in feature-rich images that helps it detect more inliers, but it performs less effectively in near-uniform scene images, as evidenced in Figure 4.12. The ePDE- g_5

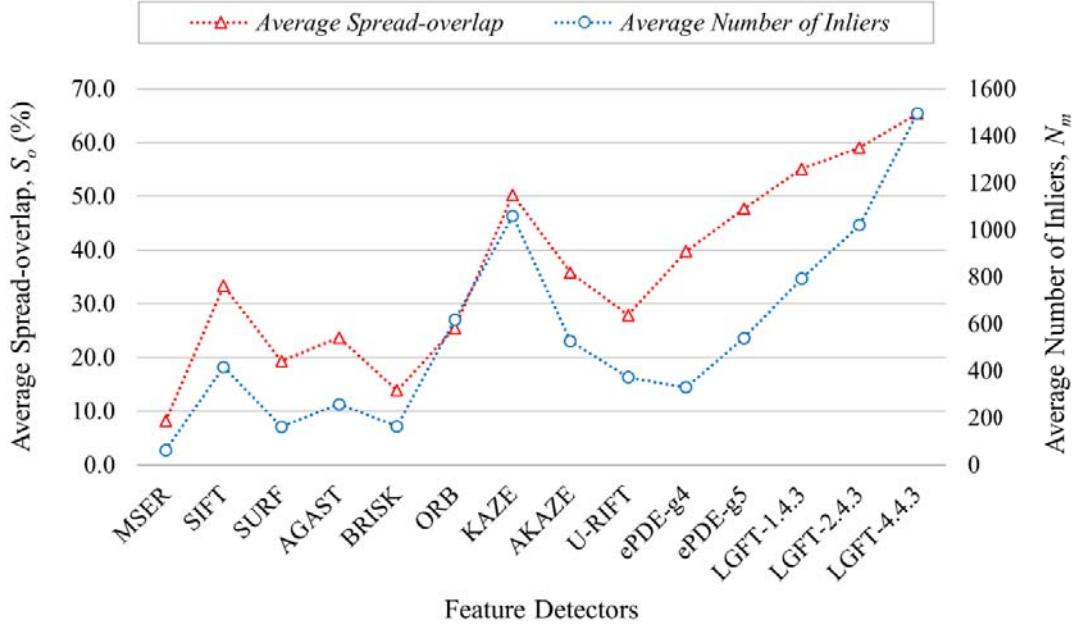


Figure 4.14: The average spread-overlap, S_o and average number of inliers, N_m varies among different feature detectors for 60 sample image pairs

performs better than ePDE- g_4 in terms of number of detected inliers, due to the ability of g_5 to preserve most of the prominent structures in its generated scale-spaces at higher evolution time units.

Though a higher number of inliers, N_m is important, a narrower spread of these inliers within a small overlapping area can lead to poor image stitching and significant distortion in the final product. Figure 4.14 illustrates the ability of each feature detector to produce substantial number of well-distributed inliers within the overlapping area. The trends in the responses for inliers, N_m and spread-overlap, S_o are quite alike. Among them, the proposed LGFT methods achieve the most extensive spatial distribution of inliers, covering over 50% of the overlapping area, surpassing other feature detectors. The spread-overlap, S_o responses of both ePDE- g_4 and ePDE- g_5 fall within a satisfactory range of 40%

to 48%, which is higher than most feature detectors. KAZE also shows strong results regarding the number of inliers, N_m and their spatial distribution, S_o , but it does not perform as well in stitching near-uniform images compared to LGFT and ePDE. In this experiment, MSER records the lowest values for both the number of inliers, N_m , and spread-overlap, S_o when compared to all other feature detectors, reflecting the shortcomings of the local region detection approach, especially in near-uniform scenes.

Recall is calculated as the ratio of inliers to the total number of initial correspondences within the overlapping regions of the scene images. The recall values, RC presented in Figure 4.15 show that all feature detectors retain more than 50% of their initially identified correspondences as stable inliers. Although the proposed LGFT methods achieve the lowest recall, RC rate among all detectors, it still preserves a high number of inliers

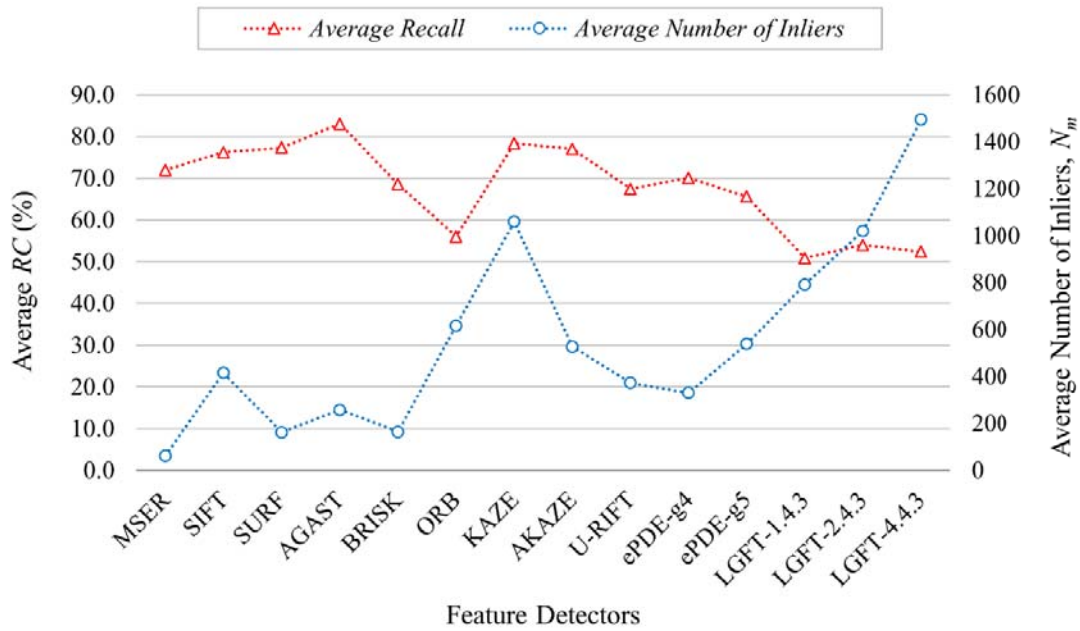


Figure 4.15: The average recall, RC and average number of inliers, N_m varies among different feature detectors for 60 sample image pairs

(meaning half of its initial matches are valid inliers) sufficiently for robust stitching performance. This suggests that LGFT's lower recall, RC is not necessarily a drawback, as successful image stitching depends more on the quantity and spatial distribution of inliers than on recall alone. On the other hand, MSER appears to hold 72% of its lowest number of corresponding interest points as stable inliers, primarily originating from the feature-rich regions of the images.

Figure 4.16 shows the results of RC/S_o score. The proposed LGFT methods clearly outperform other feature detectors, achieving the lowest RC/S_o scores, close to one (ranging between 0.8 and 0.9). This suggests that LGFT methods are highly effective in identifying a sufficient number of inliers with extensive spatial distribution, a crucial factor for accurately estimating global homography transformation models between images. In contrast, higher

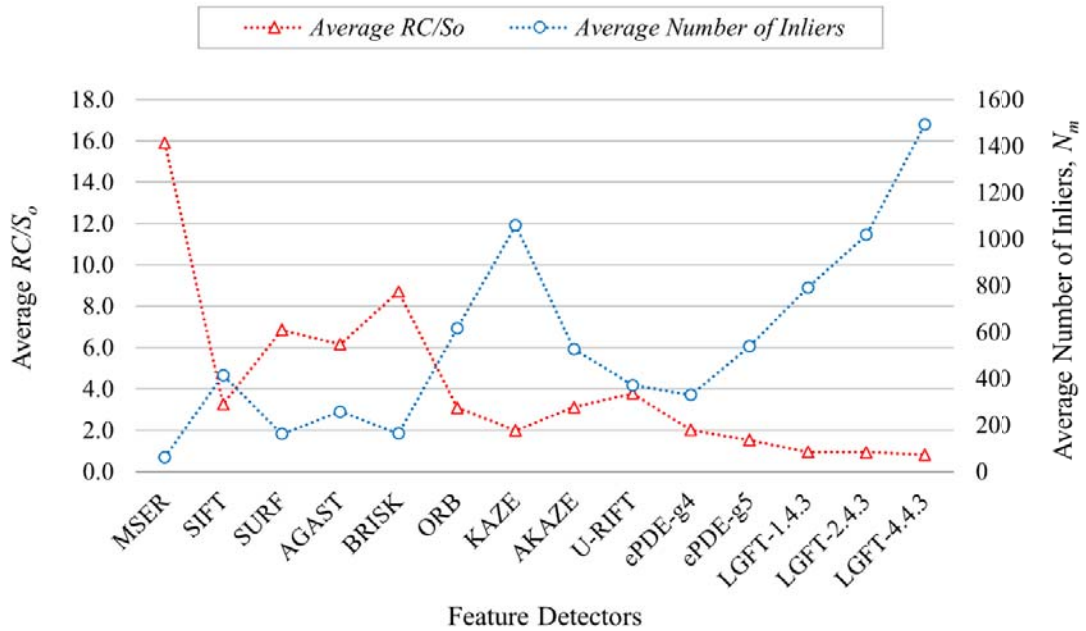


Figure 4.16: The average RC/S_o scores and average number of inliers, N_m varies among different feature detectors for 60 sample image pairs

RC/S_o scores are more likely to result in noticeable image distortion and, in some cases, complete failure of the image stitching process. After LGFT, the ePDE method ranks second with RC/S_o scores also close to one. The RC/S_o scores for both ePDE- g_4 and ePDE- g_5 are slightly higher, ranging between 1.5 and 2.0. These scores are still considered marginally good, indicating that they provide sufficiently widespread inliers to ensure precise alignment in image stitching, reducing misalignment errors and geometric distortion.

MSER performs the worst, with the highest RC/S_o score of 15.9, as shown in Figure 4.16, accounting for its poor success rate. The number of extracted interest points is significantly lower than those of the proposed methods, and their spatial distribution is insufficient for effective image stitching, particularly in near-uniform scenes (see Figure 4.14). After MSER, both the methods SURF and BRISK also show high RC/S_o scores of 6.9 and 8.7, respectively, a good indicator of their image stitching success rate falls below 90%. Although ORB has a lower RC/S_o score of 3.1, its success rate for image stitching also falls below 90%, implying that the detected inliers are less precise in approximating good image alignment, especially in near-uniform scenes.

To further confirm the superior performance of the proposed ePDE and LGFT methods, Figure 4.17 presents their success rate in image stitching in comparison to other advanced feature detectors across all 123 pairs of HiRISE images. These images, typically contain near-uniform regions, often pose challenges for other feature detectors, leading to poor outcomes or failures. Similar to Figure 4.12, the image stitching success rate for each feature detector in the HiRISE dataset is determined by visually inspecting each stitched pair to ensure the final image is free of severe distortion. Images that fail to stitch or have significant distortion that cannot be repaired using warping, blending, and seam-cutting

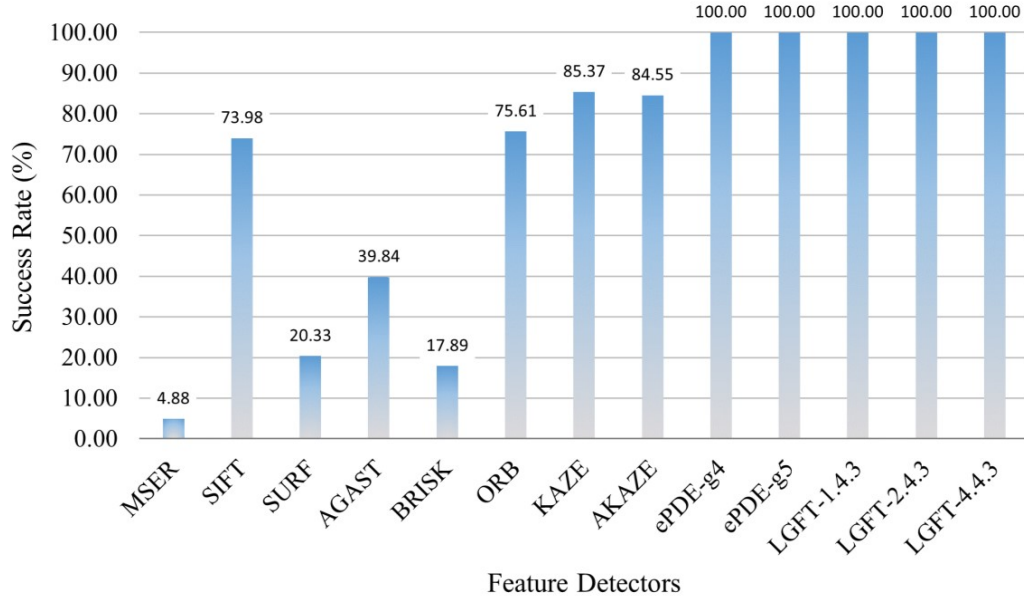


Figure 4.17: Success rate of different feature-based image stitching across 123 pairs of HiRISE images

methods, are excluded from the count. The success rate is then calculated as the percentage of accurately stitched images from the 123 pairs of HiRISE images. Figure 4.17 shows that both the ePDE and LGFT methods excel, achieving a 100% success rate and surpassing other feature detectors. This demonstrates their effectiveness and robustness in stitching near-uniform scenes by producing a sufficient and well-distributed number of inliers within the overlapping area. It should be noted that the upright version of RIFT (U-RIFT) is excluded from Figure 4.17 due to its inability to detect interest points in some rotated HiRISE images.

In summary, these quantitative evaluations clearly indicate that the LGFT and ePDE methods are generally more robust than other feature detectors tested for stitching near-uniform images, as they generate sufficient well-distributed inliers within the overlapping area. A high number of detected inliers, such as those by KAZE in Figure 4.13, does not

guarantee its capability to achieve a better success rate in image stitching, especially for near-uniform scenes. Instead, having a stable and adequately distributed set of inliers across the overlapping region is crucial for achieving satisfactory stitched images. Poorly stitched images usually result from an insufficient and clustered inliers within a small overlapping area.

Furthermore, the recall, RC metric alone does not fully reflect whether the extracted inliers from feature detectors are robust enough for successful image stitching, as it only measures the proportion of correct matches. In contrast, the RC/S_o score serves as a more reliable metric by integrating recall, RC with spread-overlap, S_o , thereby capturing both the accuracy and spatial distribution of inliers. Experimental results presented in this section show that RC/S_o score approaching one consistently correspond to more accurate and seamless stitched images with higher stitching success rate, confirming the metric's effectiveness in evaluating the robustness of feature detectors for successful image stitching.

4.5.2 Qualitative Comparison

This section evaluates the performance of the proposed ePDE and LGFT methods by visually inspecting the resulting stitched images. This evaluation highlights the strengths and weaknesses of each feature detector in terms of image stitching quality, especially for near-uniform scenes. The quality of image stitching is primarily assessed by visually inspecting the approximate positions of inliers and their distribution within the overlapping area detected by the feature detectors. The dataset used for these assessments includes benchmark images with photometric transformations, general images with partially near-uniform

content, and HiRISE images with predominantly near-uniform scenes. The visual assessments are organized into three different approaches:

- i. **Inlier Distribution Comparison:** Visually compare the number and spatial distribution of inliers produced by various feature detectors, including the proposed ePDE and LGFT methods, on photometrically transformed benchmark images.
- ii. **Stitching Quality Comparison:** Visually evaluate the final stitched images from each feature detector, with emphasis on comparison against the proposed ePDE- g_5 and LGFT-1.4.3.
- iii. **Focused Method Comparison:** Specifically assess the stitched image quality of KAZE and AKAZE in contrast to the proposed ePDE- g_5 and LGFT-1.4.3 using selected examples.

In the **Inlier Distribution Comparison**, the performance of various feature detectors, including the proposed ePDE and LGFT methods, is evaluated on photometrically transformed benchmark images. These images are generally less challenging to stitch than near-uniform scenes, due to their nonuniformity and feature-rich content. However, when images undergo photometric transformations that introduce blurriness, dimness, lossiness, or noise, the feature detectors vary in their ability to identify a sufficient number of well-distributed inliers for successful and accurate image stitching. Figure 4.18 illustrates how various detectors perform on the blurry version of the 'bikes' image. Figure 4.19 shows the results on the light-adjusted 'leuven' image. In both figures, the approximate positions of detected inliers are marked by white circles and cross symbols.

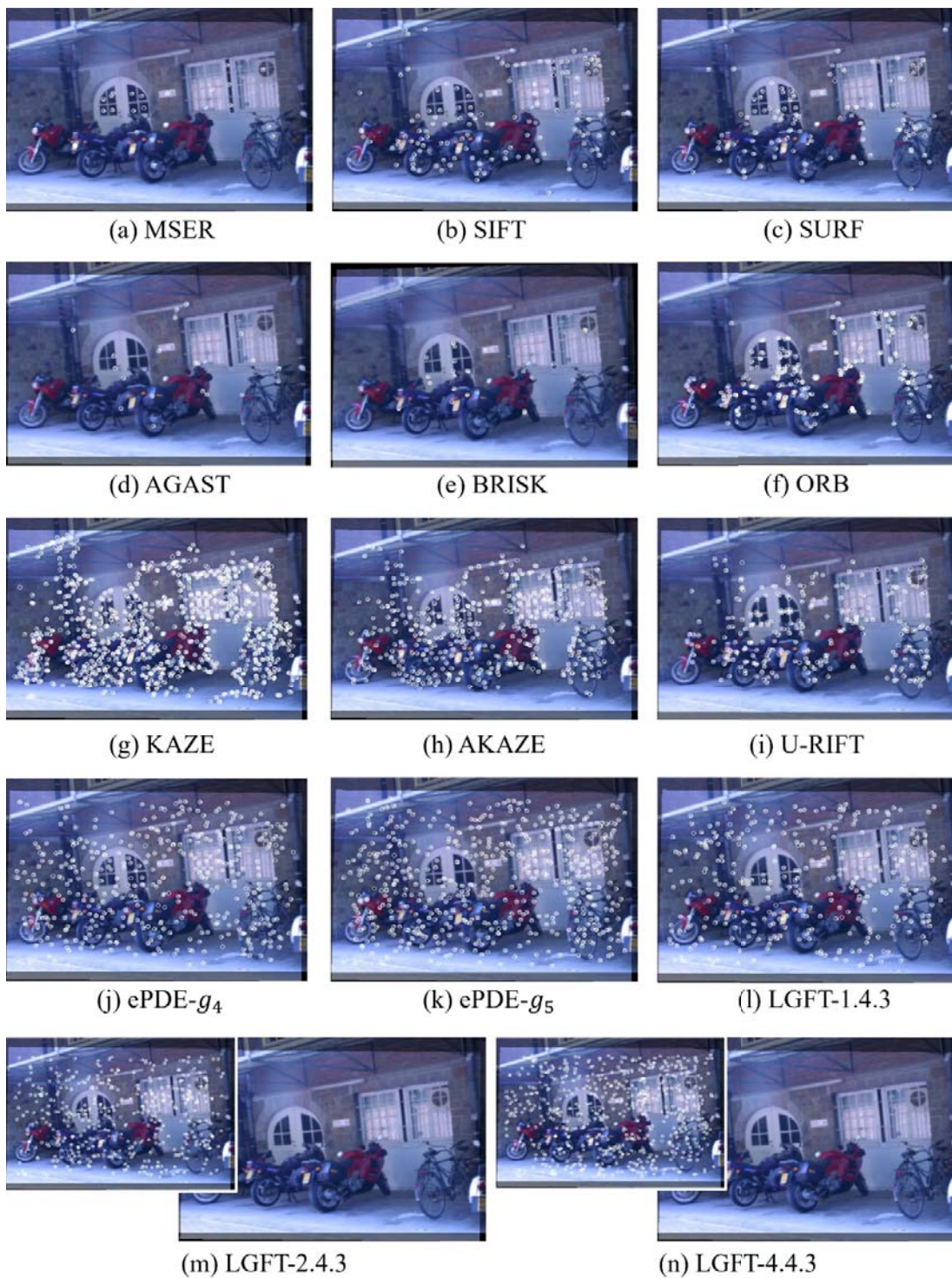


Figure 4.18: Comparison of inlier count and distribution in the stitched images of blurry 'bikes' using different feature detectors. Inliers are marked by white circles and crosses. (m) & (n) include stitched images without markers for clarity



Figure 4.19: Comparison of inlier count and distribution in the stitched images of light-changed 'leuven' using different feature detectors. Inliers are marked by white circles and crosses. (m) & (n) include stitched images without markers for clarity

Although all feature detectors can detect inliers and produce successful image stitching, there are significant differences in the number of inliers detected and their distribution patterns. As shown in Figure 4.18 and Figure 4.19, MSER, SIFT, SURF, AGAST, BRISK, and ORB detect relatively fewer inliers, whereas KAZE, AKAZE, and U-RIFT demonstrate moderate performance. In contrast, both ePDE and LGFT detect a higher number of inliers that are more widely distributed across the overlapping region in comparison to other feature detectors. A closer inspection of Figure 4.18 and Figure 4.19 reveals that ePDE and LGFT can detect inliers in blurrier, dimmer, and smoother areas that are difficult for other detectors to identify. This suggests that ePDE and LGFT are highly effective in handling feature-rich scenes under various photometric transformations, outperforming many existing feature detectors in image stitching.

In the **Stitching Quality Comparison**, a visual assessment is conducted using nine sets of final stitched images, shown in Figure 4.20 to Figure 4.28. Each figure displays the stitching results of different feature detectors, along with the estimated positions of inliers in the overlapping regions. The seams are intentionally retained to reveal how the images are stitched together. Each set contains three stitched images: the first image is created by a tested feature detector, the second image by ePDE- g_5 , and the third image by LGFT-1.4.3. Since these images partially contain near-uniform regions, the stitching process is particularly challenging and more prone to image distortion, misalignment and visible artifacts. Only the ePDE- g_5 and LGFT-1.4.3 variants are included in these examples, as they generally demonstrate superior robustness and efficiency compared to other versions.

The resulting stitched image tends to exhibit notable distortion when inliers are fewer and clustered in a small area. This issue is evident in Figure 4.20, where the stitched output

produced by MSER shows far more severe distortions compared to those generated by ePDE- g_5 and LGFT-1.4.3. The primary cause is the limited number of inliers clustered in a small area in the MSER-stitched image, which leads to inaccuracies in estimating the global geometric transformation, ultimately resulting in obvious distorted stitching errors.

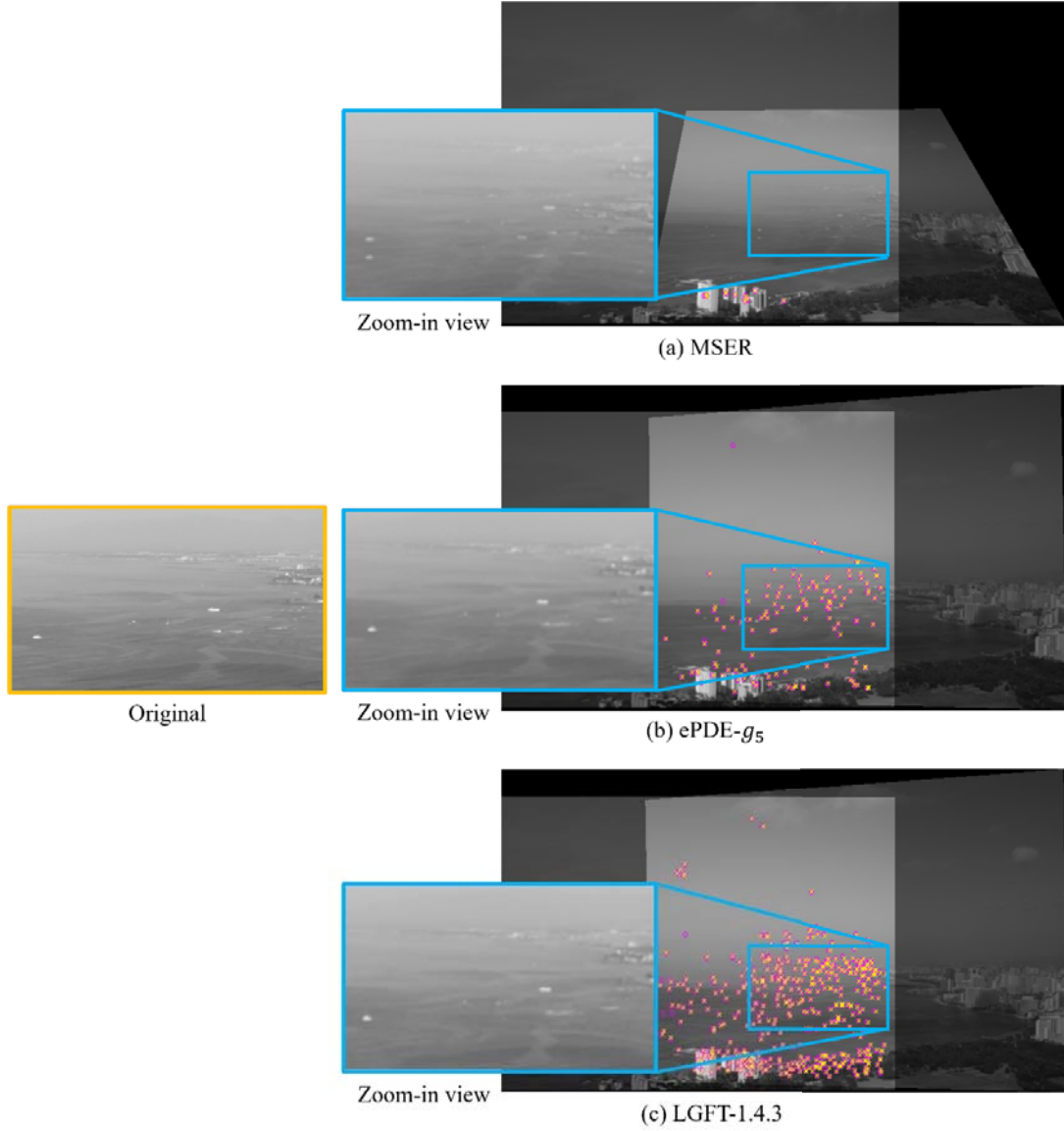


Figure 4.20: Comparison of resulting stitched image produced by the MSER, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers

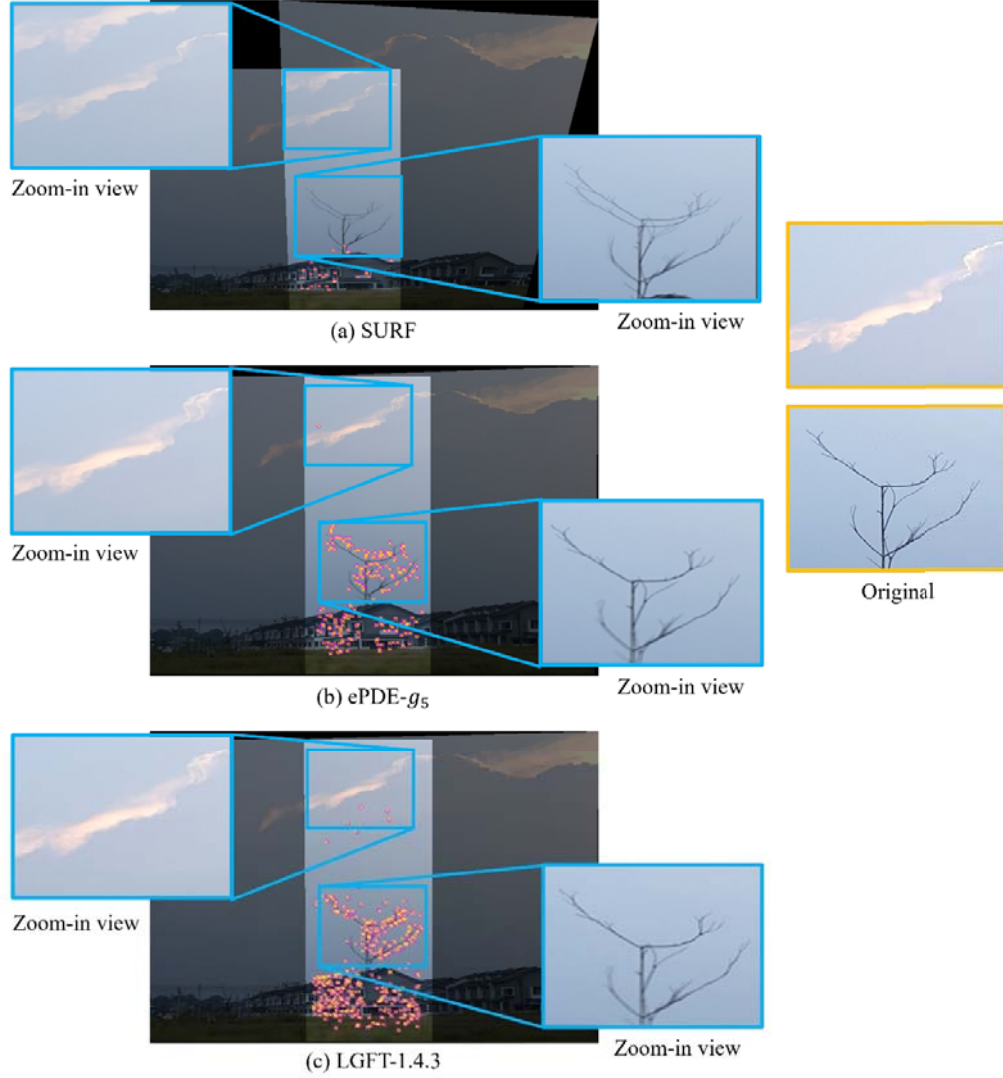


Figure 4.21: Comparison of resulting stitched image produced by the SURF, $ePDE-g_5$, and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers

A similar stitching distortion is observed in the SURF-stitched image shown in Figure 4.21(a). The obvious misalignment in the cloud and tree regions results from a lower number of inliers concentrated in feature-rich house area, leaving the near-uniform cloud region undetected. Conversely, LGFT-1.4.3 and $ePDE-g_5$ achieve generally better stitching results compared to SURF, though a slight misalignment remains in the cloud region. This

minor error is attributed to the higher number of inliers concentrated in the house and tree areas. Inliers detected by LGFT-1.4.3 and ePDE- g_5 improve the accuracy of global geometric transformation estimation, thereby minimizing overall stitching artifacts.

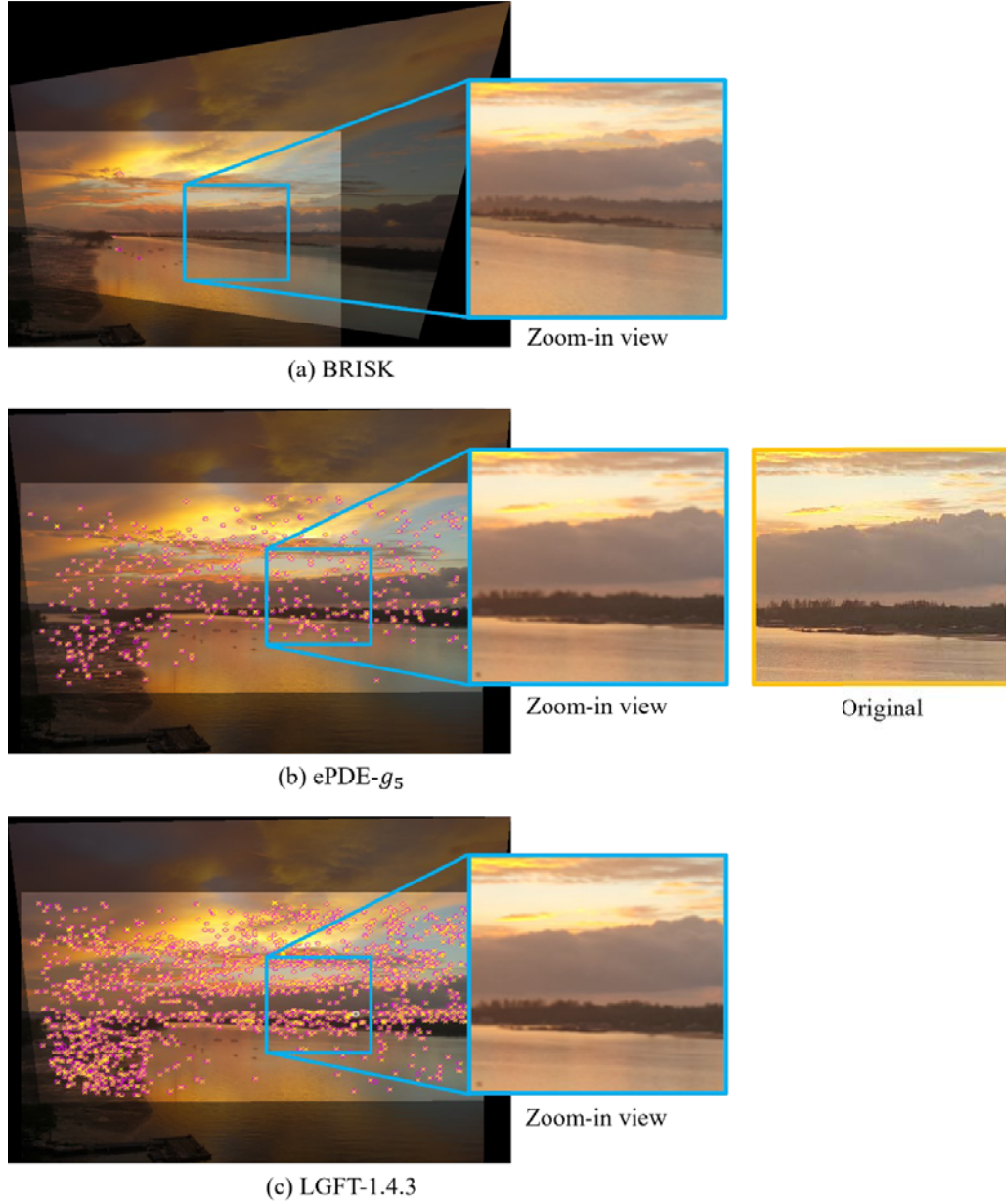


Figure 4.22: Comparison of resulting stitched image produced by the BRISK, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers

Figure 4.22 presents an example of distortion in the BRISK-stitched image, primarily caused by an insufficient number of inliers. Instead, the stitched outputs from ePDE- g_5 and LGFT-1.4.3 show better alignment due to a significantly higher number of well-distributed inliers. Figure 4.23 further shows the challenge of a near-uniform coastal scene, where KAZE struggles to detect enough inliers compared to ePDE- g_5 and LGFT-1.4.3.

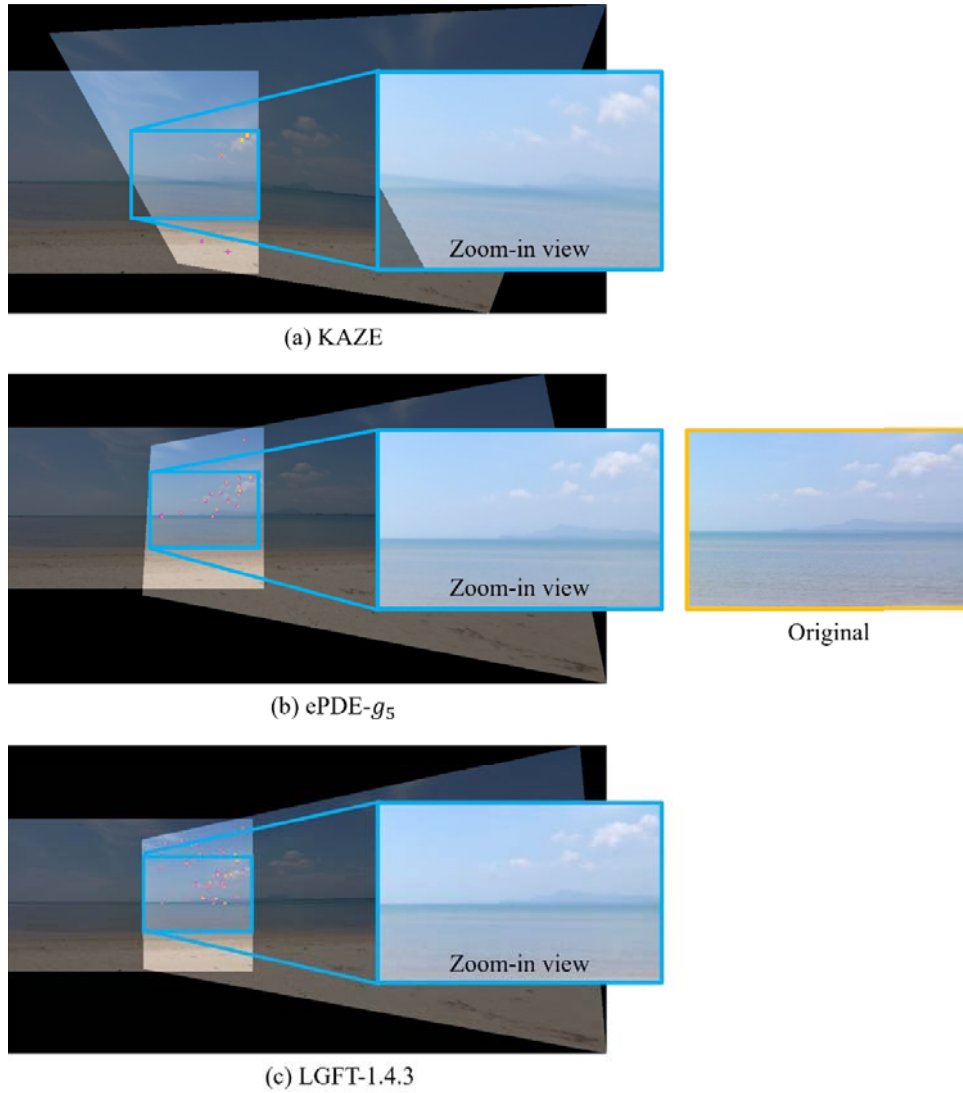


Figure 4.23: Comparison of resulting stitched image produced by the KAZE, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers

SIFT is not designed to address near-uniform scene (Lowe, 1999, 2004). In Figure 4.24, it fails to produce accurate stitching, resulting in visible misalignment and distortion. AKAZE-stitched image in Figure 4.25 shows similar misalignment. These stitching errors occur because both SIFT and AKAZE struggle to detect enough inliers in the overlapping near-uniform regions. Though, the outputs from ePDE- g_5 and LGFT-1.4.3 achieve accurate alignment, supported by a greater number of well-distributed inliers.

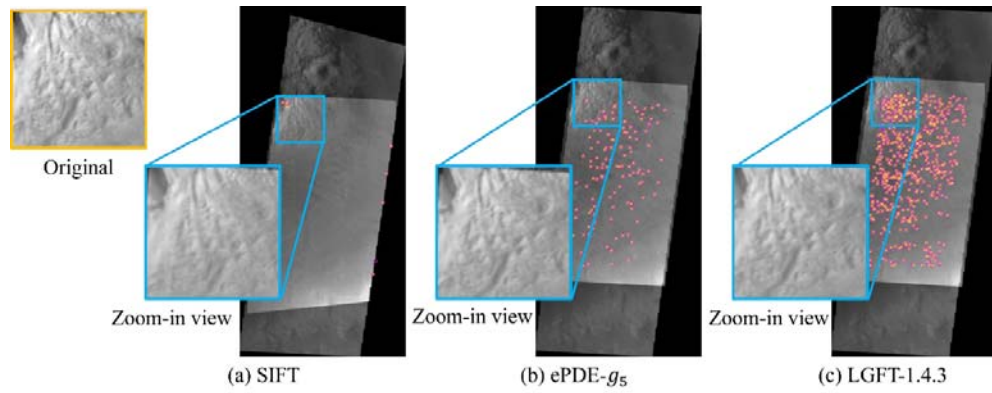


Figure 4.24: Comparison of resulting stitched image produced by the SIFT, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers

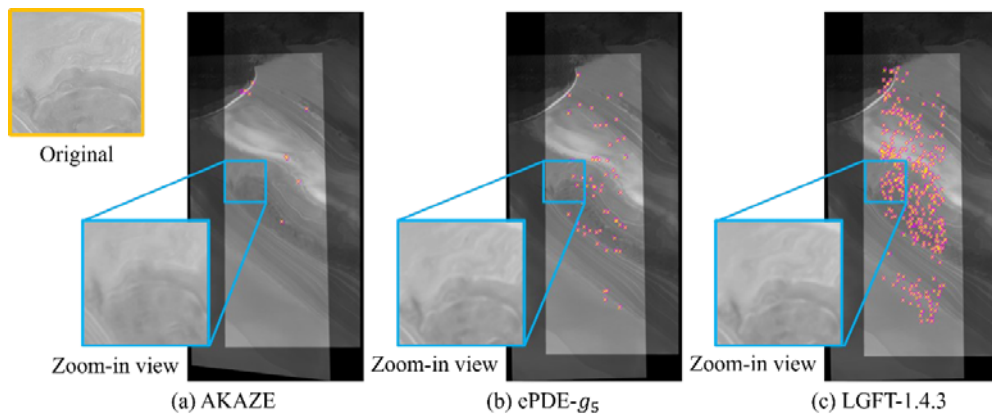


Figure 4.25: Comparison of resulting stitched image produced by the AKAZE, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers

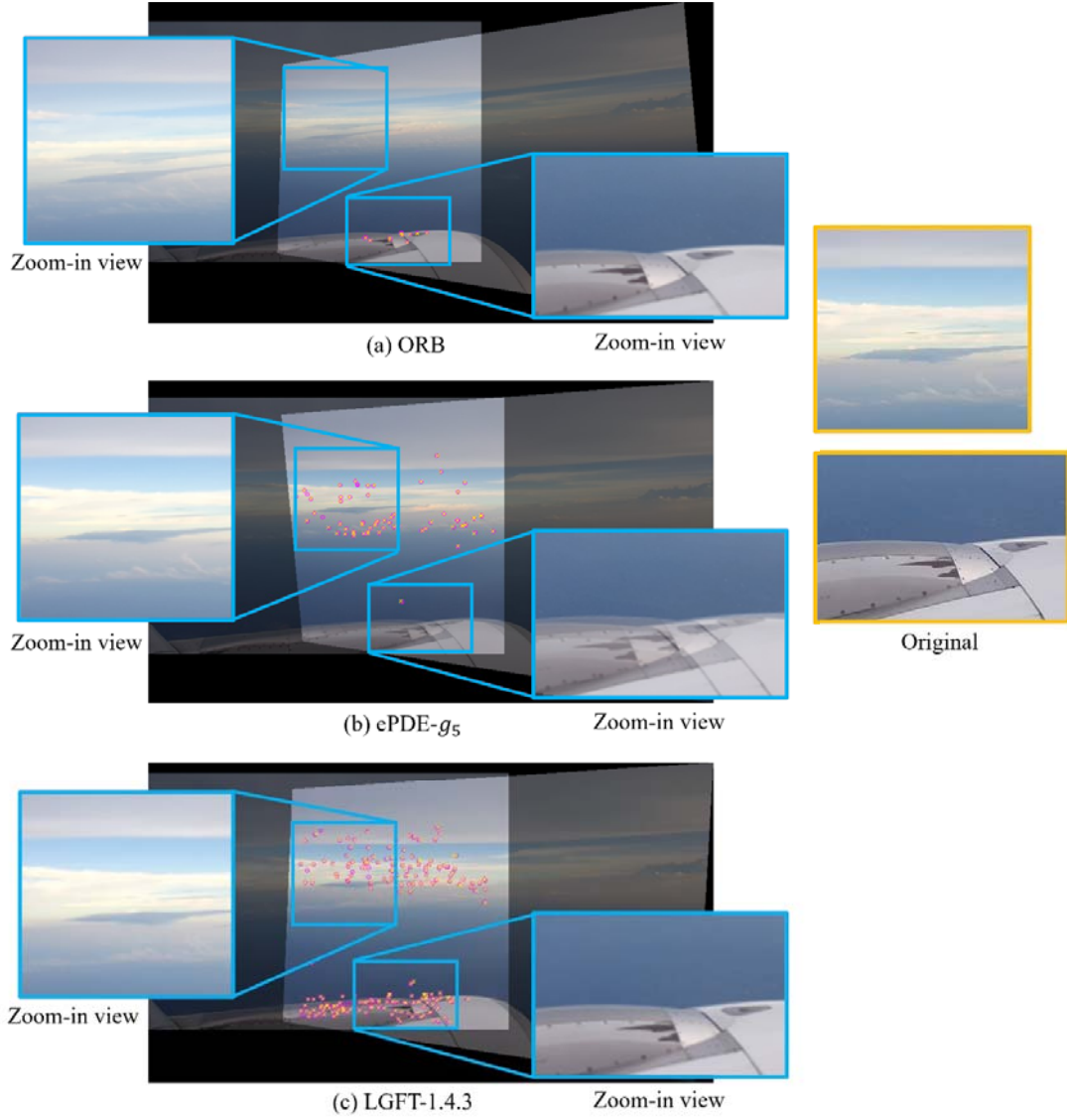


Figure 4.26: Comparison of resulting stitched image produced by the ORB, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers

Figure 4.26 shows significant distortion in the ORB-stitched image, mainly due to a very limited number of inliers clustered within a small area, leading to inaccurate estimation of global transformation model. The ePDE- g_5 -stitched image also shows minor misalignment, as its inliers are concentrated in the upper cloud region of the overlap. By

comparison, LGFT-1.4.3 demonstrates greater robustness by detecting sufficient inliers across both the upper cloud region and the lower aircraft engine area, enabling more accurate stitching. Additionally, the U-RIFT-stitched image in Figure 4.27 also exhibits misalignment when compared to the outputs of the proposed ePDE- g_5 and LGFT-1.4.3.

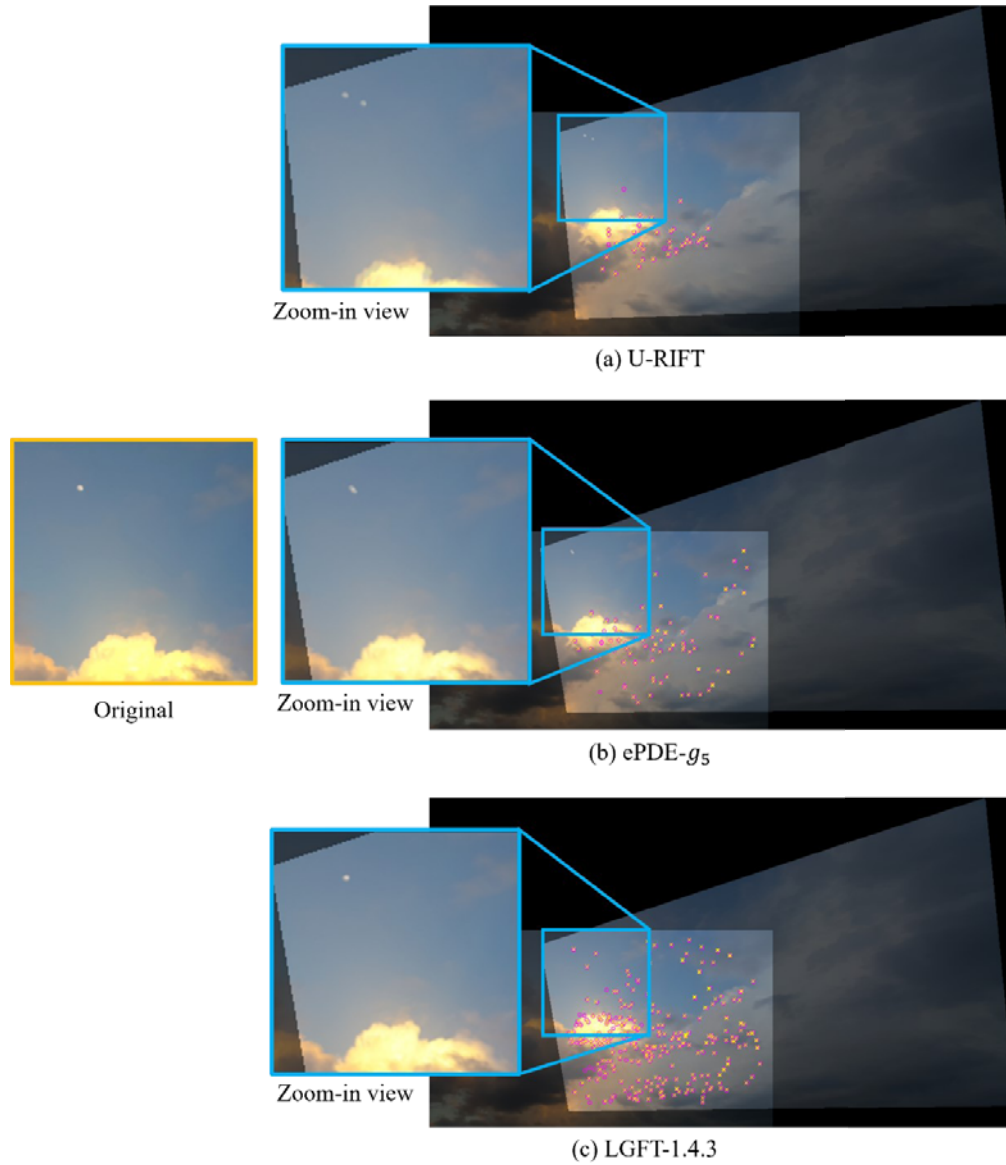


Figure 4.27: Comparison of resulting stitched image produced by the U-RIFT, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers

Figure 4.28 shows less obvious misalignment in the AGAST-stitched image. But a noticeable blurring effect appears in the green field, particularly in the upper-left overlapping area. This is due to most inliers are concentrated along the edges of the elliptical shape, unlike the more evenly distributed inliers observed in the outputs of the proposed methods.

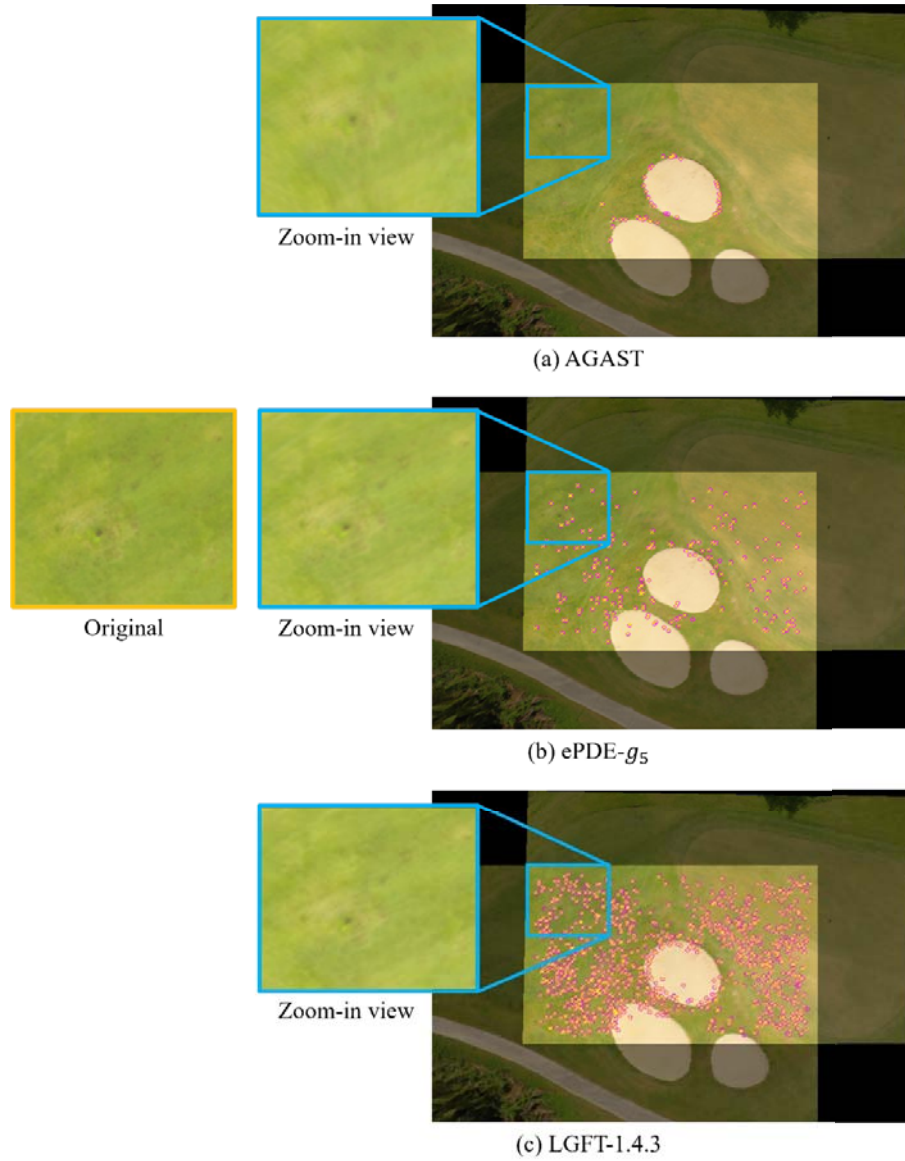


Figure 4.28: Comparison of resulting stitched image produced by the AGAST, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by magenta circles and yellow cross markers

In the **Focused Method Comparison**, attention is given to a direct comparison between KAZE, AKAZE, ePDE- g_5 and LGFT-1.4.3. This comparison is essential, as KAZE and AKAZE generally achieve higher average stitching success rates than most other methods, excluding ePDE and LGFT. While KAZE and AKAZE excel in feature-rich areas, their performance diminishes in near-uniform regions. To illustrate this, the following Figure 4.29 to Figure 4.32 present additional stitched image examples using KAZE, AKAZE, ePDE- g_5 , and LGFT-1.4.3. These comparisons highlight the strengths and weaknesses of each method, showing how ePDE- g_5 and LGFT-1.4.3 maintain superior performance in challenging near-uniform scenes, where KAZE and AKAZE may falter.

KAZE and AKAZE generally perform well in detecting inliers within feature-rich regions but are less effective in near-uniform scenes. This limitation is evident in Figure 4.29 to Figure 4.31, where both methods fail to identify sufficient inliers in near-uniform areas. This observation is consistent with previous works, where KAZE and AKAZE are shown to perform well in textured environments but did not address the challenges of near-uniform scene (Alcantarilla et al., 2013; Alcantarilla et al., 2012). In addition, Figure 4.29 to Figure 4.31 also show that KAZE detects slightly more inliers than AKAZE that prioritizes faster detection speed (Alcantarilla et al., 2013).

In Figure 4.29, the images stitched using KAZE and AKAZE exhibit misalignment in the cloud region. Both methods fail to detect inliers in the ocean and cloud regions that are largely featureless or near-uniform. Additionally, the number of inliers detected in the town building area is insufficient, resulting in poor estimation of global transformation for accurate stitching. Instead, both ePDE- g_5 and LGFT-1.4.3 successfully detect enough well-distributed inliers across the overlapping region, leading to distortion-free image stitching.

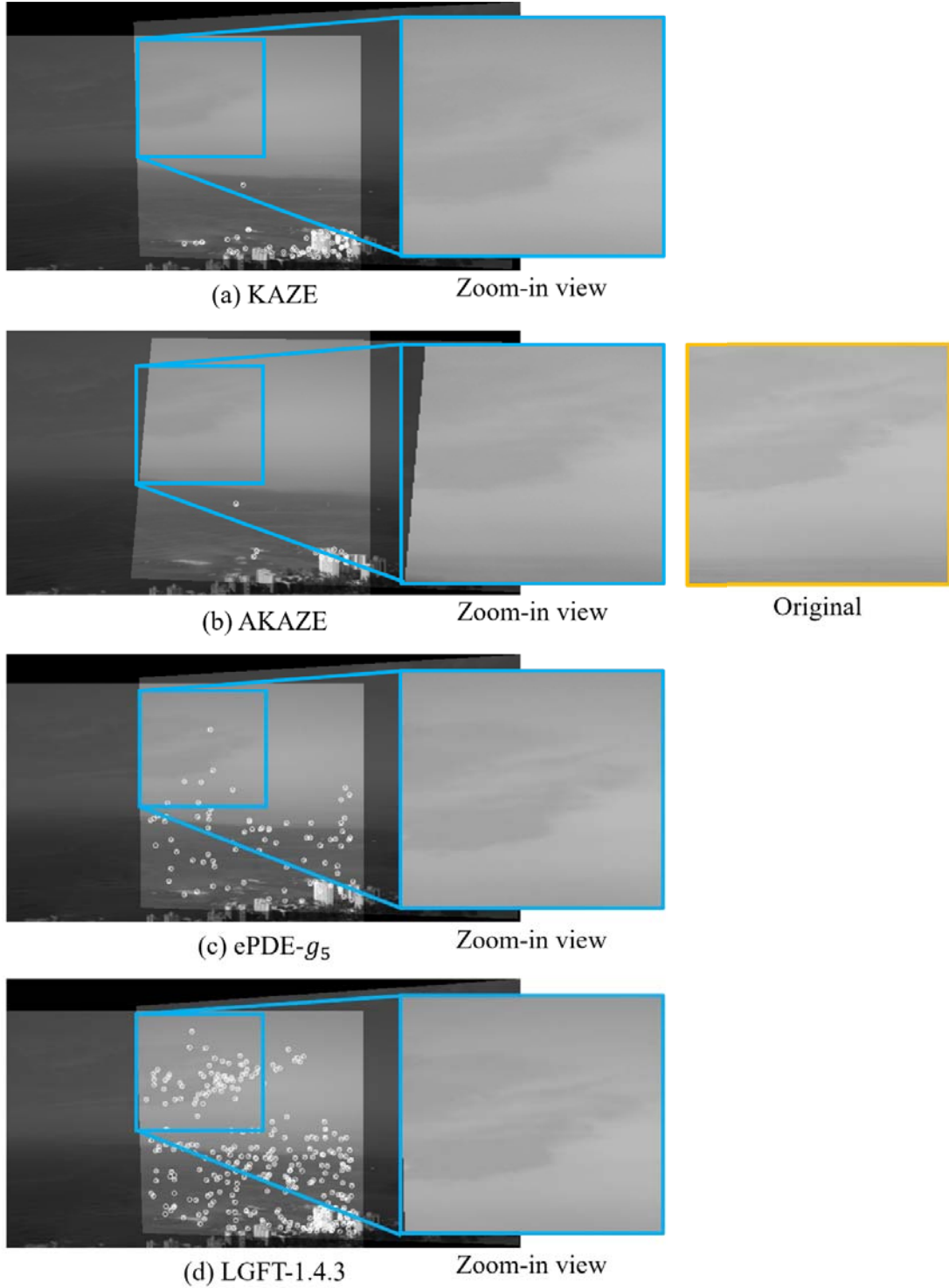


Figure 4.29: Comparison 1 of resulting stitched images produced by the KAZE, AKAZE, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by white circles and cross markers

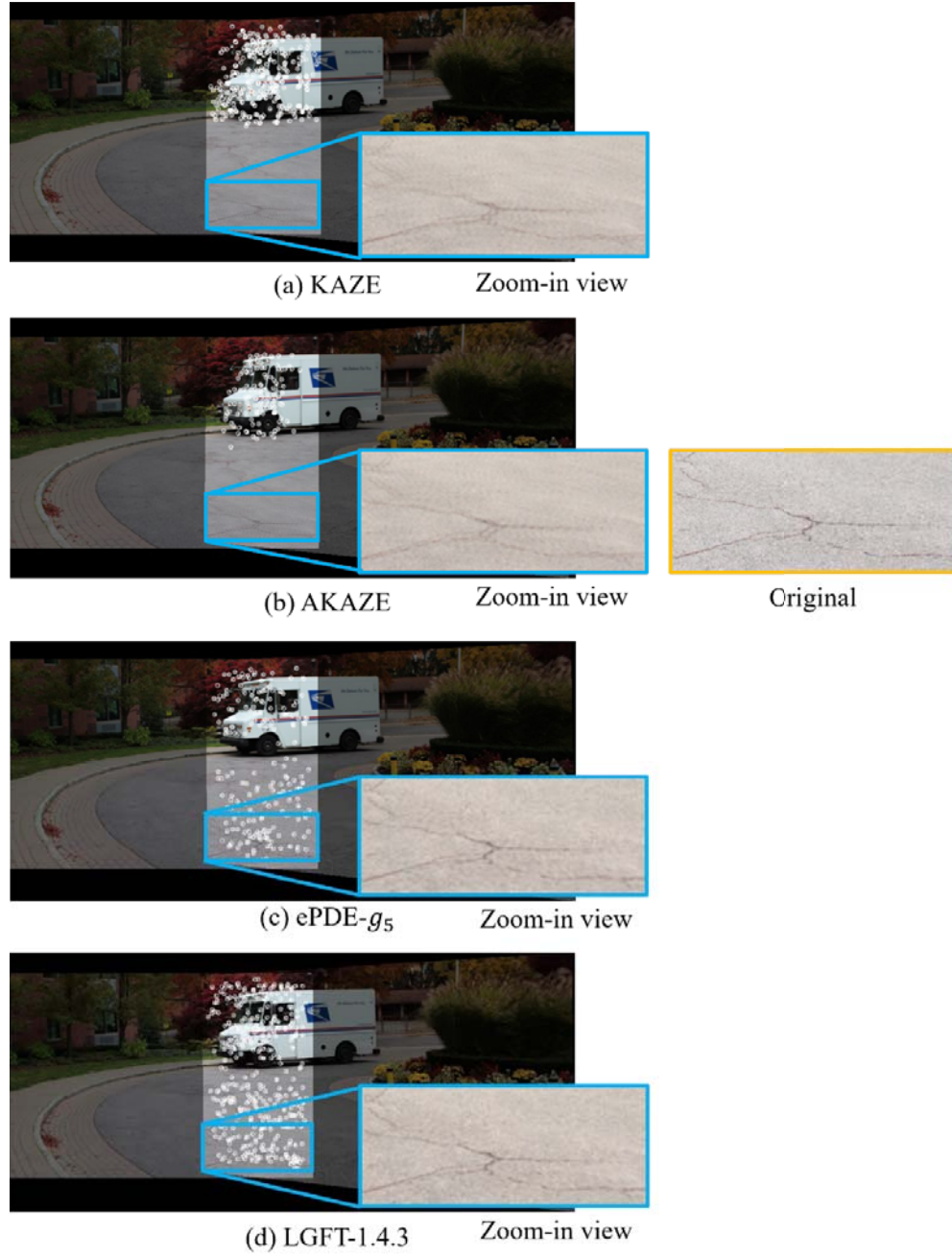


Figure 4.30: Comparison 2 of resulting stitched images produced by the KAZE, AKAZE, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by white circles and cross markers

Figures 4.30 and 4.31 show similar weaknesses in the KAZE- and AKAZE-stitched images, particularly noticeable misalignment compared to those from ePDE- g_5 and

LGFT-1.4.3. While KAZE and AKAZE detect a sufficient number of inliers, they are mostly concentrated in feature-rich regions, with none detected in near-uniform areas. In contrast, $\text{ePDE-}g_5$ and LGFT-1.4.3 produce more inliers that are well-distributed across the entire overlap, resulting in more accurate homography model for precise stitching alignment.

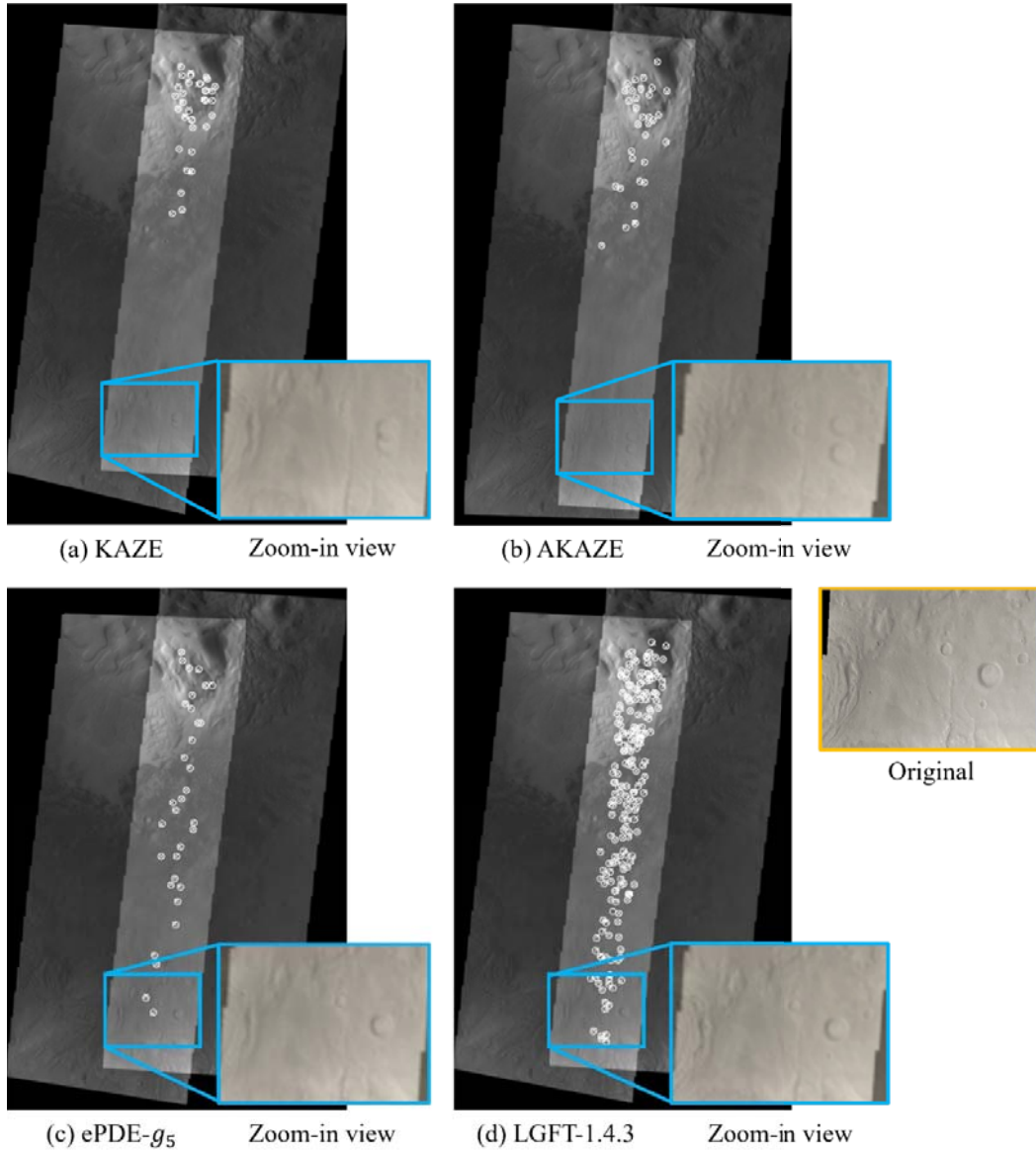


Figure 4.31: Comparison 3 of resulting stitched images produced by the KAZE, AKAZE, $\text{ePDE-}g_5$, and LGFT-1.4.3 methods. The approximate inlier locations are marked by white circles and cross markers

Interestingly, in Figure 4.32, the stitched images produced by KAZE and AKAZE do not show clear distortion or misalignment, despite what seems to be an insufficient number of inliers. However, the distribution of these inliers suggests that they are adequate for accurate global geometric transformation model in this particular case. On the other hand, ePDE- g_5 and LGFT-1.4.3 achieve a greater number of inliers with wider spatial coverage, providing stronger confidence in the accuracy of the stitching results.

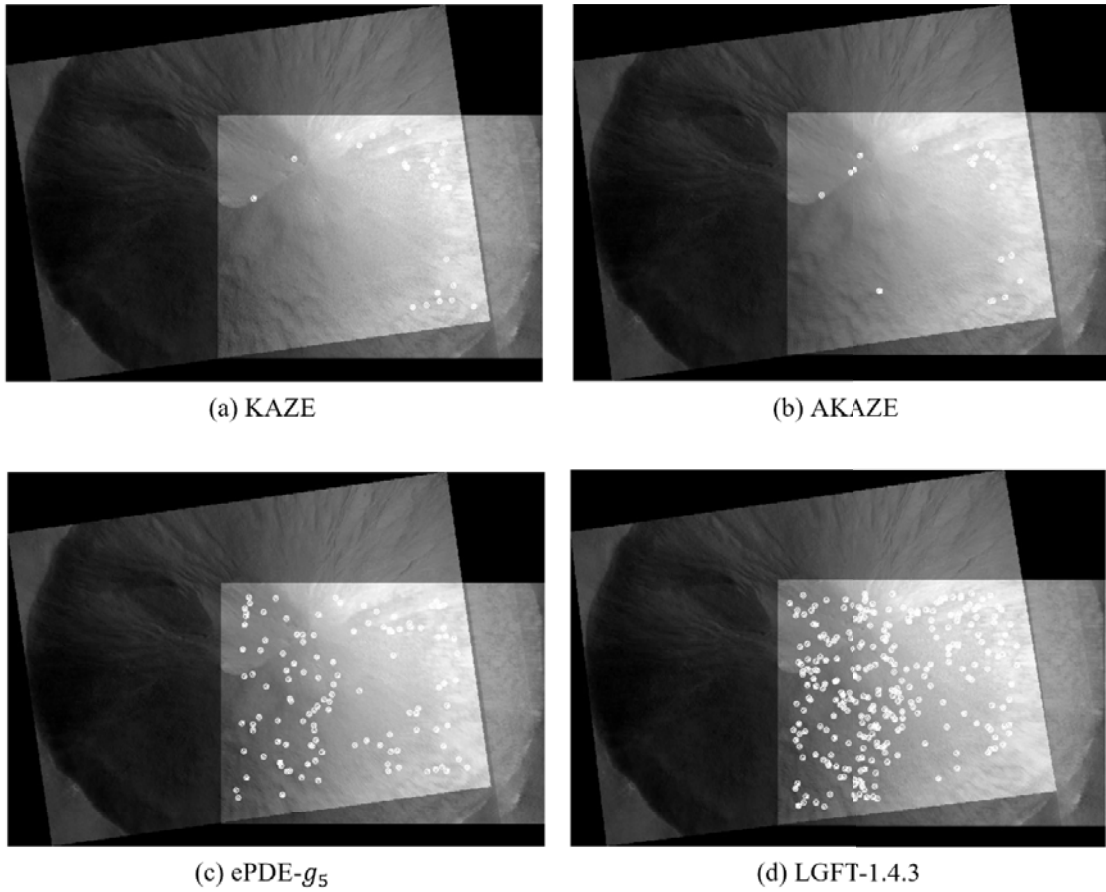


Figure 4.32: Comparison 4 of resulting stitched images produced by the KAZE, AKAZE, ePDE- g_5 , and LGFT-1.4.3 methods. The approximate inlier locations are marked by white circles and cross markers

Figures 4.29 to 4.32 also reveal that LGFT-1.4.3 consistently detects more inliers than ePDE- g_5 , highlighting its greater effectiveness in feature detection. This advantage arises mainly from the texture enhancement capabilities of LGFT that generates granule-like patterns in its scale-spaces. These patterns provide richer structural details that enhance feature detection. By comparison, the ePDE approach emphasizes preserving object boundaries while smoothing homogeneous regions through nonlinear diffusion (Alcantarilla et al., 2012). Due to differences in texture detail within their scale-spaces, ePDE detects slightly fewer inliers compared to LGFT. The increased inlier detection by LGFT-1.4.3 contributes to more precise global geometric transformation model, consistently enhancing the accuracy of the stitched images across both feature-rich and near-uniform scenes.

Based on the visual assessment approaches described above, the proposed ePDE- g_5 and LGFT-1.4.3 techniques demonstrate enhanced performance in image stitching, particularly for near-uniform scene images. This is supported by the examples provided in this section, which evident that existing state-of-the-art methods struggle to detect a sufficient number of inliers in challenging low-texture or near-uniform regions (Lyu et al., 2019; Pandey & Pati, 2019; Yang et al., 2025). In comparison, the proposed ePDE- g_5 and LGFT-1.4.3 methods successfully identify a greater number of inliers, widely spread across both feature-rich and near-uniform regions. This broadly distributed inliers provide essential information required for accurately estimating the global geometric transformation model. Consequently, this enables precise image alignment with minimal or no distortion and visible artifacts between overlapping images.

4.5.3 Execution Efficiency

This section evaluates the efficiency of the proposed ePDE and LGFT methods by comparing their processing time with those of other feature detectors. While faster processing time are particularly advantageous for real-time applications, they are not the strongest aspect of the proposed ePDE and LGFT methods in this study. Table 4.3 highlights the fastest processing time recorded by each feature detection method. To ensure a fair comparison, processing time for all feature detectors are averaged over the same set of 20 image pairs. This set includes 12 benchmark and 8 general image pairs. Obviously, ORB stands out with the fastest processing time, outperforming the other feature detectors. This efficiency is largely due to ORB's design that prioritizes computational speed and benefits from the binary nature of its BRIEF descriptors, resulting in lower memory usage.

Table 4.3: Comparison of the average processing time for each feature detector, calculated over 20 images and measured in seconds

Rank	Feature Detector	Average Processing Time (s)
1	ORB	0.022
2	AKAZE	0.047
3	SURF	0.053
4	AGAST	0.077
5	SIFT	0.087
6	MSER	0.226
7	BRISK	0.229
8	KAZE	0.404
9	LGFT-1.4.3	1.309
10	LGFT-2.4.3	2.246

Table 4.3 continued

11	U-RIFT	2.375
12	ePDE- g_5	3.568
13	ePDE- g_4	3.620
14	LGFT-4.4.3	4.149

As shown in Table 4.3, AKAZE, SURF, and AGAST have relatively fast processing time, although they are still slower than ORB. These methods are designed to be more efficient than their predecessors, making them ideal for real-time applications, particularly in situations where computational resources are limited. Conversely, SIFT, MSER, and KAZE are slower because they are not specifically optimized for efficiency, which is reflected in their processing time rankings in Table 4.3. Although BRISK is intended to be computationally efficient for applications requiring fast performance, its performance in this experiment is surprisingly slower than both SIFT and MSER, as shown in Table 4.3. U-RIFT, being more computationally demanding, is designed to handle multi-modal images, where analysing the inherent influence of rotations on the maximum index map for feature description can significantly increase its processing time.

The proposed ePDE and LGFT methods prioritize robustness and accuracy in feature detection, particularly in near-uniform scenes, rather than computational efficiency. Consequently, they are relatively computationally expensive. As indicated in Table 4.3, the proposed methods are regrettably positioned among the slowest feature detectors. This is expected, as the longer processing time of these methods primarily arise from the iterative processes required to generate detailed nonlinear diffusion and log-Gabor scale-space representations, as well as the extensive inlier detection.

In Table 4.3, the LGFT-1.4.3 variant is generally more efficient than other LGFT versions, and it is approximately 81.4% faster than U-RIFT. However, as the number of orientations, n_o increases, processing time also rise, making LGFT-4.4.3 the slowest among all feature detectors. Overall, the ePDE methods are more computationally demanding than LGFT, though ePDE- g_5 performs slightly faster than ePDE- g_4 . Although the proposed methods are not specifically optimized for computational efficiency in real-time applications, the additional processing time is well justified by the substantial enhancements in image stitching quality and the overall performance. The trade-off between accuracy and speed is consistent with findings from prior studies. For instance, fast detectors like ORB and BRISK prioritize speed but often sacrifice robustness and descriptiveness (Leutenegger et al., 2011; Rublee et al., 2011). Conversely, more accurate approach such as KAZE that rely on nonlinear diffusion filtering, typically involve higher computational costs (Alcantarilla et al., 2012).

4.6 Chapter Summary

This chapter presents a series of experiments, analyses, and results aims at evaluating the performance of the proposed feature-based image stitching methods in comparison with several state-of-the-art techniques, particularly in near-uniform scenes. The evaluation is conducted using a dataset comprising 218 image pairs, of which 157 are newly created near-uniform image pairs specifically designed to assess the effectiveness of the proposed methods. For optimal configuration, parameter fitting for the enhanced Partial Differential Equation (ePDE) method indicates that a contrast factor of $k = 0.5$ is most suitable for both ePDE- g_4 and ePDE- g_5 variants. In the case of the Log-Gabor Feature Transform (LGFT)

method, the optimal parameters are identified as Set3: radial bandwidth, $\sigma_{f_s} = 0.75$, filter scaling factor, $C = 1.6$, and parameter weight, $w = 2.5$, using the number of octaves, $n_o = \{1, 2, 4\}$ and number of sublevels, $n_s = 4$. Based on these configurations, three LGFT variants, LGFT-1.4.3, LGFT-2.4.3, and LGFT-4.4.3, are selected for further evaluation. These parameter choices are based on achieving optimal performance while taking processing time into consideration.

Overall, the experimental results demonstrate that the proposed LGFT and ePDE methods deliver outstanding performance in stitching near-uniform images, achieving over a 14.63% higher stitching success rate compared to existing feature-based stitching techniques. This superior performance is attributed to the richer texture granularity details captured by the LGFT scale-spaces, and the improved edge-preserving nonlinear scale-spaces generated by ePDE, both of which contribute to enhanced interest point detection. Whereas other conventional feature detectors perform significantly worse in near-uniform scenes. For instance, SIFT achieves a stitching success rate of 73.98%, while MSER records the lowest at just 4.88%.

LGFT-4.4.3 consistently detects the highest average number of inliers at 2,665 with an RC/S_o score of 0.8. However, this advantage comes at the cost of increased computational complexity. By comparison, LGFT-1.4.3 offers a better balance between accuracy and efficiency, achieving an average of 1,435 inliers with an improved RC/S_o score of 0.9. For the ePDE approach, the g_5 variant outperforms g_4 due to its more detailed scale-space structures, attaining an average of 818 inliers and an RC/S_o score of 1.5. On the contrary, other conventional feature detectors exhibit higher RC/S_o score, indicating poorer performance. For example, MSER detects only 87 inliers with a significantly higher RC/S_o

score of 15.9, suggesting a greater risk of misalignment and severe distortions in the resulting stitched images due to sparse or poorly distributed inliers.

In general, both LGFT and ePDE methods are more computationally extensive than the other evaluated feature detectors, typically complete execution in just under 0.5 seconds. LGFT-1.4.3 recorded an average processing time of 1.309 seconds, while maintaining strong interest points detection performance. This efficiency is largely due to its use of a single oriented scale-space. By comparison, LGFT-4.4.3 employs multiple orientations has the longest processing time of 4.149 seconds, making it the slowest among all the evaluated feature detectors. Generally, ePDE methods are found to be more computationally intensive than LGFT. In conclusion, when considering the balance between accuracy and efficiency, LGFT-1.4.3 followed by ePDE- g_5 emerge as the most effective and reliable feature detectors for stitching near-uniform images. Their ability to detect a large number of accurately matched and well-distributed inliers underlines their suitability for challenging near-uniform image stitching, although this comes at the cost of longer processing times compared to the faster execution of existing feature detectors.

CHAPTER 5

CONCLUSION AND FUTURE WORKS

5.1 Conclusion

Image stitching has become a key technique in aligning multiple overlapping images of the same scene to create seamless and high-resolution image. It plays a vital role in numerous modern multimedia applications, including panoramic photography, virtual reality, aerial surveillance, medical imaging, and astronomical imaging. While significant advancements have been made in this field, image stitching remains a persistent challenge in computer vision. In particular, stitching near-uniform scenes has received limited research attention, a gap that this thesis aims to address.

To overcome these challenges, this thesis proposes two novel feature-based image stitching methods: the enhanced Partial Differential Equation (ePDE) and the Log-Gabor Feature Transform (LGFT). The ePDE method utilizes a Lorentz-modulated nonlinear diffusion model in the spatial domain, while LGFT employs two-dimensional Log-Gabor filters in the frequency domain. These methods, forming the first two objectives of this research, are designed to enhance scale-space representations and improve interest point detection across wider spatial regions, particularly in near-uniform scenes. In addition, two new evaluation metrics, the spread-overlap, S_o and the recall-to-spread-overlap ratio, RC/S_o , are proposed as the third objective to provide a more comprehensive assessment of feature detection effectiveness in relation to image stitching quality.

A total of 218 image pairs, including 157 newly generated near-uniform pairs, are used to evaluate the performance of the proposed methods. Comparative experiments are

conducted against several state-of-the-art feature detectors, including MSER, SIFT, SURF, BRISK, KAZE, AKAZE, AGAST, ORB, and U-RIFT. The experimental findings indicate that LGFT and ePDE achieve over a 14.63% higher image stitching success rate, significantly outperforming other evaluated feature methods in handling near-uniform scenes.

While the proposed methods exhibit superior performance, they incur higher computational costs. The average execution time for LGFT-1.4.3, for instance, is 1.309 seconds, compared to just under 0.5 seconds for many conventional techniques. However, given that the proposed methods prioritize robustness and accuracy over speed, the increased processing time is considered acceptable considering the substantial improvements in stitching quality. In conclusion, the proposed ePDE and LGFT methods effectively address the challenges of stitching near-uniform images and provide a solid basis for future research, especially toward real-time optimization.

5.2 Limitations

The ePDE and LGFT methods are designed to enhance the detection of interest points, particularly in near-uniform scenes characterized by homogeneous or featureless regions. However, this improvement comes with a trade-off of higher computational complexity when applied to nonuniform or feature-rich scenes. Although generating detailed nonlinear diffusion and Log-Gabor scale-space representations involves iterative processing that contribute to longer runtimes (Li et al., 2020), the primary factor behind the extended processing time is the additional time required to identify a greater number of inliers during non-maxima suppression across multiple scale-spaces (Bay et al., 2006; Lowe, 2004). In

nonuniform scenes with more complex and salient features, both ePDE and LGFT tend to detect even more inliers than in near-uniform scenes, resulting in further increases in processing time. Moreover, as the number of scale-spaces increases, so does the computational cost of performing non-maxima suppression. However, in certain applications, the limitation of lower computational speed may be acceptable or even advantageous. These include offline high-resolution image stitching tasks such as aerial imaging (Li & Zhou, 2024), medical imaging (Mohammadi et al., 2024), and astronomical imaging (Wang et al., 2024), where accuracy is prioritized over real-time performance.

It is important to note that the ePDE and LGFT methods are not intended for aligning and stitching images with significant depth variation. When images are captured from different viewpoints as the camera moves between shots, objects within the images can shift relative to each other. This shift can cause the generated warps to "break," resulting in visible seams, distortions, or ghosting, where objects fail to align properly. These issues, known as parallax errors, are particularly challenging in scenes with severe depth variation and require specialized methods to address them (Li et al., 2018; K. Lin et al., 2016; Zaragoza et al., 2014). As a result, the nonlinear scale-spaces generated by ePDE and LGFT are less effective at detecting inliers, especially in images with significant viewpoint changes, such as the ‘graffiti’, ‘wall’, ‘bark’, and ‘boat’ images from the Affine Covariant Features (2007) dataset.

5.3 Contributions

This thesis introduces the ePDE and LGFT methods, making a significant contribution to the field of image stitching, especially in handling near-uniform scenes. Such

scenes typically consist of extensive uniform regions, either partially or entirely, that lack distinctive features, making it challenging for current state-of-the-art methods to identify sufficient corresponding interest points across overlapping areas. The proposed methods effectively overcome this limitation, consistently delivering well-aligned stitching results with minimal or no misalignment. This marks an improvement over current methods, such as MSER, SIFT, SURF, BRISK, KAZE, AKAZE, AGAST, ORB, and U-RIFT. Furthermore, this thesis demonstrates that the proposed methods also perform well across a variety of general nonuniform datasets.

Performance evaluation is essential in image processing, including image stitching, to validate the quality, reliability, applicability, and efficiency of the stitched images. Unfortunately, there is no single gold standard for assessing the performance of image stitching (Bergen & Wittenberg, 2016). Researchers commonly rely on visual comparison (Bergen & Wittenberg, 2016) and the metrics, such as repeatability (Mikolajczyk & Schmid, 2001; Schmid et al., 2000), recall and precision (Mikolajczyk et al., 2005), primarily designed for evaluating feature detectors and descriptors. These metrics have no direct correlation to the quality of image stitching outcomes. To address this issue, this thesis proposes and implements two new evaluation metrics for image stitching: the spread-overlap, S_o and the recall-to-spread-overlap ratio, RC/S_o . These metrics are designed to assess stitching quality by analysing the spatial distribution of detected inliers. They provide a meaningful measure of how the distribution and accuracy of inliers influence the quality of the final stitched images, serving as performance indicators for the proposed feature-based stitching techniques.

Image datasets are essential for various tasks in image processing, machine learning, and computer vision, serving as benchmarks for algorithm development and assessment. The majority of current datasets are primarily created for tasks including image classification, image retrieval, facial recognition and object detection. However, to the best of the author's knowledge, very few standard benchmark datasets are available for algorithm development and evaluation in image stitching, and it is especially difficult to find image pairs involving near-uniform scenes (Yang et al., 2025). This lack of resources hinders researchers from making effective benchmarking comparisons of algorithm performance. To address this gap, this thesis contributes datasets comprising 218 image pairs, of which 157 are newly generated. These include images captured by the author as well as those cropped from HiRISE | Image Catalog (n.d.). These datasets will expand the pool of available resources for future research.

5.4 Future Works

The proposed methods, while effective in improving stitching quality for near-uniform scenes, exhibit limitations in terms of computational efficiency. This poses challenges for real-time deployment, particularly in low-latency applications such as virtual reality, augmented reality and autonomous navigation. To mitigate this issue, a promising direction for future research is the adoption of patch-based mapping techniques that focus on smaller, localized regions of an image instead of processing the entire image globally. Patch-based strategies have been explored in several studies. For instance, Liu et al. (2018) proposed a patch-wise segmentation framework for large-scale remote sensing images, where each image is divided into smaller patches and processed individually using a

dedicated patch-wise network. Similarly, recent surveys on deep learning for local feature matching highlight the patch-based matching strategies in enhancing correspondences accuracy by extracting features from localized image patches (Huang et al., 2024; Xu et al., 2024). Therefore, applying this strategy to the proposed methods can reduce unnecessary processing by concentrating computational resources on local image regions. It also enables parallel or hierarchical execution, thereby improving scalability and runtime efficiency.

Another key challenge in image stitching research is the lack of standard benchmark datasets, especially for near-uniform scenes (Yang et al., 2025). Developing new and diverse image datasets, including near-uniform and low-texture image pairs, presents a valuable direction for future work. Such datasets would not only support the development of more robust algorithms, both the handcrafted and deep learning-based feature detection methods but also serve as standardized benchmarks for objective performance evaluation.

A particularly promising application area lies in astronomical imaging, where scenes are often near-uniform in nature. Faint and blurred celestial objects are typically embedded within uniform and noisy backgrounds, making it difficult to detect enough well-distributed inliers for accurate alignment. This often results in stitching artifacts such as ghosting, misalignment, and geometric distortion that may affect scientific interpretations. A recent study by Wang et al. (2024) applies deep learning techniques for stitching astronomical images. Although their method successfully extracted and matched keypoints, the keypoints tended to cluster in specific regions, and the authors also highlighted persistent challenges in handling the extreme noisy or low-contrast conditions common in deep-space imagery. Given the demonstrated robustness of the proposed ePDE and LGFT methods in handling near-uniform scenes, future work could explore their application to astronomical image

stitching, an area that similarly features low-texture and noisy backgrounds, where the proposed methods may offer substantial improvements in alignment accuracy and visual quality.

REFERENCES

- Abdel-Hakim, A. E., & Farag, A. A. (2006). CSIFT: A SIFT descriptor with color invariant characteristics. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2, 1978–1983. <https://doi.org/10.1109/CVPR.2006.95>
- Adobe Open Source Data Sets*. (n.d.). Retrieved November 19, 2023, from <https://sourceforge.net/adobe/adobedatasets/home/Home/>
- Affine Covariant Features*. (2007). Retrieved November 19, 2023, from <https://www.robots.ox.ac.uk/~vgg/research/affine/>
- Agrawal, M., Konolige, K., & Blas, M. R. (2008). CenSurE: Center surround extremas for realtime feature detection and matching. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5305 LNCS(PART 4), 102–115. https://doi.org/10.1007/978-3-540-88693-8_8/COVER
- Alahi, A., Ortiz, R., & Vandergheynst, P. (2012). FREAK: Fast retina keypoint. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 510–517. <https://doi.org/10.1109/CVPR.2012.6247715>
- Alcantarilla, P. F., Bartoli, A., & Davison, A. J. (2012). KAZE Features. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7577 LNCS(PART 6), 214–227. https://doi.org/10.1007/978-3-642-33783-3_16
- Alcantarilla, P., Nuevo, J., & Bartoli, A. (2013). Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces. *Procedings of the British Machine Vision Conference 2013*, 13.1-13.11. <https://doi.org/10.5244/C.27.13>

- Awad, A. I., & Hassaballah, M. (2016). Image Feature Detectors and Descriptors. *Studies in Computational Intelligence*, 630. <https://doi.org/10.1007/978-3-319-28854-3>
- Arrospide, J., & Salgado, L. (2013). Log-Gabor Filters for Image-Based Vehicle Verification. *IEEE Transactions on Image Processing*, 22(6), 2286–2295. <https://doi.org/10.1109/TIP.2013.2249080>
- Bambi, C. (2018). *Introduction to General Relativity*. <https://doi.org/10.1007/978-981-13-1090-4>
- Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding*, 110(3), 346–359. <https://doi.org/10.1016/j.cviu.2007.09.014>
- Bay, H., Tuytelaars, T., & Van Gool, L. (2006). SURF: Speeded up robust features. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3951 LNCS, 404–417. https://doi.org/10.1007/11744023_32/COVER
- Bergen, T., & Wittenberg, T. (2016). Stitching and surface reconstruction from endoscopic image sequences: A review of applications and methods. In *IEEE Journal of Biomedical and Health Informatics* (Vol. 20, Issue 1, pp. 304–321). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/JBHI.2014.2384134>
- Bhatia, H., Kirby, R. M., Pascucci, V., & Bremer, P. T. (2021). Vector Field Decompositions Using Multiscale Poisson Kernel. *IEEE Transactions on Visualization and Computer Graphics*, 27(9), 3781–3793. <https://doi.org/10.1109/TVCG.2020.2984413>
- Biljecki, F., & Ito, K. (2021). Street view imagery in urban analytics and GIS: A review. *Landscape and Urban Planning*, 215, 104217. <https://doi.org/10.1016/J.LANDURBPLAN.2021.104217>

- Boykov, Y., Veksler, O., & Zabih, R. (1999). Fast approximate energy minimization via graph cuts. *Proceedings of the IEEE International Conference on Computer Vision*, 1, 377–384. <https://doi.org/10.1109/ICCV.1999.791245>
- Brown, M., & Lowe, D. G. (2007). Automatic panoramic image stitching using invariant features. *International Journal of Computer Vision*, 74(1), 59–73. <https://doi.org/10.1007/S11263-006-0002-3/METRICS>
- Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (2010). BRIEF: Binary Robust Independent Elementary Features. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (pp. 778–792). https://doi.org/10.1007/978-3-642-15561-1_56
- Chang, C. H., Sato, Y., & Chuang, Y. Y. (2014). Shape-preserving half-projective warps for image stitching. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 3254–3261. <https://doi.org/10.1109/CVPR.2014.422>
- Daugman, J. G. (1985). Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America A*, 2(7), 1160. <https://doi.org/10.1364/JOSAA.2.001160>
- Detone, D., Malisiewicz, T., & Rabinovich, A. (2018). SuperPoint: Self-supervised interest point detection and description. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2018-June*, 337–349. <https://doi.org/10.1109/CVPRW.2018.00060>
- Dong, H., & Ganz, A. (2019). Cost Efficient Virtual Environment Generation Framework Using Annotated Panoramic Videos. *IEEE Access*, 7, 121024–121036. <https://doi.org/10.1109/ACCESS.2019.2937877>
- Duits, R., Florack, L., De Graaf, J., & Ter Haar Romeny, B. (2004). On the axioms of scale

- space theory. *Journal of Mathematical Imaging and Vision*, 20(3), 267–298.
<https://doi.org/10.1023/B:JMIV.0000024043.96722.AA/METRICS>
- El Abbadi, N. K., Al Hassani, S. A., & Abdulkhaleq, A. H. (2021). A Review over Panoramic Image Stitching Techniques. *Journal of Physics: Conference Series*, 1999(1).
<https://doi.org/10.1088/1742-6596/1999/1/012115>
- Evans, C. (2009). Notes on the OpenSURF Library. *University of Bristol Tech Rep CSTR09001 January*.
- Field, D. J. (1987). Relations between the statistics of natural images and the response properties of cortical cells. *Journal of the Optical Society of America A*, 4(12), 2379.
<https://doi.org/10.1364/JOSAA.4.002379>
- Fischer, S., Šroubek, F., Perrinet, L., Redondo, R., & Cristóbal, G. (2007). Self-Invertible 2D Log-Gabor Wavelets. *International Journal of Computer Vision*, 75(2), 231–246.
<https://doi.org/10.1007/s11263-006-0026-8>
- Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus. *Communications of the ACM*, 24(6), 381–395. <https://doi.org/10.1145/358669.358692>
- Fu, M., Liang, H., Zhu, C., Dong, Z., Sun, R., Yue, Y., & Yang, Y. (2023). Image Stitching Techniques Applied to Plane or 3-D Models: A Review. *IEEE Sensors Journal*, 23(8), 8060–8079. <https://doi.org/10.1109/JSEN.2023.3251661>
- Gao, J., Kim, S. J., & Brown, M. S. (2011). Constructing image panoramas using dual-homography warping. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 49–56.
<https://doi.org/10.1109/CVPR.2011.5995433>
- Gao, J., Li, Y., Chin, T., & Brown, M. (2013). *Seam-driven image stitching*.
<https://digital.library.adelaide.edu.au/dspace/handle/2440/83952>

- Gao, X., Sattar, F., & Venkateswarlu, R. (2007). Multiscale Corner Detection of Gray Level Images Based on Log-Gabor Wavelet Transform. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(7), 868–875. <https://doi.org/10.1109/TCSVT.2007.897473>
- Ghosh, D., & Kaabouch, N. (2016). A survey on image mosaicing techniques. *Journal of Visual Communication and Image Representation*, 34, 1–11. <https://doi.org/10.1016/J.JVCIR.2015.10.014>
- Gómez-Reyes, J. K., Benítez-Rangel, J. P., Morales-Hernández, L. A., Resendiz-Ochoa, E., & Camarillo-Gomez, K. A. (2022). Image Mosaicing Applied on UAVs Survey. *Applied Sciences (Switzerland)*, 12(5). <https://doi.org/10.3390/APP12052729>
- Ha, S. J., Koo, H. Il, Lee, S. H., Cho, N. I., & Kim, S. K. (2007). Panorama mosaic optimization for mobile camera systems. *IEEE Transactions on Consumer Electronics*, 53(4), 1217–1225. <https://doi.org/10.1109/TCE.2007.4429204>
- Harris, C., & Stephens, M. (1988). A Combined Corner and Edge Detector. *Proceedings of the Alvey Vision Conference 1988*, 23.1-23.6. <https://doi.org/10.5244/C.2.23>
- HiRISE | Image Catalog. (n.d.). Retrieved July 28, 2024, from <https://www.uahirise.org/catalog/>
- Hu, M., Sun, B., Kang, X., & Li, S. (2023). Multiscale structural feature transform for multi-modal image matching. *Information Fusion*, 95, 341–354. <https://doi.org/10.1016/J.INFFUS.2023.02.026>
- Huang, Q., Guo, X., Wang, Y., Sun, H., & Yang, L. (2024). A survey of feature matching methods. *IET Image Processing*, 18(6), 1385–1410. <https://doi.org/10.1049/IPR2.13032;PAGEGROUP:STRING:PUBLICATION>
- Jagadeeswari, M., Manikandababu, C. S., Dhviya, M. S., & Meenakshi, J. V. (2021).

- Review: A Comparative Study based on Video Stitching Methods. *Proceedings of the 2nd International Conference on Electronics and Sustainable Communication Systems, ICESC 2021*, 1157–1163. <https://doi.org/10.1109/ICESC51422.2021.9532635>
- Jiang, S., Li, Y., Weng, D., You, K., & Chen, W. (2023). *3D reconstruction from spherical images: A review of techniques, applications, and prospects*. <http://arxiv.org/abs/2302.04495>
- Joshi, K., & Patel, M. I. (2020). Recent advances in local feature detector and descriptor: a literature survey. *International Journal of Multimedia Information Retrieval*, 9(4), 231–247. <https://doi.org/10.1007/s13735-020-00200-3>
- Ke, Y., & Sukthankar, R. (2004). PCA-SIFT: A more distinctive representation for local image descriptors. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2. <https://doi.org/10.1109/CVPR.2004.1315206>
- Koenderink, J. J. (1984). The structure of images. *Biological Cybernetics*, 50(5), 363–370. <https://doi.org/10.1007/BF00336961/METRICS>
- Kovesi, P. (2003). Phase congruency detects corners and edges. *Digital Image Computing: Techniques and Applications: Proceedings of the VIIth Biennial Australian Pattern Recognition Society Conference - DICTA 2003*.
- Laato, S., Söbke, H., & Baer, M. F. (2024). Augmented future: tracing the trajectory of location-based augmented reality gaming for the next ten years. *I-Com*, 23(2), 189–203. https://doi.org/10.1515/ICOM-2024-0018/ASSET/GRAPHIC/J_ICOM-2024-0018_CV_003.JPG
- Leutenegger, S., Chli, M., & Siegwart, R. Y. (2011). BRISK: Binary Robust invariant scalable keypoints. *Proceedings of the IEEE International Conference on Computer Vision*, 2548–2555. <https://doi.org/10.1109/ICCV.2011.6126542>

- Li, D., Xu, Q., Yu, W., & Wang, B. (2020). SRP-AKAZE: an improved accelerated KAZE algorithm based on sparse random projection. *IET Computer Vision*, 14(4), 131–137. <https://doi.org/10.1049/iet-cvi.2019.0622>
- Li, J., Deng, B., Tang, R., Wang, Z., & Yan, Y. (2020). Local-Adaptive Image Alignment Based on Triangular Facet Approximation. *IEEE Transactions on Image Processing*, 29, 2356–2369. <https://doi.org/10.1109/TIP.2019.2949424>
- Li, J., Hu, Q., & Ai, M. (2020). RIFT: Multi-Modal Image Matching Based on Radiation-Variation Insensitive Feature Transform. *IEEE Transactions on Image Processing*, 29, 3296–3310. <https://doi.org/10.1109/TIP.2019.2959244>
- Li, J., Wang, Z., Lai, S., Zhai, Y., & Zhang, M. (2018). Parallax-Tolerant Image Stitching Based on Robust Elastic Warping. *IEEE Transactions on Multimedia*, 20(7), 1672–1687. <https://doi.org/10.1109/TMM.2017.2777461>
- Li, J., & Zhou, Y. (2024). Seam-Adaptive Structure-Preserving Image Stitching for Drone Images. *IEEE Transactions on Geoscience and Remote Sensing*. <https://doi.org/10.1109/TGRS.2024.3515111>
- Li, Y., Wang, S., Tian, Q., & Ding, X. (2015). A survey of recent advances in visual feature detection. *Neurocomputing*, 149, 736–751. <https://doi.org/10.1016/j.neucom.2014.08.003>
- Lin, C. C., Pankanti, S. U., Ramamurthy, K. N., & Aravkin, A. Y. (2015). Adaptive as-natural-as-possible image stitching. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 07-12-June-2015*, 1155–1163. <https://doi.org/10.1109/CVPR.2015.7298719>
- Lin, K., Jiang, N., Cheong, L.-F., Do, M., & Lu, J. (2016). SEAGULL: Seam-Guided Local Alignment for Parallax-Tolerant Image Stitching. In *Lecture Notes in Computer*

- Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (pp. 370–385). https://doi.org/10.1007/978-3-319-46487-9_23
- Lin, W. Y., Liu, S., Matsushita, Y., Ng, T. T., & Cheong, L. F. (2011). Smoothly varying affine stitching. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 345–352. <https://doi.org/10.1109/CVPR.2011.5995314>
- 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. <https://doi.org/10.1007/BF01469346/METRICS>
- Lindeberg, T. (1994). Scale-Space Theory: A Basic Tool for Analysing Structures at Different Scales. *Journal of Applied Statistics*, 21(1–2), 225–270. <https://doi.org/10.1080/757582976>
- Liu, Y., Ren, Q., Geng, J., Ding, M., & Li, J. (2018). Efficient Patch-Wise Semantic Segmentation for Large-Scale Remote Sensing Images. *Sensors 2018, Vol. 18, Page 3232*, 18(10), 3232. <https://doi.org/10.3390/S18103232>
- Lowe, D. G. (1999). Object recognition from local scale-invariant features. *Proceedings of the IEEE International Conference on Computer Vision*, 2, 1150–1157. <https://doi.org/10.1109/ICCV.1999.790410>
- Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2), 91–110. <https://doi.org/10.1023/B:VISI.0000029664.99615.94>
- Lyu, W., Zhou, Z., Chen, L., & Zhou, Y. (2019). A survey on image and video stitching. *Virtual Reality and Intelligent Hardware*, 1(1), 55–83. <https://doi.org/10.3724/SP.J.2096-5796.2018.0008>

- Ma, J., Jiang, X., Fan, A., Jiang, J., & Yan, J. (2021). Image Matching from Handcrafted to Deep Features: A Survey. *International Journal of Computer Vision*, 129(1), 23–79. <https://doi.org/10.1007/S11263-020-01359-2>
- Madhusudana, P. C., & Soundararajan, R. (2019). Subjective and objective quality assessment of stitched images for virtual reality. *IEEE Transactions on Image Processing*, 28(11), 5620–5635. <https://doi.org/10.1109/TIP.2019.2921858>
- Mair, E., Hager, G. D., Burschka, D., Suppa, M., & Hirzinger, G. (2010). Adaptive and Generic Corner Detection Based on the Accelerated Segment Test. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (pp. 183–196). https://doi.org/10.1007/978-3-642-15552-9_14
- Marmol, A., Peynot, T., Eriksson, A., Jaiprakash, A., Roberts, J., & Crawford, R. (2017). Evaluation of Keypoint Detectors and Descriptors in Arthroscopic Images for Feature-Based Matching Applications. *IEEE Robotics and Automation Letters*, 2(4), 2135–2142. <https://doi.org/10.1109/LRA.2017.2714150>
- Matas, J., Chum, O., Urban, M., & Pajdla, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10), 761–767. <https://doi.org/10.1016/j.imavis.2004.02.006>
- Miao, X., Qu, T., Chen, X., & He, C. (2023). Superpixel-based foreground-preserving image stitching. *Machine Vision and Applications*, 34(1), 1–13. <https://doi.org/10.1007/S00138-022-01363-1/METRICS>
- Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., & Gool, L. Van. (2005). A Comparison of Affine Region Detectors. *International Journal of Computer Vision*, 65(1–2), 43–72.

- <https://doi.org/10.1007/s11263-005-3848-x>
- Mikolajczyk, K., & Schmid, C. (2001). Indexing based on scale invariant interest points. *Proceedings of the IEEE International Conference on Computer Vision, 1*, 525–531. <https://doi.org/10.1109/ICCV.2001.937561>
- Mikolajczyk, K., & Schmid, C. (2002). An Affine Invariant Interest Point Detector. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2350, 128–142. https://doi.org/10.1007/3-540-47969-4_9
- Mikolajczyk, K., & Schmid, C. (2004). Scale & affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1), 63–86. <https://doi.org/10.1023/B:VISI.0000027790.02288.F2/METRICS>
- Mikolajczyk, K., & Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10), 1615–1630. <https://doi.org/10.1109/TPAMI.2005.188>
- Mohammadi, F. S., Mohammadi, S. E., Mojarad Adi, P., Mirkarimi, S. M. A., & Shabani, H. (2024). A comparative analysis of pairwise image stitching techniques for microscopy images. *Scientific Reports 2024 14:1*, 14(1), 1–10. <https://doi.org/10.1038/s41598-024-59626-y>
- Moravec, H. P. (1977). Towards Automatic Visual Obstacle Avoidance. *International Joint Conference on Artificial Intelligence*.
- Morel, J. M., & Yu, G. (2009). ASIFT: A New Framework for Fully Affine Invariant Image Comparison. <https://doi.org/10.1137/080732730>, 2(2), 438–469. <https://doi.org/10.1137/080732730>
- Muja, M., & Lowe, D. G. (2009). Fast Approximate Nearest Neighbors with Automatic

- Algorithm Configuration. *Proceedings of the Fourth International Conference on Computer Vision Theory and Applications*, 331–340. <https://doi.org/10.5220/0001787803310340>
- Nam, D. Y., & Han, J. K. (2021). An Efficient Algorithm for Generating Harmonized Stereoscopic 360° VR Images. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(12), 4864–4882. <https://doi.org/10.1109/TCSVT.2021.3056065>
- Nava, R., Escalante-Ramírez, B., & Cristóbal, G. (2012). Texture Image Retrieval Based on Log-Gabor Features. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (pp. 414–421). https://doi.org/10.1007/978-3-642-33275-3_51
- Ni, J., Li, Y., Ke, C., Zhang, Z., Cao, W., & Yang, S. X. (2024). A Fast Unsupervised Image Stitching Model Based on Homography Estimation. *IEEE Sensors Journal*, 24(18), 29452–29467. <https://doi.org/10.1109/JSEN.2024.3436051>
- Nie, L., Lin, C., Liao, K., Liu, M., & Zhao, Y. (2020). A view-free image stitching network based on global homography. *Journal of Visual Communication and Image Representation*, 73, 102950. <https://doi.org/10.1016/j.jvcir.2020.102950>
- OpenCV Extra Data*. (n.d.). Retrieved November 19, 2023, from https://github.com/opencv/opencv_extra/tree/master/testdata/stitching
- O’Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., & Walsh, J. (2020). Deep Learning vs. Traditional Computer Vision. *Advances in Intelligent Systems and Computing*, 943(Cv), 128–144. https://doi.org/10.1007/978-3-030-17795-9_10
- Pandey, A., & Pati, U. C. (2019). Image mosaicing: A deeper insight. *Image and Vision Computing*, 89, 236–257. <https://doi.org/https://doi.org/10.1016/j.imavis.2019.07.002>

- Paul, S., & Pati, U. C. (2021). A comprehensive review on remote sensing image registration. *International Journal of Remote Sensing*, 42(14), 5400–5436. <https://doi.org/10.1080/01431161.2021.1906985>
- Perona, P., & Malik, J. (1990). Scale-Space and Edge Detection Using Anisotropic Diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7), 629–639. <https://doi.org/10.1109/34.56205>
- Photojournal: NASA's Image Access Home Page*. (n.d.). Retrieved February 11, 2023, from <https://photojournal.jpl.nasa.gov/>
- Qureshi, H. S., Khan, M. M., Hafiz, R., Cho, Y., & Cha, J. (2012). Quantitative quality assessment of stitched panoramic images. *IET Image Processing*, 6(9), 1348–1358. <https://doi.org/10.1049/IET-IPR.2011.0641/CITE/REFWORKS>
- Ralli, J. (2011). *Fusion and Regularisation of Image Information in Variational Correspondence Methods*.
- Rosten, E., & Drummond, T. (2005). Fusing points and lines for high performance tracking. *Proceedings of the IEEE International Conference on Computer Vision, II*, 1508–1515. <https://doi.org/10.1109/ICCV.2005.104>
- Rosten, E., & Drummond, T. (2006). Machine Learning for High-Speed Corner Detection. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (pp. 430–443). https://doi.org/10.1007/11744023_34
- Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. *Proceedings of the IEEE International Conference on Computer Vision*, 2564–2571. <https://doi.org/10.1109/ICCV.2011.6126544>
- Rudin, L. I., Osher, S., & Fatemi, E. (1992). Nonlinear total variation based noise removal

- algorithms. *Physica D: Nonlinear Phenomena*, 60(1–4), 259–268.
[https://doi.org/10.1016/0167-2789\(92\)90242-F](https://doi.org/10.1016/0167-2789(92)90242-F)
- Sarlin, P. E., Detone, D., Malisiewicz, T., & Rabinovich, A. (2020). SuperGlue: Learning Feature Matching with Graph Neural Networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 4937–4946.
<https://doi.org/10.1109/CVPR42600.2020.00499>
- Schmid, C., Mohr, R., & Bauckhage, C. (2000). Evaluation of interest point detectors. *International Journal of Computer Vision*, 37(2), 151–172.
<https://doi.org/10.1023/A:1008199403446/METRICS>
- Shi, J., & Tomasi, C. (1994). Good features to track. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 593–600.
<https://doi.org/10.1109/CVPR.1994.323794>
- Smith, S. M., & Brady, J. M. (1997). SUSAN - A new approach to low level image processing. *International Journal of Computer Vision*, 23(1), 45–78.
<https://doi.org/10.1023/A:1007963824710/METRICS>
- Szeliski, R. (2007). Image Alignment and Stitching: A Tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 2(1), 1–104. <https://doi.org/10.1561/06000000009>
- Szeliski, R. (2022). *Computer Vision*. <https://doi.org/10.1007/978-3-030-34372-9>
- Tian, C., Chai, X., & Shao, F. (2021). Stitched image quality assessment based on local measurement errors and global statistical properties. *Journal of Visual Communication and Image Representation*, 81, 103324. <https://doi.org/10.1016/J.JVCIR.2021.103324>
- Torr, P. H. S., & Zisserman, A. (2000). MLESAC: A New Robust Estimator with Application to Estimating Image Geometry. *Computer Vision and Image Understanding*, 78(1), 138–156. <https://doi.org/10.1006/cviu.1999.0832>

- Ullah, H., Afzal, S., & Khan, I. U. (2022). Perceptual Quality Assessment of Panoramic Stitched Contents for Immersive Applications: A Prospective Survey. *Virtual Reality and Intelligent Hardware*, 4(3), 223–246. <https://doi.org/10.1016/j.vrih.2022.03.004>
- Ullah, H., Irfan, M., Han, K., & Lee, J. W. (2020). DLNR-SIQA: Deep Learning-Based No-Reference Stitched Image Quality Assessment. *Sensors* 2020, Vol. 20, Page 6457, 20(22), 6457. <https://doi.org/10.3390/S20226457>
- Vedaldi, A., & Fulkerson, B. (2010). Vlfeat - An open and portable library of computer vision algorithms. *MM'10 - Proceedings of the ACM Multimedia 2010 International Conference*, 1469–1472. <https://doi.org/10.1145/1873951.1874249>
- Wang, J. G., Li, J., Lee, C. Y., & Yau, W. Y. (2010). Dense SIFT and Gabor descriptors-based face representation with applications to gender recognition. *11th International Conference on Control, Automation, Robotics and Vision, ICARCV 2010*, 1860–1864. <https://doi.org/10.1109/ICARCV.2010.5707370>
- Wang, S., You, H., & Fu, K. (2012). BFSIFT: A Novel Method to Find Feature Matches for SAR Image Registration. *IEEE Geoscience and Remote Sensing Letters*, 9(4), 649–653. <https://doi.org/10.1109/LGRS.2011.2177437>
- Wang, T., Lin, X., & Zhang, P. (2024). Enhancing Astronomical Image Stitching with XFeat: A Deep Learning Approach. *2024 International Conference on Image Processing, Computer Vision and Machine Learning, ICICML 2024*, 708–712. <https://doi.org/10.1109/ICICML63543.2024.10958013>
- Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4), 600–612. <https://doi.org/10.1109/TIP.2003.819861>
- Wang, Z., & Yang, Z. (2020). Review on image-stitching techniques. *Multimedia Systems*,

- 26(4), 413–430. <https://doi.org/10.1007/S00530-020-00651-Y/METRICS>
- Wei, Z., Lan, C., Xu, Q., Wang, L., Gao, T., Yao, F., & Hou, H. (2024). SatellStitch: Satellite Imagery-Assisted UAV Image Seamless Stitching for Emergency Response without GCP and GNSS. *Remote Sensing* 2024, Vol. 16, Page 309, 16(2), 309. <https://doi.org/10.3390/RS16020309>
- Weickert, J. (1996). *Theoretical Foundations of Anisotropic Diffusion in Image Processing*. 221–236. https://doi.org/10.1007/978-3-7091-6586-7_13
- Weickert, J. (2001). Efficient image segmentation using partial differential equations and morphology. *Pattern Recognition*, 34(9), 1813–1824. [https://doi.org/10.1016/S0031-3203\(00\)00109-6](https://doi.org/10.1016/S0031-3203(00)00109-6)
- Weickert, J., Ishikawa, S., & Imiya, A. (1999). Linear scale-space has first been proposed in Japan. *Journal of Mathematical Imaging and Vision*, 10(3), 237–252. <https://doi.org/10.1023/A:1008344623873/METRICS>
- Weickert, J., Romeny, B. M. T. H., & Viergever, M. A. (1998). Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Transactions on Image Processing*, 7(3), 398–410. <https://doi.org/10.1109/83.661190>
- Witkin, A. P. (1984). Scale-Space Filtering: A New Approach To Multi-Scale Description. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 3. <https://doi.org/10.1109/ICASSP.1984.1172729>
- Xiang, T.-Z., Xia, G.-S., Bai, X., & Zhang, L. (2018). Image stitching by line-guided local warping with global similarity constraint. *Pattern Recognition*, 83, 481–497. <https://doi.org/10.1016/j.patcog.2018.06.013>
- Xiong, Y., & Pulli, K. (2010). Fast panorama stitching for high-quality panoramic images on mobile phones. *IEEE Transactions on Consumer Electronics*, 56(2), 298–306.

- <https://doi.org/10.1109/TCE.2010.5505931>
- Xu, S., Chen, S., Xu, R., Wang, C., Lu, P., & Guo, L. (2024). *Local Feature Matching Using Deep Learning: A Survey*. <https://doi.org/10.1016/j.inffus.2024.102344>
- Yang, Z., Yin, Y., Xu, H., Jing, Q., Jiang, Z., Liao, T., & Guedes Soares, C. (2025). Advancements of Image and Video Stitching Techniques: A Review. *IEEE Sensors Journal*. <https://doi.org/10.1109/JSEN.2025.3563082>
- Ye, Y., Shan, J., Bruzzone, L., & 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. <https://doi.org/10.1109/TGRS.2017.2656380>
- Zaragoza, J., Chin, T. J., Tran, Q. H., Brown, M. S., & Suter, D. (2014). As-Projective-As-Possible Image Stitching with Moving DLT. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7), 1285–1298. <https://doi.org/10.1109/TPAMI.2013.247>
- Zhang, J., & Xiu, Y. (2023). Image stitching based on human visual system and SIFT algorithm. *Visual Computer*, 1–13. <https://doi.org/10.1007/S00371-023-02791-4/METRICS>
- Zhang, Q., Wang, Y., & Wang, L. (2015). Registration of images with affine geometric distortion based on Maximally Stable Extremal Regions and phase congruency. *Image and Vision Computing*, 36, 23–39. <https://doi.org/https://doi.org/10.1016/j.imavis.2015.01.008>
- Zheng, L., Yang, Y., & Tian, Q. (2018). SIFT Meets CNN: A Decade Survey of Instance Retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(5), 1224–1244. <https://doi.org/10.1109/TPAMI.2017.2709749>

- Zhou, Y., Gong, W., Sun, Y., Li, L., Gu, K., & Wu, J. (2023). Quality Assessment for Stitched Panoramic Images via Patch Registration and Bidimensional Feature Aggregation. *IEEE Transactions on Multimedia*.
<https://doi.org/10.1109/TMM.2023.3310276>
- Zhou, Z., Meng, M., Zhou, Y., Zhu, Z., & You, J. (2023). Model-guided 3D stitching for augmented virtual environment. *Science China Information Sciences*, 66(1), 1–16.
<https://doi.org/10.1007/S11432-021-3323-2/METRICS>

APPENDICES

Appendix A: Publications

1. **Jong, T. K.**, & Bong, D. B. L. (2024). Lorentz-modulated multi-scale nonlinear diffusion for stitching in near-uniform scenes. *Multimedia Tools and Applications 2024*, 1-24.
2. **Jong, T. K.**, & Bong, D. B. L. (2023). An effective feature detection approach for image stitching of near-uniform scenes. *Signal Processing: Image Communication*, 110, 116872.
3. Bong, D. B. L. & **Jong, T.K.** (2023). Image Stitching Evaluation Metric. MyIPO No. LY2023Q01292. *Intellectual Property Corporation of Malaysia (MyIPO)*. (Copyright).
4. **Jong, T. K.**, & Bong, D. B. L. (2019). Evaluation of Local Features for Near-Uniform Scene Images. *Proceedings of the 2019 IEEE International Conference on Signal and Image Processing Applications, ICSIPA 2019*, 95-99.