COLLEGES SCIENCES
BRETAGNE POUR L'INGENIEUR
LOIRE ET LE NUMERIQUE

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

THÈSE DE DOCTORAT DE

L'ÉCOLE NATIONALE SUPÉRIEURE
MINES-TÉLÉCOM ATLANTIQUE BRETAGNE
PAYS-DE-LA-LOIRE - IMT ATLANTIQUE

ÉCOLE DOCTORALE N° 648
Sciences pour l'Ingénieur et le Numérique
Spécialité : *sciences et technologies de l'information et de la communication*

Par
Victor MARTIN-LAC

Aerial Navigation Based on SAR Imaging and Reference Geospatial Data

Thèse présentée et soutenue à IMT Atlantique, Brest, le 30 avril 2024
Unité de recherche : LabSTICC, CNRS, UMR 6285 et Thales Land and Air Systems France
Thèse N° : 2024IMTA0400

Rapporteurs avant soutenance :

Laurent FERRO-CAMIL	Professeur, ISAE-Supaero
Hélène ORIOT	Directrice de recherche, ONERA

Composition du Jury :

Président :	M. Ali KHENCHAF	Professeur, ENSTA Bretagne
Directeur de thèse :	M. Jean-Marc LE CAILLEC	Professeur, IMT Atlantique
Examinateurs :	M. Ayoub KARINE	Maitre de conférence, ISEN-Ouest
	M. Laurent FERRO-FAMIL	Professeur, ISAE-SUPAERO
	Mme Hélène ORIOT	Directeur de recherche, ONERA
	M. Jacques PETIT-FRÈRE	Docteur et ingénieur de recherche, Thales LAS

Acknowledgement

This work has benefited from the counselling, review and support from Jacques Petit-Frère, Jean-Marc Le Caillec, Thomas Merlet, Cécile Baudry, Cécile Chenot, Frédéric Jacquelin and Joan Broussole. My parents, my sister and my grand-parents brought up the author of this work or where present for him. Researchers from IMT Atlantique Frédéric Maussang et Christophe Sintès let part of this work serve as practice work. The knowledge gained by the author during his previous work experiences contributed to this work. Among my colleagues were Abdelmalek Nigro, Thibaud Kloczko, Benjamin Rollet, Boris Lenseigne, Marie Lasserre, Matthieu Baudry, Roland Brochard, Amsha Proag, Matthieu Le Goff and Grégory Jonniaux. Many teachers from primary, secondary and higher education passed on some of their arts to the author. This work is the fruit of their efforts. They include Jean-Charles Passieux, Robin Bouclier and all my other teachers.

Table of Contents

Acronyms **8**

Résumé en français **13**

Introduction **21**

1 Navigation based on SAR imaging **23**
1.1 Related topics . . . 23
1.1.1 Themes . . . 23
1.1.2 Inertial navigation . . . 24
1.1.3 Radar imaging . . . 26
1.2 Problem statement . . . 31
1.2.1 Objective . . . 31
1.2.2 Easy instanciation to a specific geographic location . . . 32
1.2.3 Lack of a prior observation by the device . . . 33
1.2.4 Challenges of interpreting ground clutter . . . 33
1.2.5 Limited time, energy and computational means . . . 33
1.2.6 Unavailability of synthetic observations . . . 34
1.2.7 What is a good solution? . . . 34
1.2.8 Other applications . . . 35
1.3 State of the art . . . 35
1.3.1 Aerial navigation based on radar imaging of the ground . . . 35
1.3.2 Aerial navigation based on observing the ground . . . 37
1.3.3 Pose estimation . . . 37
1.3.4 Radar-based automotive navigation . . . 38
1.3.5 Multimodal geospatial data alignment . . . 39
1.3.6 Remote sensing content-based image retrieval . . . 40
1.3.7 Bridging the modality gap . . . 41
1.4 Contributions . . . 41
1.4.1 Kinematic state estimation by aligning contours between distance/Doppler geometry and ground geometry . . . 42
1.4.2 Kinematic state estimation relying on a multimodal similarity metric 42
1.4.3 Adaptiveness to various geospatial data . . . 43
1.4.4 State estimation by scene coordinates regression . . . 43
1.5 Data . . . 44
1.5.1 What is a geospatial product? . . . 44
1.5.2 Multimodal stacks . . . 44

1.5.3 Modalities . . . 46
1.5.4 Preparation . . . 48
1.6 Conclusion . . . 49

2 State estimation by matching contours 51
2.1 Geometric background . . . 51
2.1.1 Definitions . . . 51
2.1.2 Projection from space to image coordinates . . . 52
2.1.3 Transformation between different coordinate systems . . . 54
2.1.4 Pseudo image coordinates . . . 57
2.1.5 Illustration . . . 57
2.2 Determining the kinematic state with known correspondences . . . 61
2.3 Determining the kinematic state without known correspondences . . . 63
2.4 Experiments with iterative closest point . . . 64
2.4.1 Protocol . . . 64
2.4.2 Implementation . . . 65
2.4.3 Data . . . 65
2.4.4 Results and interpretation . . . 66
2.5 Discussion . . . 66
2.6 Complete estimation system . . . 74
2.6.1 Overview of the estimation procedure . . . 74
2.6.2 Training to detect contours . . . 75
2.6.3 Building the navigation data . . . 77
2.6.4 Navigation . . . 77
2.6.5 Results . . . 78
2.7 Conclusion . . . 85

3 State estimation based on a multimodal similarity metric 87
3.1 Local descriptors . . . 87
3.1.1 Description procedure . . . 87
3.1.2 Constructing raster image patches . . . 88
3.1.3 Handcrafted descriptors . . . 89
3.1.4 Learned descriptors . . . 91
3.2 Description models training and evaluation . . . 96
3.2.1 Raster patch construction processes . . . 96
3.2.2 Performance metrics . . . 96
3.2.3 Performance of description models . . . 97
3.3 Estimating a translation . . . 98
3.3.1 Method . . . 98
3.3.2 Relation to image retrieval . . . 100
3.3.3 Experiments . . . 100
3.4 Estimating the full kinematic state . . . 106
3.4.1 Multimodal correlation . . . 106

3.4.2 Kinematic parameters . . . 111
3.4.3 Computational affordability . . . 113
3.4.4 Specifications of an operational system . . . 115
3.4.5 The navigation data . . . 116
3.4.6 Compiling the navigation data . . . 118
3.4.7 Navigating . . . 120
3.4.8 Theoretical discussion . . . 120
3.4.9 Implementation . . . 121
3.4.10 Experiments . . . 122
3.5 Conclusion . . . 124

4 State estimation based on scene coordinates regression 131
4.1 Scene coordinates regression . . . 131
4.2 Proposed method . . . 131
4.2.1 Generating training image patches . . . 132
4.2.2 Regression model architecture and training . . . 133
4.2.3 Kinematic state estimation . . . 133
4.3 Experiments . . . 136
4.4 Towards multimodal SCR . . . 138
4.4.1 Directions to investigate . . . 138
4.4.2 What does not work . . . 139
4.5 Conclusion . . . 140

Conclusion 149

A SAR processing with an imprecise motion estimate 151
A.1 Sources of image defects . . . 151
A.2 Illustrative experiment . . . 152
A.2.1 Signal simulation . . . 152
A.2.2 Focalization . . . 154
A.2.3 Autofocusing . . . 158
A.3 Conclusion . . . 160

B Computer vision on SAR images 161
B.1 Intensity statistics . . . 161
B.2 Despeckling . . . 163
B.2.1 Methods . . . 163
B.2.2 Comparison . . . 166
B.3 Contour detection . . . 170
B.3.1 Ratio of averages . . . 170
B.3.2 Phase congruency . . . 171
B.3.3 Other methods . . . 174
B.4 Segmentation . . . 177

B.4.1 Model-based segmentation . . . 177
B.4.2 Segmentation based on handcrafted features . . . 177
B.4.3 Segmentation based on deep learning . . . 179
B.5 Object detection . . . 181
B.6 Popular datasets . . . 181

C Object detection using a cascade of regressions **183**
C.1 Generic detection algorithm . . . 183
C.2 Detecting buildings . . . 186
C.3 Experiments . . . 186
C.4 Conclusion . . . 188

Bibliography **193**

Acronyms

APR Absolute Pose Regression.
ATD Automatic Target Detection.
ATR Automatic Target Recognition.

BOVW Bag of Visual Words.
BP Back-projection.

CBIR Content-Based Image Retrieval.
CD Closest Distance.
CFAR Constant False Alarm Rate.
CPI Coherent Processing Interval.
CPR Cascaded Pose Regression.
CPU Central Processing Unit.
CV Computer Vision.
CW Continuous Wave.

DEM Digital Elevation Model.
DOP Digitale Orthophotos.
DSMAC Digital Scene Matching Area Correlator.
DTM Digital Terrain Model.

ECEF Earth Centered Earth Fixed.
ENU East North Up.
ES Enhanced Standard.
EU European Union.
EW Electronic Warfare.

FFT Fast Fourier Transform.
FMCW Frequency Modulated Continuous Wave.

GA Genetic Algorithm.
GAN Generative Adversarial Models.
GIS Geographic Information System.

GNC Guidance, Navigation and Control.
GNSS Global Navigation by Satellite System.
GPS Global Positioning System.
GPU Graphics Processing Unit.
GSD Ground Sampling Distance.

HOG Histogram of Oriented Gradient.

ICP Iterative Closest Point.
ID Instantaneous Distance.
IDF Instantaneous Doppler Frequency.
IMU Inertial Measurement Unit.
INRIA Institut National de Recherche en Informatique et en Automatique.
INS Inertial Navigation System.
InSAR Interferometric Synthetic Aperture Radar.
ISTAR Intelligence, Surveillance, Target Acquisition and Reconnaissance.

JPL Jet Propulsion Laboratory.

KOMPSAT KOrean Multi-Purpose SATellite.

LAS Land and Air Systems.
LFM Linear Frequency Modulated.

MALE Medium Altitude Long Endurance.
MI Mutual Information.
MTI Moving Target Indicator.

NASA National Aeronautics and Space Administration.
NCC Normalized Cross-Correlation.
NED North East Down.
NNI Natural Neighbor Interpolation.

ONERA Office National d'Études et de Recherches Aérospatiales.
OSM OpenStreetMap.

PCA Principal Components Analysis.
PGA Phase Gradient Autofocus.
PRF Pulse Repetition Frequency.

RAM Random Access Memory.
RANSAC RANdom SAmple Consensus.
RAR Real Aperture Radar.
RD Range Doppler.
RDA Range Doppler Algorithm.
RF Radio Frequency.
RGB Red Green Blue.
RGB-D Red Green Blue Depth.
RSO Radar à Synthèse d'Ouverture.

SAR Synthetic Aperture Radar.
SCR Scene Coordinate Regression.
SGD Stochastic Gradient Descent.
SIFT Scale Invariant Feature Transform.
SLAM Simultaneous Localization and Map Building.
SLIC Simple Linear Iterative Clustering.
SVD Singular Value Decomposition.

TCA Time of Closest Approach.
TDC Time Domain Correlation.
TERCOM Terrain Contour Matching.

UAV Unmanned Aerial Vehicle.
UK United Kingdom.
UTM Universal Transverse Mercator.

VLAD Vector of Locally Aggregated Descriptors.

WYSIWYG What You See Is What You Get.

Résumé en français

L'augmentation contemporaine des performances des calculateurs et des capteurs embarqués ainsi que les progrès des traitements algorithmiques d'analyse de données et de prise de décision rendent possible la construction de dispositifs dotés d'un plus haut niveau d'autonomie. Ces dispositifs peuvent extraire automatiquement des informations utiles mais diluées, discerner des formes complexes et finalement opérer plus intelligemment dans des environnements peu structurés à partir de signes moins faciles à interpréter. Ces progrès généraux trouvent application en robotique spatiale, aérienne, marine et terrestre. Dans le secteur de la défense, ces progrès peuvent engendrer une extension du domaine opérationnel de nombreux systèmes d'armes ou la couverture de nouvelles fonctions opérationnelles. Ainsi, des appareils tels que des missiles et des drones pourraient bénéficier d'une autonomie accrue en environnements moins structurés et plus récalcitrants. Les présents travaux s'inscrivent dans l'effort général de développement des algorithmes nécessaires à des systèmes d'arme plus autonomes et concernent spécifiquement la navigation de missiles et de drones.

De même que tout système de contrôle en boucle fermée inclut un sous-système dédié à l'estimation de l'état du système contrôlé, le dispositif guidant un missile ou un drone jusqu'à un point géographique cible contient un sous-système dédié à l'estimation de l'état cinématique du missile. Dans ces travaux, nous considérons que l'état cinématique du missile est constitué de sa position et de sa vitesse relativement à la terre. L'estimation de l'état cinématique d'un missile repose sur des mesures physiques (inertielles ou électromagnétiques) ou par exemple sur un système de navigation par satellite. Nous considérons un contexte où l'environnement de l'appareil à guider est peu structuré et dépourvu d'infrastructure dédiée spécifiquement à la navigation de l'appareil. Le système d'estimation de l'état cinématique repose donc seulement sur des mesures physiques brutes, sans interaction avec une infrastructure dédiée.

Des mesures inertielles, issues d'un gyroscope et d'un accéléromètre, seraient avantageuses dans notre contexte car elles dépendent très peu de l'environnement extérieur à l'appareil. Elles semblent ainsi compatibles avec notre objectif d'autonomie. Cependant, l'estimation inertielle de l'état cinématique repose sur l'intégration de mesures bruitées à partir d'un état connu, si bien que la précision de l'estimation de l'état cinématique décroit irrémédiablement au cours du temps. Afin d'obtenir un système d'estimation dont la précision ne décroit pas asymptotiquement, les mesures inertielles doivent donc être périodiquement jointes à des mesures qui portent directement non pas sur la variation de l'état cinématique mais sur l'état cinématique lui-même. Comme complément aux mesures inertielles, les présents travaux considèrent l'estimation de l'état cinématique de l'appareil par observation radar du sol.

Reconnaître des formes dans l'observation radar brute du sol, c'est-à-dire le fouilli de

sol, est difficile étant donné notamment la très faible résolution spatiale de l'image non focalisée, proportionnelle à la distance entre l'appareil et le sol et de l'ordre de la centaine de mètres. Or l'estimation de la position et de la vitesse de l'appareil repose sur la reconnaissance de formes dans l'observation radar. Ainsi, pour déduire l'état cinématique à partir de l'observation radar, il serait pertinent de focaliser celle-ci en une image Radar à Synthèse d'Ouverture (RSO). L'imagerie RSO désigne la construction de carte haute résolution de la réflexivité radar du sol par traitement du signal radar. Elle repose premièrement sur la collecte d'échos radar selon des points de vue variés relativement au sol, sous l'hypothèse d'une scène imagée statique et de géométrie connue (par exemple plane). L'imagerie RSO repose deuxièmement sur la comparaison, pour chaque cellule de résolution, entre les échos collectés et les échos théoriques prédits pour cette cellule de résolution seule. Le système RSO est dimensionné de sorte que les échos associés à des cellules de résolution différentes soient orthogonaux. Nous admettons que l'image RSO, de résolution métrique voire centimétrique, est plus facilement exploitable que l'image non focalisée pour inférer algorithmiquement l'état cinématique. Certaines propriétés des mêmes images RSO gènent toutefois la reconnaissance de forme donc l'inférence de l'état cinématique. Les images RSO présentent par exemple un fort bruit multiplicatif, appelé chatoiement. Les images RSO peuvent être géométriquement déformées si les hypothèse concernant la géométrie du sol sont fausses (par exemple si le sol présente du relief mais que l'algorithme de focalisation suppose un sol plan). Les éléments verticaux de la scène observée, comme les bâtiments, apparaissent par exemple repliés. Synthétiser des images RSO réalistes par simulation et avec une charge de calcul admissible est hors de l'état de l'art, ce qui restreint le champ des solutions algorithmiques envisageables. Malgré les propriétés adverses précédemment énumérées, l'imagerie RSO présente les avantages d'une résolution indépendante de la distance entre l'antenne et le sol et d'une disponibilité en conditions nuageuses ou nocturnes.

Une autre caractéristique de l'imagerie RSO importante dans notre contexte est que la construction d'une image RSO requiert des hypothèses concernant la géométrie de la scène observée et la trajectoire de l'antenne relativement à la scène durant la collecte des échos. En effet, le modèle théorique de l'écho associé à une cellule de résolution dépend de ces deux données. Ainsi, pour un sol plan et une trajectoire rectiligne uniforme, il faut au moins connaître la norme du vecteur vitesse. L'obtention de l'image RSO dépend donc partiellement de l'état cinématique recherché. Néanmoins, des techniques d'autofocus permettent de focaliser l'observation radar malgré une connaissance imprécise du mouvement de l'antenne. Les méthodes d'autofocus paramétriques permettent également d'estimer une partie des mouvements de l'antenne par optimisation d'un critère de qualité de focalisation. Dans le cadre des présents travaux, nous considérons un sol plan. Nous supposons disposer du système de focalisation et n'étudions pas la formation de l'image RSO. Nous étudions l'inférence de l'état cinématique à partir de celle-ci, par des méthodes d'analyse d'image et de vision par ordinateur.

La détermination de l'état cinématique repose non seulement sur une observation de l'environnement de l'appareil, en l'espèce une observation radar focalisée, mais égale-

ment sur des connaissances a priori concernant cet environnement. Afin que le système d'estimation de l'état cinématique puisse être appliqué facilement au lieu géographique dicté par la mission, ces connaissances concernant le lieu de la mission doivent être une entrée du système, que nous appelons données géospatiales de référence. Par ailleurs, généralement, un capteur du même modèle que celui utilisé pour la mission ne survolera pas le lieu de la mission avant la mission, si bien que nous ne disposons pas d'une donnée géospatiale de référence de même modalité que l'observation. En d'autres terme, l'observation et la référence sont hétérogènes. Elles proviennent de différents capteurs ou de différents processus de formation. Une même structure physique apparaît donc différemment dans l'observation et dans la référence, ce qui gène substantiellement l'interprétation conjointe de la donnée de référence et de l'observation. Toutefois, pour faciliter ce transfert, nous supposons que nous disposons de couples de données géospatiales, selon la modalité d'observation et la modalité de référence, associés à d'autres lieux que le lieu de mission. Ces couples sont autant d'exemples de formes correspondantes, de mêmes structures vues dans les deux modalités, en des lieux servant d'exemple.

Nous avons considéré et utilisé pour nos expériences des données de référence RSO, optiques et vectorielles. Nos expériences s'appuient sur des données RSO Miranda (imageur RSO aéroporté FMCW développé par *Fraunhofer Institute for High Frequency Physics*), des données RSO KOMPSAT-5 (imageur RSO satellite développé par la Corée du Sud), des données RSO I-Master (imageur RSO aéroporté développé par Thales UK), des données RSO Capella Aerospace (imagerie RSO satellite de résolution sol sous-métrique), des données optiques Pléiades (système d'imagerie optique satellite de résolution métrique), des données optiques DOP (couverture optique aérienne de Allemagne), des données optiques correspondant au fond de carte du système d'information géographique Microsoft Bing et des données vectorielles OpenStreetMap (base de données géographiques vectorielles mondiale et ouverte).

Nous réduisons la recherche de l'état cinématique à un ajustement de coordonnées entre le système de coordonnées dans lequel est définie l'observation RSO et celui dans lequel est définie la référence. En d'autres termes, nous recherchons une transformation de coordonnées, entre l'observation RSO et la référence, dont les paramètres sont l'état cinématique. La donnée géospatiale de référence est définie dans un repère local tangent cartésien, tel qu'un repère East North Up (ENU). Nous considérons deux systèmes de coordonnées pour l'observation RSO. Avant de définir les coordonnées qui les composent, notons que l'état cinématique recherché, identifié à une position et un vecteur vitesse, représente aussi une certaine trajectoire rectiligne uniforme relativement à laquelle chaque point de l'image RSO ou du sol peuvent être identifiés. Ainsi, Le premier système de coordonnées comprend le temps d'approche proximal (l'instant de la trajectoire rectiligne uniforme où l'appareil serait le plus proche du point géographique considéré) et la distance proximale (la distance à cet instant). Le second système comprend la fréquence Doppler instantanée (la fréquence Doppler du point géographique à l'instant initial de la trajectoire rectiligne uniforme) et la distance instantanée (la distance au point géographique considéré, à l'instant initial de la trajectoire rectiligne uniforme).

Des méthodes d'alignement d'image et d'estimation de pose inspirées de la vision par ordinateur classique peuvent être appliquées à la détermination de l'état cinématique, réduite à l'ajustement d'une transformation de coordonnées entre des données géospatiales hétérogènes. Une telle méthode consiste par exemple en la détection d'objets séparément dans chaque donnée, leur appariement et l'ajustement d'une transformation géométrique entre les coordonnées appariées. Une seconde méthode est la recherche d'une transformation géométrique maximisant la ressemblance entre une donnée et l'autre donnée géométriquement transformée. Enfin une troisième méthode consiste à prédire, à l'aide d'un modèle appris, les coordonnées de scène de chaque point de l'observation puis à ajuster une transformation entre paires de coordonnées relatives à la scène et de coordonnées relative à l'appareil. Nous documentons dans ce mémoire les travaux originaux suivants dans le cadre de la problématique exposée précédemment.

Nous montrons premièrement comment déterminer la transformation de coordonnées reliant observation et référence, c'est-à-dire comment déterminer l'état cinématique, sachant trois correspondances point-à-point entre les deux domaines, c'est-à-dire à partir du plus petit nombre de correspondances. Nous présentons un procédé de calcul direct (i.e. non itératif). Nous décrivons aussi comment déterminer la transformation de coordonnées sachant un ensemble de correspondances point-à-point, en résolvant numériquement un problème de maximum a posteriori (en l'espèce un problème de moindres carrés non-linéaire). Finalement, nous montrons comment déterminer la transformation de coordonnées à partir d'une correspondance ensemble-à-ensemble. Ceci repose sur l'algorithme Iterative Closest Point (ICP), typiquement utilisé pour estimer une transformation rigide tridimensionnelle entre deux nuages de points. Cette méthode itérative, de type descente par bloc, consiste à estimer alternativement les correspondances point-à-point les plus vraisemblables sachant un état cinématique puis l'état cinématique sachant un ensemble de correspondances point-à-point. Nous illustrons ces méthodes de calcul par des expérimentations. Nous dessinons manuellement des structures identiques dans l'observation RSO (en l'espèce des images issues de I-Master) et dans la référence (en l'espèce un fond de carte optique Microsoft Bing) puis nous estimons par Monte-Carlo l'erreur d'estimation. À partir d'une estimation initiale égale à la vérité-terrain plus une erreur aléatoire, l'algorithme ICP produit une estimation a posteriori de l'état cinématique. Sur nos données de test, malgré une erreur initiale uniforme d'amplitude 1 km en position horizontale, 200 m en altitude, 10° en angles du vecteur vitesse et 10 m/s en vitesse, l'algorithme converge avec une probabilité d'environ 90 % au voisinage de la solution. Ces travaux concernant l'ajustement d'une transformation de coordonnées sont présentés dans les chapitres 2 et 4.

La mesure de la similarité entre le voisinage d'un point de l'observation RSO et le voisinage d'un point de la référence est génée par l'hétérogénéité de celles-ci. Or, elle est nécessaire aux méthodes d'alignement d'image directes, basées sur la recherche de la transformation de coordonnées maximisant la ressemblance entre une donnée et l'autre donnée transformée géométriquement. Pour mesurer cette similarité, nous avons considéré d'une part une détection de contours multimodale et d'autre part des descripteurs

multimodaux.

La détection de contours multimodale que nous avons investiguée consiste à entraîner un algorithme (en l'espèce un réseau de neurones) à prédire à partir d'une image de la modalité de référence (e.g. optique) les contours qui seraient détectés par un algorithme classique sur une image de la modalité d'observation (i.e. RSO) du même lieu. Plus précisément, nous prédisons une carte de distance aux contours. Elle associe à chaque point la distance au contour le plus proche. En effet, la carte de distance aux contours est la donnée exploitée directement par les méthodes d'alignement de type chamfrein (*chamfer matching*) ou l'optimisation de la distance moyenne entre les contours transformés depuis une modalité et les contours de l'autre modalité. Un algorithme de détection de contour classique, tel que le ratio des moyennes (voir section B.3), peut être utilisé pour la modalité RSO. Nous avons mené des expériences basées sur cette approche. Nous avons obtenu l'estimation d'une translation entre une observation RSO (issue de I-Master) et une référence optique (issue du fond de carte de Microsoft Bing) s'appuyant sur cette méthode. Cependant, l'estimation de l'état cinématique complet échoue, étant données l'existence de minima locaux et à la difficulté pratique d'échantillonner suffisamment densément l'espace des états cinématiques, de dimension six. Ces travaux sont décrits dans le chapitre 2.

Nous avons investigué l'utilisation de descripteurs multimodaux pour mesurer la similarité entre le voisinage d'un point de l'observation et le voisinage d'un point de la référence. En vision par ordinateur, un descripteur est une représentation d'un objet adaptée à une certaine tâche. En l'espèce, la tâche que nous souhaitons résoudre est l'identification de correspondances entre l'observation et la référence malgré leur différentes modalités. L'objet à décrire est la restriction de la référence ou de l'observation au voisinage d'un point. La similarité entre deux voisinages est définie comme la similarité de leur représentations intermédiaires respectives appelées descripteurs. Les descripteurs décrits dans l'état de l'art et fréquemment employés dans des applications prennent la forme d'une collection de nombre réels ou de chiffres binaires. La similarité de deux descripteurs correspond quant à elle typiquement à la distance euclidienne, au nombre de chiffres binaires différents voire à un réseau de neurones. L'algorithme de construction d'un descripteur à partir de l'objet original peut être issu d'ingénierie humaine ou appris. Les descripteurs issus d'ingénierie humaine et destinés à l'alignement d'image multimodal reposent fréquemment sur des histogrammes d'orientation de contours ou sur l'auto-similarité (voir chapitre 3). Les descripteurs que nous avons développés reposent sur de l'apprentissage. Un réseau de neurone est entraîné à prédire si deux voisinages, de modalités différentes, représentent ou non le même lieu physique. Ce réseau est composé de deux branches, selon l'architecture pseudo-siamoise. Chaque branche représente le calcul d'un descripteur à partir d'un voisinage. Les deux branches sont entraînées conjointement. Seulement des descripteurs issus de branches entraînées conjointement peuvent être validement comparés. Les descripteurs appris n'ont généralement pas d'interprétation ni de signification évidente. Les procédés d'entraînement rapportés dans l'état de l'art ou correspondant à nos descripteurs sont décrits dans le chapitre 3. Nous y rapportons égale-

ment nos résultats expérimentaux liés à l'entraînement des descripteurs et à l'utilisation de ces derniers pour l'alignement. Nous entraînons, en validation croisée, des descripteurs correspondant à huit combinaisons de modalités parmi Miranda (RSO), I-Master (RSO), KOMPSAT-5 (RSO), Pléiades (optique), DOP (optique), Microsoft Bing (optique) et OpenStreetMap (vectoriel). Au total, nous entraînons 44 réseaux. Nous menons plusieurs centaînes d'expériences d'alignement consistant à estimer une translation entre une observation couvrant 0,64 km^2 (800 m par 800 m) et une référence couvrant 225 km^2 (15 km par 15 km). Sur un ordinateur de bureau, le temps de calcul pour estimer une translation est de l'ordre de deux secondes. Ce temps n'inclut pas l'entraînement des descripteurs ni l'échantillonnage de descripteurs sur la donnée géospatiale de référence, ces tâches étant effectuées en phase de préparation mission. Malgré l'écart de modalité et l'asymétrie entre l'aire couverte par l'observation et celle couverte par la référence, l'estimation de translation est très robuste, avec, pour la plupart des combinaisons de modalités, 50 % de probabilité d'une erreure inférieure à 5 m et 95 % à 20 m. L'alignement multimodal basé sur des descripteurs multimodaux est beaucoup plus robuste que celui reposant sur des algorithmes classiques issus d'ingénierie humaine, tels que l'information mutuelle et le coefficient de corrélation linéaire. Les descripteurs mis en œuvre dans nos expériences sont en virgule flottante. Des descripteurs binaires présenteraient un meilleur temps de calcul et une meilleure compacité. Nous avons investigué l'estimation de l'état cinématique complet basée sur des descripteurs multimodaux. Nous avons réduit le nombre d'extraction de descripteurs en tirant parti de la décomposition de la transformation reliant observation et référence en une translation et une autre transformation. Cependant, le temps de calcul reste prohibitif, au moins pour des descripteurs en virgule flottante. Tous nos travaux basés sur des descripteurs multimodaux sont décrits dans le chapitre 3.

Nous avons investigué l'estimation de l'état cinématique basée sur la régression directe des coordonnées géographiques de voisinages de points de l'observation. Dans l'état de l'art de la vision par ordinateur générale, cette approche est appelée régression de coordonnées de scène. Elle est classiquement appliquée à l'estimation de la pose d'une caméra. Elle consiste à choisir arbitrairement des points dans l'observation, à en extraire les voisinages puis à prédire les coordonnées des points de la référence correspondant. À partir des correspondances entre les coordonnées dans l'observation et les coordonnées dans la référence, nous estimons l'état cinématique. Une difficulté de cette approche, que nous n'avons pas surmontée, est l'entraînement du modèle de régression qui prédit les coordonnées de référence à partir du voisinage d'un point de l'observation. En effet, dans notre cas, nos seules données d'entraînement sont une donnée géospatiale de référence, dont la modalité diffère de la modalité de l'observation utilisée en inférence, et des piles multimodales associées à d'autres lieux géographiques. L'état de l'art existant de l'apprentissage par transfert ne comprend pas de solution à cette tâche. Malgré son inaccessibilité dans l'état de l'art actuel, l'estimation d'état cinématique basée sur la régression de coordonnées de scène laisse envisager d'excellentes propriétés calculatoires. En effet, le volume de calcul et de données en inférence serait au premier ordre indépendant du volume et de la dimension de l'espace des états cinématiques initiaux.

Nous rapportons des résultats expérimentaux illustrant la chaîne algorithmique mais en utilisant l'observation en entraînement. Ces travaux sont documentés dans le chapitre 4.

La suite du mémoire est organisée ainsi. Le chapitre 1 définit la problématique et les éléments de l'état de l'art liés à celle-ci. Nous décrivons également en détail les données géospatiales utilisées dans les expérimentations. Le chapitre 2 décrit les systèmes de coordonnées de l'observation RSO et des données géospatiales de référence. Il décrit l'estimation de la transformation entre les systèmes de coordonnées, par moindres carrés non-linéaires ou par l'algorithme ICP. Nous décrivons également la détection de contours multimodale et son application à l'estimation de l'état cinématique. Le chapitre 3 décrit l'estimation de l'état cinématique basée sur des descripteurs multimodaux. Nous y rapportons les éléments de l'état de l'art concernant les descripteurs multimodaux issus d'ingénierie humaine ou appris. Nous décrivons les descripteurs multimodaux mis en œuvre dans nos expérimentations. Ceci inclut leur architecture, les données d'entraînement, les paramètres de l'entraînement et l'évaluation en validation croisée. Nous décrivons l'utilisation des descripteurs multimodaux pour estimer une translation et nous présentons nos travaux concernant leur utilisation pour estimer l'état cinématique complet. Dans le chapitre 4, nous décrivons l'approche basée sur la régression de coordonnées de scène. Nous décrivons les étapes de cette méthode et rapportons nos résultats expérimentaux. L'annexe A illustre la construction de l'image RSO malgré un déplacement de l'antenne inconnu. Nous décrivons les bases de la focalisation RSO ainsi qu'un algorithme d'autofocus primitif. L'annexe B présente les différences entre le traitement d'image classique, généralement optique, et le traitement d'image RSO. En effet, les images RSO présentent des particularités qui doivent être prises en compte lors de la conception d'un algorithme de traitement d'image RSO. Nous présentons premièrement les modèles statistiques du chatoiement, le bruit multiplicatif qui contamine les images RSO et qui implique des traitements spécifiques. Nous présentons les méthodes de débruitage, c'est-à-dire de construction d'une image lisse par morceau, exempte de chatoiement, à partir d'une image RSO. Nous décrivons les algorithmes classiques de détection de contours dans une image contaminée par du chatoiement. Enfin, nous discutons de la segmentation sémantique et de la détection d'objet dans des images RSO. Finalement, l'annexe C rapporte des travaux que nous avons menés sur la détection de bâtiment dans une image RSO. Nous présentons un algorithme inspiré de la régression de pose en cascade, une technique de vision par ordinateur typiquement utilisée pour ajuster un modèle géométrique sur une image. Nous présentons également les résultats expérimentaux obtenus sur des images issues de Miranda.

INTRODUCTION

We can integrate ever more computational power and better sensors in our devices. Given the right algorithms, we may consequently design systems that operate in less structured environments with a higher degree of autonomy. A particular ability to be unlocked is the self-localization of aerial devices, such as aircrafts, Unmanned Aerial Vehicle (UAV) and missiles, in a non-cooperative environment. In this thesis, we investigate the algorithmic means of autonomously estimating of the position and velocity of an aerial device based on Synthetic Aperture Radar (SAR) imaging of the ground.

The reliance on SAR imaging for the self-localization of an aerial device is poorly documented in the scientific literature. The use of optical imaging is dominant. Optical sensors are generally more affordable and easier to come by than imaging radars. In contrast, we may not recognize much useful pattern from a radar observation of the ground without focusing it, which requires sophisticated and computationally costly processings, especially when the motion of the antenna relative to the ground is incompletely or inaccurately known. However, a self-localization system based on SAR imaging may serve the purpose of autonomous navigation or driver assistance in an operational domain defined by radar imaging only (as far as sensing is concerned), which notably implies independently of cloud cover, of natural illumination and of the availability of a Global Navigation by Satellite System (GNSS). The resolution of SAR images is also independent of range. In summary, overcoming the technical challenge associated with constructing a SAR based localization system would unlock new mission settings that fall outside the operational domain of systems that rely on optical imaging.

A localization system must integrate knowledge about the geographic location where it is supposed to operate. The ease of deploying the localization system to a given location requires that that knowledge take the form of a standalone piece of geospatial data (e.g. a remote sensing image, a geographic information system). However, for reasons to be developed in chapter 1, the reference and the observation must have different natures in general, which complicates the discovery and matching of co-occurring patterns. Relating the reference and the observation despite their different natures lies at the core of this thesis.

In this work, we focus primarily on the computer vision algorithms needed for SAR-based self-localization. We deal with the signal processing aspects as much as required for justifying our hypotheses.

In chapter 1, we explain in more details what we are trying to achieve in this thesis and we formally define the characteristics of the desired algorithm. We review the elements of the technical and scientific state of art related to our purpose. We describe the data on which our experiments are based.

In chapter 2, we explore a first approach towards estimating the kinematic state of

the aerial device. We begin the chapter by defining several coordinate systems involved in SAR imaging. We show how to determine the kinematic state given point-to-point correspondences between two coordinate systems. We consecutively show how to determine the kinematic state given a set-to-set correspondence instead of point-to-point correspondences. The algorithm that we introduce for this purpose is derived from Iterative Closest Point (ICP) in the sense that we alternatively estimate the kinematic state given point-to-point correspondences and point-to-point correspondences given the kinematic state. However, instead of fitting a rigid transform, we fit a class of coordinate transforms whose parameters are the kinematic state. We experimentally validate this algorithm. In a second part, we show that we can train a neural network to predict, from an optical image, the distance map of contours found on a SAR image of the same place. Building upon this result, we investigate a complete estimation pipeline, one that comprises multimodal contour detection on the reference and observation geospatial data.

In chapter 3, we explore a second approach towards estimating the kinematic state of the aerial device. We consider using a multimodal similarity metric, in the form of local descriptors. They are the means of measuring the likelihood that a location within the reference geospatial data and a location within the observation geospatial data represent the same physical location. We first provide an overview of the literature on this subject, considering both handcrafted and learned local multimodal descriptors. We then describe how to reliably estimate a translation and illustrate it with extensive experiments. Afterwards, we consider how to estimate the full kinematic state using local descriptors. We propose a method and examine it experimentally.

In chapter 4, we explore a third approach towards estimating the kinematic state of the aerial device. It is inspired from Scene Coordinate Regression (SCR). We consider directly learning to predict geographic coordinates from patches extracted from the observation SAR image. One can then infer the kinematic state from the predicted corresponding observation and reference coordinates. However, this approach is restricted to the particular and unlikely case of identical reference and observation modalities. We report experimental results. We finally propose directions of investigation towards a multimodal SCR.

In the conclusion, we summarize our work and propose new directions of investigation.

We denote vectors with bold lowercase letters and matrices with bold uppercase letters.

Chapter 1

Navigation based on SAR imaging

We start this first chapter by defining the notions that are required for stating the problem on which our work is focused. In particular, we develop the topics of inertial navigation and radar imaging. Secondly, we describe the technical and scientific question that this work is about. We introduce the problem to solve and we detail all the constraints that a solution to this problem has to satisfy. We explain the implications of these constraints on the solutions. Thirdly, we review the scientific state of the art related to our problem. We review all the existing developments that may be involved in the formulation of a solution to our problem. Fourthly, we describe the contributions of this thesis to the state of the art. Finally, we describe all the data that the experiments presented in this thesis rely upon.

1.1 Related topics

1.1.1 Themes

Before introducing the problem that we are dealing with, let us remind some related concepts.

Autonomous navigation Autonomous navigation is the driving of a platform without the need of an human pilot. An unstructured environment is an environment that is not equiped with devices whose purpose is to help the autonomous navigation, so that the autonomous navigation system is entirely contained within the platform.

Radar imaging Radar is a technology of remote sensing by which modulated radio-frequency electromagnetic waves are emitted onto an object and information is extracted from the received echoes. Synthetic Aperture Radar (SAR) refers to a radar technique capable of producing a high resolution map of the ground reflectivity.

Guidance, navigation and control The systems whose purpose is to drive vehicles automatically are generally composed of three parts. The navigation subsystem estimates the kinematic state of the vehicle from sensed information. The guidance subsystem plans a desired trajectory for the vehicle. The control subsystem determines the commands to the actuators so that the vehicle follows the desired trajectory.

Missiles and missile seekers A projectile is an object or a device thrown at a target for the purpose of destroying or incapacitating it. A missile is a self-propelled guided projectile. A missile is composed of a payload, a propulsion system, a Guidance, Navigation and Control (GNC) system, actuated deflection surfaces and sensors. Generally, the payload is a destructive charge and the propulsion system is a rocket motor or a ramjet. The GNC system analyzes sensed information, plans a trajectory and realizes it by moving the control surfaces. The missile seeker is the part of the missile that integrates the sensors and the navigation system.

Artificial intelligence Artificial intelligence is perceiving and inferring information by computational means. Thanks to ever more powerful processors and new techniques, we can solve a growing number of cognitive problems algorithmically. The contemporary progress in artificial intelligence induces a progress in the design of navigation systems which can infer the kinematic state of the platform from ever more convoluted sensory information.

1.1.2 Inertial navigation

A motion relative to an inertial frame can be estimated relying upon these physical phenomena:

— The linear acceleration undergone by a body relative to an inertial frame is identifiable with the force that should be applied to an otherwise unforced other body so that the two bodies exhibit no relative motion.

— Using the gyroscopic effect, we can design gimbaled platforms having a constant orientation relative to an inertial frame.

— Using the Sagnac effect, we can mesure rotation rates relative to an inertial frame.

Based on these physical phenomena, we can design instruments for estimating a trajectory relative to an inertial frame. An accelerometer is a device for estimating linear accelerations by measuring the force applied to a test mass by a spring attached to the body. However, since the test mass is also forced by gravity, realizable accelerometers actually measure the linear acceleration minus the local gravity. Their readings are expressed in the frame of the accelerometer. By placing an accelerometer on a gyroscopically stabilized platform and knowing the direction of the vertical relative to this platform, the readings of the accelerometer can be corrected from the local gravity and integrated. The trajectory of the device relative to its initial state can so be estimated. This kind of inertial navigation system is called inertially stabilized. A second kind of inertial navigation system, called strapdown, relies on measuring the rotation rates of the device using the Sagnac or gyroscopic effect. The rotation rate measurements are integrated so that the attitude of the body relative to its initial state is estimated at any time. As well, the accelerometer measurements are transformed from the current to the initial body axes, corrected from the local gravity and integrated.

The platform trajectory, in the inertial navigation frame, in function of the instantaneous physical quantities that strapdown inertial instruments aim to measure, can be

modeled as the following dynamical system.

$$\begin{cases} \dot{\boldsymbol{p}}(t) = \boldsymbol{v}(t) \\ \dot{\boldsymbol{v}}(t) = \boldsymbol{Q}(t)\boldsymbol{\nu}(t) + \boldsymbol{g} \\ \dot{\boldsymbol{Q}}(t) = \boldsymbol{Q}(t)\boldsymbol{\omega}_{\times}(t) \end{cases} \tag{1.1}$$

We denote $\boldsymbol{v}$ the velocity of the body. The mapping between the body and navigation spatial coordinates (respectively $\boldsymbol{x}_{\text{nav}}$ and $\boldsymbol{x}_{\text{body}}$) at time t is identified with $\boldsymbol{p}(t) \in \mathbb{R}^3$ and $\boldsymbol{Q}(t) \in \mathbb{R}^{3\times3}$ which are respectively the position and attitude (i.e. rotation) of the body:

$$\boldsymbol{x}_{\text{nav}} = \boldsymbol{p}(t) + \boldsymbol{Q}(t)\boldsymbol{x}_{\text{body}} \tag{1.2}$$

We denote $\boldsymbol{g} \in \mathbb{R}^3$ the local gravity in the inertial navigation frame. We respectively denote $\boldsymbol{\nu}(t) \in \mathbb{R}^3$ and $\boldsymbol{\omega}(t) \in \mathbb{R}^3$ the accelerometer and gyroscope measurements. Finally, for any $\boldsymbol{a} \in \mathbb{R}^3$, we denote $\boldsymbol{a}_{\times}$ the matrix of the cross product (also called vector product) with $\boldsymbol{a}$. It is the antisymmetric matrix such that $\forall \boldsymbol{b} \in \mathbb{R}^3 : \boldsymbol{a}_{\times}\boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$. The previous system can be written as

$$\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), \boldsymbol{u}(t)) \tag{1.3}$$

with $\boldsymbol{x}(t) = (\boldsymbol{p}(t), \boldsymbol{v}(t), \boldsymbol{Q}(t))$ and $\boldsymbol{u}(t) = (\boldsymbol{\nu}(t), \boldsymbol{\omega}(t))$. Function f is:

$$f((\boldsymbol{p}, \boldsymbol{v}, \boldsymbol{Q}), (\boldsymbol{\nu}, \boldsymbol{\omega})) = \begin{pmatrix} \boldsymbol{v} \\ \boldsymbol{Q}\boldsymbol{\nu} + \boldsymbol{g} \\ \boldsymbol{Q}\boldsymbol{\omega}_{\times} \end{pmatrix} \tag{1.4}$$

Given noisy inertial measurements $\hat{\boldsymbol{u}}(t)$, the strapdown inertial navigation unit obtain the state estimates $\hat{\boldsymbol{x}}(t)$ by integrating the following differential equation.

$$\dot{\hat{\boldsymbol{x}}}(t) = f(\hat{\boldsymbol{x}}(t), \hat{\boldsymbol{u}}(t)) \tag{1.5}$$

Given that the integration performed by an actual Inertial Measurement Unit (IMU) is discrete (finite arithmetic and finite integration rate), given the calibration defects and given the measurement noise, the precision of the kinematic state estimate deteriorates over time. As an illustration, we have sampled trajectory estimates based on equation 1.5. We have introduced a bias on the measurements of $\boldsymbol{\nu}(t)$ and $\boldsymbol{\omega}(t)$ (the bias is randomly chosen for each run) and velocity and attitude noise at each timestep, as reported in table 1.1. These values are representative of a tactical grade IMU. The numerical integration is performed using an Euler scheme, with a timestep of 10^{-4} seconds, with a quaternion parameterization of the rotation matrix. We simulate 100 five minutes trajectories. The error quantiles as a function of time are reported in figure 1.1. The attitude error is the angle between the estimated and actual rotation matrices. As it can be seen, after five minutes, the position error is in the order of hundreds of meters and the attitude error is in the order of a few hundredths of degrees.

In many mission scenarii, the rapid growth of estimation errors prohibits using purely

Parameter	Value	Unit
Linear acceleration bias	$\mathcal{U}(-0.005, 0.005)$	$m.s^{-2}$
Velocity noise density	10^{-3}	$m.s^{-3/2}$
Rotation rate bias	$\mathcal{U}(-0.5, 0.5)$	$deg.hour^{-1}$
Rotation noise density	0.05	$deg.hour^{-1/2}$

Table 1.1 – Parameters of the IMU simulation.

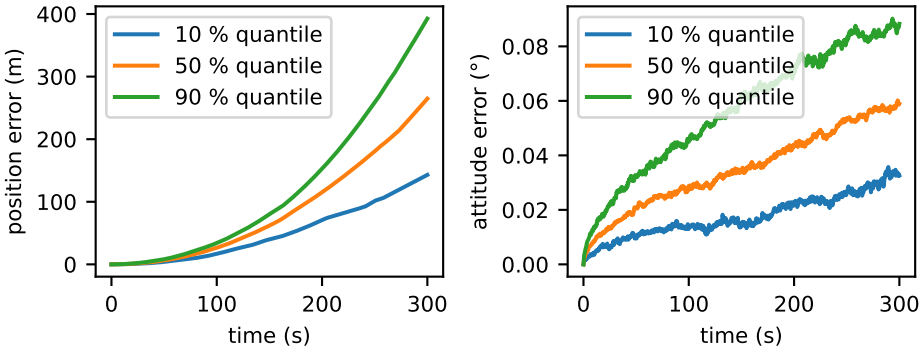


Figure 1.1 – Simple numerical simulation of IMU errors.

inertial navigation. However, inertial navigation can be used in conjunction with another navigation system with complementary properties. For instance, the precision of Global Navigation by Satellite System (GNSS) typically does not deteriorate over time but the estimation rate is lower than that of inertial navigation. As well, sporadic electro-optical and electromagnetic observations can be fused with inertial measurements to reduce the estimation error and calibrate the IMU.

1.1.3 Radar imaging

The variations of the intensity of an electric current generate waves in the electromagnetic field. The shape of an emitted wave, called a waveform, can be engineered so that its echo is easier to recognize. The detection of an echo is typically performed by matched filtering or stretch processing. Matched filtering consists in a convolution with the prototype emitted wave. It is proved that the convolution kernel whereby the signal to noise ratio of the convoluted signal is maximal is the prototype signal itself (modulo time-inversion and conjugation). Stretch filtering involves mixing (i.e. multiplying) the received signal and the emitted signal. In the case of a continuous-wave linear frequency modulated waveform, the amplitude of a frequency component of the mixed signal is proportional to the reflectivity at a range corresponding to that frequency. So spectral analysis of the mixed signal reveals the return amplitude at each range. An advantage of deramping is that the bandwidth of the mixed signal is lower than the bandwidth of the prototype signal, reducing the sampling rate requirements.

Pulse compression is the process of designing a waveform together with a detector for this waveform. When designing a waveform, a concern is the shape of the detector output signal as a function of the scatterer radial velocity. For instance, depending on the particular waveform and detector, the velocity of the scatterer may induce a time offset on the detector output signal. Also, the detector output signal may exhibit higher or lower peak amplitude, smaller or bigger sidelobes. The ambiguity function maps the range and radial velocity of a scatterer to a detector output signal. It subsumes all the pulse compression performance metrics such as the peak to sidelobe ratio, the 3 dB peak width, the integrated sidelobe ratio, the Rayleigh bandwidth, etc. Computing an analytical expression of the detector output signal generally involves integrating complex exponentials. Approximate analytical solutions to these integrals can be obtained using the principle of stationary phase, also called stationary phase approximation, which describes how to neglect the high frequency part of the integrand.

A typical waveform for SAR imaging is the linear frequency modulated signal, which is called a chirp. A notional continuous-wave linear frequency modulated signal is represented in figure 1.3. It is characterized by a comparatively high peak to sidelobe ratio and by (to some extent) an ambiguity between time delay and radial velocity (the ambiguity can be averaged out by considering two pulses with opposite frequency modulation rates). Other waveforms include the constant frequency pulse or continuous-wave, phase codes, pulse-to-pulse waveforms (which are the concatenation of several prototype waveforms), stepped frequency waveforms, etc.

The amplitude of the echo from an ideal isotropic scatterer can be modeled by the radar equation. In the case of a monostatic radar, the radar equation writes:

$$P_r = \frac{P_t G^2 \sigma \lambda^2}{(4\pi)^3 R^4 L_a(R) L_s} \tag{1.6}$$

The received echo power P_r is proportional to the average transmitted power P_t, to the target radar cross section σ, to the square of the radar wavelength λ, to the square of the antenna gain G, inversely proportional to the fourth power of the distance to the target R, inversely proportional to the hardware losses L_s and inversely proportional to the atmospheric attenuation $L_a(R)$. The atmospheric attenuation may be modeled as $L_a(R) = 10^{2\alpha R}$ with α the one-way attenuation coefficient. The attenuation coefficient notably depends on the radar frequency and on the water content of the atmosphere.

Under the quasi-stationnary approximation, up to an amplitude modulation determined by the radar equation and the antenna directivity, the received signal from an ideal scatterer can be approximated as:

$$x_r(t) = x_e\left(t - \frac{2R(t)}{c}\right) \tag{1.7}$$

Here, x_r is the received signal, x_e is the emitted signal and $R(t)$ is the range at time t. More precise relativistic models are presented in [181]. In a pulsed radar, the distance to

the target may be determined from the time delay between the emission of a pulse and the reception of its echo.

$$R = \frac{\Delta t}{2c} \tag{1.8}$$

In the case of a pulsed radar, the pulse from which the echo originates is typically not known. As a result, unless we apply another technique, the range is measured up to an integer number of periods. The period depends on the Pulse Repetition Frequency (PRF).

The radial velocity of the scatterer induces a shift between the frequencies of the emitted and received signals. It is called Doppler frequency, it is typically very small in comparison with the radar frequency and it is typically not measured directly. Instead, it is usually measured by analyzing the phase of several pulses at the same range bin. For instance, suppose an ideal point scatterer. As it travels, the time of interception varies linearly and so does the phase of the echo for a constant range. The signal obtained by sampling the same range bins of a sequence of pulses contains a spectral component whose frequency is proportional to the radial velocity of the scatterer. As modeled by the Nyquist theorem, the Doppler frequency is measured up to an integer number of periods and the period depends on the Pulse Repetition Frequency (PRF). Also, the echo of the target is not found in the same range bin for all pulses. This phenomenon is called range migration. From a data cube, we can filter out the echos of targets with known radial velocities, in a processing called Moving Target Indicator (MTI). For instance we can filter out the ground clutter so that only echos of moving targets are left and are consequently easier to detect.

SAR imaging radars often rely on in-phase / quadrature demodulation. The high-frequency signal is decomposed into two low-frequency components by projection onto two orthogonal sinusoids, as illustrated by the following equality.

$$A(t)\cos(2\pi F_c t + \phi(t)) = A(t)\cos(2\pi F_c t)\cos(\phi(t)) - A(t)\sin(2\pi F_c t)\sin(\phi(t)) \tag{1.9}$$

Here, $A(t)$ and $\phi(t)$ are respectively the amplitude and the phase of the carried signal. They are slowly varying in comparison with the frequency F_c of the carrier signal. By abuse of language, we shall call radar frequency the frequency of the carrier signal and we shall call radar wavelength the corresponding wavelength.

A block diagram of a pulsed monostatic radar is shown in figure 1.2. The antenna radiates input signals into electromagnetic waves and vice versa. The local oscillator generates the carrier signal with which modulation and demodulation are performed. The duplexer switches the antenna between emission and reception modes and so isolates the transmission and reception subsystems which operate at different power levels.

In the course of its propagation in the electro-magnetic field, the radio-frequency wave is subject to attenuation by the atmosphere: its magnitude decays exponentially with the distance travelled. The attenuation rate notably depends on the frequency of the wave and on the water content of the air.

Synthetic Aperture Radar (SAR) is a collection of techniques for producing high resolution images from radar echos of the ground [181, 50, 41]. It builds upon the observation

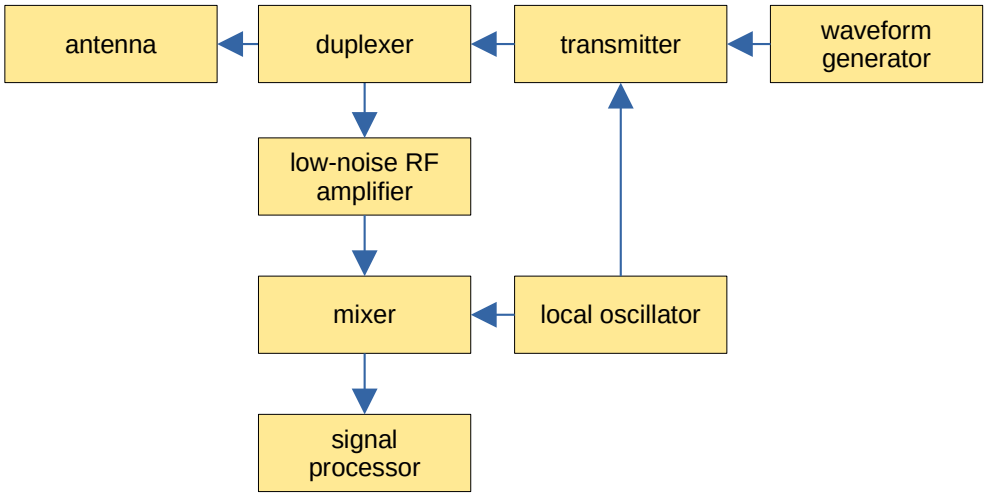


Figure 1.2 – Block diagram of a pulsed monostatic radar.

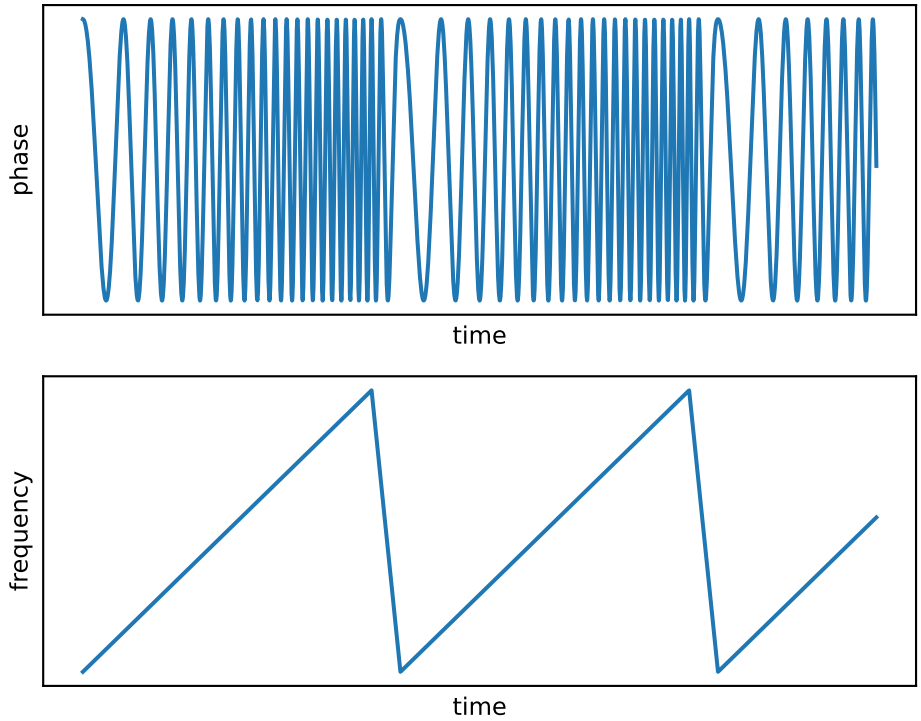


Figure 1.3 – A notional continuous-wave linear frequency modulated signal.

that the beamwidth of an aperture antenna and, consequently, its ground resolution are inversely proportional to its length. To obtain a fine resolution, instead of using an impractically large antenna, we may synthesize a virtual antenna array by moving a single antenna and assuming that the observed scene is static.

Contrary to optical imaging, radar imaging is independent of the illumination from the sun (since the radio-frequency radiation whose echo is perceived is emitted by the instrument itself) and largely independent of the cloud coverage since the attenuation rate of the electromagnetic waves at radio-frequency in the clouds is comparatively low. As well, the range resolution of SAR images is independent of the range since the range do not alter the waveform. The azimuth resolution of SAR images may be approximated as the inverse of the Doppler bandwidth (the Doppler bandwidth is the range of Doppler frequencies experimented while the scatterer lies in the antenna illumination field). As a consequence, the azimuth resolution of SAR images is also independent of range. That being said, the signal-to-noise ratio depends on the range.

With modern focusing methods, the estimated reflectivity of a SAR resolution cell is the inner product between the received echos and virtual echos that would be received if the scene was composed of only one ideal scatterer located at the center of that resolution cell. Given a bijection between the raster image plane and some physical surface, the complex value of a pixel is the inner product between the received echos and the predicted echos from an ideal scatterer located at the physical point corresponding to that pixel. The Time Domain Correlation (TDC) algorithm constructs an image by computing these cross-correlations directly. Its computational workload is impractical. Many focusing algorithms with a reduced workload have been introduced. The reduction of the workload may rely on a specialization to a particular imaging geometry, on approximations and on factorizations [180].

A simple and versatile algorithm with a reduced workload is the backprojection algorithm. It can be derived as follows. We suppose that the waveform is pulsed linear frequency modulated with chirp rate K and carrier frequency f_c. As in the previous paragraph, we denote

$$R(t) = \|\boldsymbol{p}(t) - \boldsymbol{q}\| \tag{1.10}$$

the range to the target as a function of time, with $\boldsymbol{p}$ the trajectory and $\boldsymbol{q}$ the point whose reflectivity is to be estimated. The prototype emitted signal is:

$$s_e(\tau) = W(\tau) \exp(2j\pi f_c \tau + Kj\pi\tau^2) \tag{1.11}$$

Under the quasi-stationarity approximation, neglecting the amplitude modulation, after frequency down-conversion, the received echo from pulse m is:

$$s_0(m,\tau) = W\left(\tau - \frac{2R(t_m+\tau)}{c}\right) \exp\left(-\frac{4j\pi f_c R(t_m+\tau)}{c} + Kj\pi\left(\tau - \frac{2R(t_m+\tau)}{c}\right)^2\right) \tag{1.12}$$

We denote t_m the time at the beginning of pulse m. The estimated intensity, according

to the TDC algorithm is:

$$\hat{\sigma} = \sum_m \sum_n s_0^*(m, nT_e) s_{m,n} \tag{1.13}$$

We denote T_e the sampling rate and $s_{m,n}$ the echo samples. An approximation of the range let us factorize the previous expression and derive the back-projection algorithm. Using the stop-and-go approximation, we can approximate the range as:

$$R(t_m + \tau) \approx R(t_m) \tag{1.14}$$

Under this approximation:

$$\hat{\sigma} \approx \sum_m \sum_n s_{m,n} W\left(nT_e - \frac{2R(t_m)}{c}\right) \exp\left(\frac{4j\pi f_c R(t_m)}{c} - Kj\pi\left(nT_e - \frac{2R(t_m)}{c}\right)^2\right) \tag{1.15}$$

$$\hat{\sigma} \approx \sum_m \exp\left(\frac{4j\pi f_c R(t_m)}{c}\right) \sum_n s_{m,n} W\left(T_e\left(n - \frac{2R(t_m)}{cT_e}\right)\right) \exp\left(-Kj\pi T_e^2\left(n - \frac{2R(t_m)}{cT_e}\right)^2\right) \tag{1.16}$$

$$\hat{\sigma} \approx \sum_m \exp\left(\frac{4j\pi f_c R(t_m)}{c}\right) S_m\left(\frac{2R(t_m)}{cT_e}\right) \tag{1.17}$$

We denote S_m the interpolation of the convolution of the samples of the echo from pulse m with the prototype waveform. This calculation is a backprojection algorithm. It requires one convolution (i.e. matched filtering / pulse compression) for each pulse and a monodimensional summation for each pixel. Alternatively, the Back-projection (BP) algorithm can be understood as coherently accumulating each pixel of the Real Aperture Radar (RAR) image onto the pixels of the SAR image that project onto the RAR image pixel.

The computer vision techniques that suit optical images may not suit SAR images. In appendix B, we introduce the specifics of SAR computational vision (noise models, despeckling algorithms, contour detection, etc).

1.2 Problem statement

We now describe in details the problem that we want to solve in this work. We explain our purpose and the various constraints that an admissible solution to this problem must satisfy.

1.2.1 Objective

We wish to develop the means for an aerial device to determine its kinematic state from radar imaging of the ground and a known reference description of the ground in the form of a piece of geospatial data. This description of the ground may be or derive from

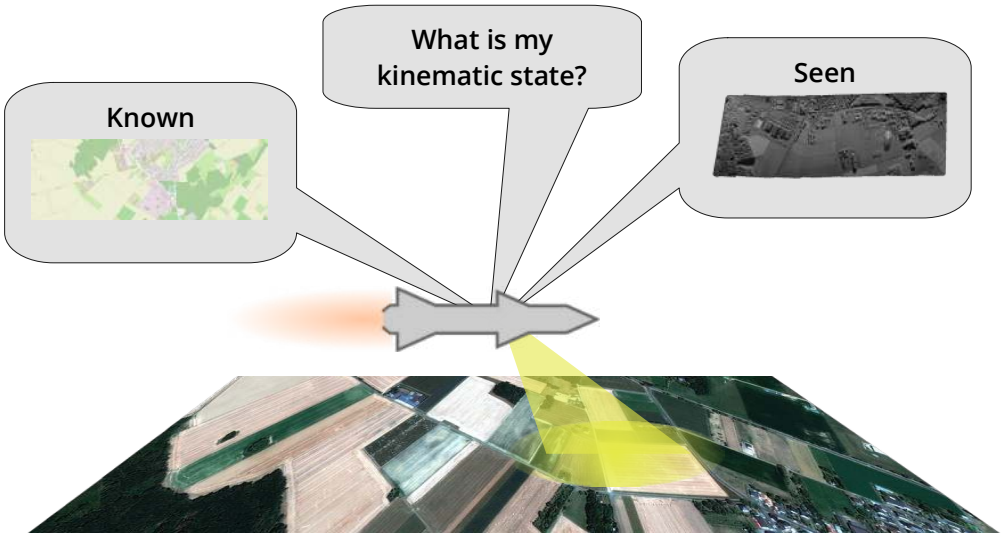


Figure 1.4 – The device determines its kinematic state from a radar observation of the ground and a known description of the ground.

an optical, SAR remote sensing image or vector geospatial data. The radar is attached to the device. We initially know that the device lies over an area whose size is in the order of 10 km by 10 km. The size of the initial altitude uncertainty interval is in the order of 200 m. The output position and velocity estimates must be expressed in the same geographic coordinate system as the reference description of the ground. The time budget for the state estimation is in the order of one second on an embedded platform. The aerial device may for instance be a missile, an aircraft, an Unmanned Aerial Vehicle (UAV). This is illustrated by figure 1.4. The process of estimating the kinematic state may happen on the device during the mission and on a support infrastructure at the stage of preparing the mission. The estimation system may also integrate navigation information from other sources, such as radar altimetry or inertial measurements.

The estimation system shall be specified in terms of an usage workflow, algorithms to be executed at the various stages of the kinematic state estimation and the description of tools.

1.2.2 Easy instanciation to a specific geographic location

As previously stated, our system relies on a radar observation and on a description of the ground to estimate its kinematic state. We want our system to be usable in various geographic places. We must consequently design a meta-system which can be specialized to a specific geographic region of interest by the adjunction of a description of this region. We should be able to build such a description easily from various kinds of sources such as vector geographic data, optical images and SAR images. This description should be

pluggable into the estimation system.

1.2.3 Lack of a prior observation by the device

In general, no observation of the region of interest from the sensor used for navigation is available before the actual use of the estimation system.

First, this means that we cannot access any sample pair of a radar observation and a kinematic state. Consequently, the estimation system cannot simply consist in a mathematical regression based on supervised learning [198].

Secondly, this implies that the modality of the description of the ground is necessarily different from the modality of the radar observation. The data to be compared with the observation to determine the state must originate from a different sensor.

Finally, this means that the merits and chances of success of the estimation system cannot be measured on the target region of interest. They can only be infered from those measured in other geographic regions.

1.2.4 Challenges of interpreting ground clutter

In a RAR image, the echo of an ideal point scatterer is not confined to a point. It is located on a segment of curve and is modulated in phase and amplitude due to the antenna motion and illumination pattern. The features of the ground are consequently spread over large areas, intermingled with other features, and difficult to recognize by human or machine vision.

In order to make the features of the ground easier to recognize, the radar observation must be focused into a SAR image (cf section 1.1.3). However, this requires some parameters of the device trajectory relative to the ground to be known, whereas we ignore them and aim to estimate them. Specifically, supposing a rectilinear uniform trajectory, we must know the norm of the velocity vector in order to produce a SAR image in the Instantaneous Doppler Frequency (IDF) - Instantaneous Distance (ID) domain or in the Time of Closest Approach (TCA) - Closest Distance (CD) domain. Additionally, if we want to project this image onto the ground up to a two dimensional rigid transform, we must also know the altitude and depression angle. Finally, we must also know the initial nadir and the geographic orientation of the trajectory to produce a georeferenced SAR image. We discuss SAR processing with an imprecise or incomplete antenna motion estimate in appendix A.

1.2.5 Limited time, energy and computational means

As an active sensing modality, radar requires more energy than passive optical measurements. Also, producing SAR images is computationally intensive in comparison with the compute power available on the embedded system of the aerial device. Yet, on many flying devices, the amount of energy available for sensing and computations is limited. For instance, devices such as UAV and such as those whose motor is a rocket rely on

batteries. More energy requirements implies heavier batteries and more heat from the electronic equipments. More heat implies the need for a heat dissipation mechanism or implies a shorter time before the electronics reach their thermal limits and fail. These are additional operational or design constraints.

1.2.6 Unavailability of synthetic observations

The localization systems based on optical imaging can benefit from the ease of synthesizing optical images. The physically realistic simulation of optical images [176] is mature enough so that computer vision algorithms can be trained and evaluated with synthetic images [111, 249]. Synthetic images are often easier to acquire than real images. For instance, in some space exploration missions, only synthetic images are available [122]. The efficient synthesis of optical images relies on the ray optics approximation, which is invalid at a radio frequency. Synthesizing realistic radar echos and SAR images requires to solve Maxwell equations for large scenes, which is computationally impractical. Existing computationally practical SAR image simulators are not realistic enough for complex scenes [121, 160, 64, 15, 114]. Simple single scatterer target models (i.e. Swerling models [181]) are good enough for evaluating detection and tracking algorithms not for simulating SAR images. As a consequence, we may not rely on simulating realistic SAR images in our context.

1.2.7 What is a good solution?

The system should be designed so that it might be implemented easily from a technological and economic point of view and so that it might best fulfill its operational purpose. To that end:

— It might exploit various modalities of geospatial data for use as a reference description of the ground.
— The duration and latency of preparing the mission should be low enough. The duration and latency may notably be influenced by the procurement and processing of the reference ground description.
— Preparing the mission should be easy, require little human involvement and rely on an affordable infrastructure.
— We might estimate the chances of success at the stage of preparing the mission.
— The smaller the amount of data describing the ground and stored on the device, the easier the technological implementation.
— The execution time and memory usage should be predictible and within budget. The algorithm corresponding to the online part of the estimation and executing on the device should accordingly be adapted for the architecture of the embedded computer.
— The lower the amount of radar observation required for estimating the kinematic state, the more convenient. This notably means a lower energy requirement for

sensing. However, a smaller ground footprint increases the risk that the imaged area does not contain any discriminative feature. This imposes a lower bound on the amount of radar observation. In summary, the estimation system should be efficient in its use of the radar observation. It should exploit all the information contained in the radar observation and not waste available information.

— The estimation should be precise. This may be measured by the probability distribution of the deviation between the estimate and the ground-truth.

1.2.8 Other applications

Navigation systems based on optical imaging of the ground might share common ideas with our work.

Our navigation system does not depend on a radio link nor on a GNSS, although it depends on radar imaging. It could consequently operate in an environment subject to electronic warfare, provided that the radar itself is not jammed.

There already exist navigation systems based on radar altimetry or optical imaging of the ground. However, a navigation system based on radar imaging could provide certain comparative benefits and unlock new operating domains. Notably, radar imaging is independent of natural illumination and of meteorological conditions when operating in an appropriate frequency band. The resolution of SAR images is independant of range.

Finally, estimating the kinematic state of an aerial device may serve the purpose of calculating the geographic coordinates of a target. An aerial device may detect a target and localize it relative to itself. For further operations, the relative coordinates might be converted to geographic coordinates. Estimating the state of the device relative to the earth might enable this.

1.3 State of the art

1.3.1 Aerial navigation based on radar imaging of the ground

Gustafsson and Sjanic have investigated correcting a part of the kinematic state through autofocus [205]. They determine a kinematic state that optimizes a criterion of image quality (i.e. image entropy). However, as this method does not rely on a reference description of the ground, a part of the kinematic state remains unknown. The two researchers have extended their work in [204] where they propose to determine the complete kinematic state of the device by matching a SAR image and a georeferenced optical image. Their method is based on detecting contours and determining the registration such that the average distance between contours of the SAR image and reference contours is minimal.

Other researchers, from the Institute of Electronic Systems of the Warsaw University of Technology, investigated the autonomous estimation of the kinematic state of an aerial

device from radar imaging of the ground. With project SARINA [1], they have simulated the use of SAR imaging for aiding the inertial navigation of missiles and UAV [80, 79]. Their system can work in two modes. In the first mode, the system produces a SAR image, detects landmarks and matches them with known landmarks. In the second mode, the system produces a Digital Terrain Model (DTM) using Interferometric Synthetic Aperture Radar (InSAR) and matches it with a reference DTM. However, InSAR is outside the scope of our problem. In practice, landmarks are difficult to detect reliably and too few of them may lie in the region of interest. A system based on specific landmarks does not exploit all the content of the scene. The geometric deformation of the SAR image is not accounted for. In [148, 81], researchers from the same institute experimented estimating a plane rigid transform to superpose a SAR image and an optical image. It relies on detecting saliency points in the two images and in aligning them using Iterative Closest Point (ICP). However, their method is not fully detailed. Seemingly, the kinematic state is only refined by a rigid transformin the horizontal plane. Their approach lacks a rigorous mathematical model and the deformation of the SAR image is not accounted for.

Nitti *et al* investigated the feasability of aiding the inertial navigation of a Medium Altitude Long Endurance (MALE) UAV by SAR imaging of known ground landmarks [163]. However, they only perform a feasability analysis but do not describe an operational system. They deal with the estimation of the position only and assume that the attitude is known with sufficient precision. They assume that the relative coordinates of the landmarks can be estimated from a SAR image but they do not address the deformation of the SAR image caused by the imprecise estimate of antenna motion relative to the ground.

During his master thesis, Colton Lindstrom investigated the use of SAR imaging for navigation. In [234], he has trained a neural network to predict the kinematic state error from the combination of a SAR image focused using an erroneous state and a SAR image focused using the true state. In [136], he has proposed and evaluated a navigation filter that leverages the shift between two SAR images to better estimate the kinematic state. In [137], he has developed a model of the deformation of a SAR image due to an erroneous state, for the backprojection focusing algorithm.

In their separate respective works [167, 195], Park and Semenov proposed to geolocalize an aerial device by deep image retrieval and interest point based alignment. However, the input to their system is a SAR image, not a raw radar observation, and they do not explain how to build a SAR image not knowing the kinematic parameters of the device. Also, the reference modality is the observation modality (i.e. no multimodality), which contradicts our requirement that we do not observe the target geographic location with the observation modality before the mission.

1. SAR-based Augmented Integrity Navigation Architecture

1.3.2 Aerial navigation based on observing the ground

Although there are currently no operational navigation system based on radar imaging of the ground, there exist operational navigation systems based on radar altimetry and on optical imaging [201]. Terrain Contour Matching (TERCOM) [78] is a class of navigation systems that rely on radar altimetry. They consist in measuring the height of the device above the ground along its trajectory and in determining a virtual trajectory whereby the predicted height profile is most similar to the observed one. Likewise, Digital Scene Matching Area Correlator (DSMAC) [36, 99] is a class of navigation systems that rely on optical imaging. They consist in producing optical images of the ground and in determining a virtual pose whereby the predicted view of the ground is most similar to the observed one. In the original DSMAC system, the correlation between a predicted and an observed image is performed by normalizing the images into binary valued images and couting the ratio of agreeing superposed bits. TERCOM and DSMAC share a common principle: the estimated state is a virtual state whose predicted observation is most similar to the actual observation.

1.3.3 Pose estimation

A fundamental problem of computer vision is to infer kinematic parameters of an object given an observation. For instance, we may estimate the pose (i.e. position and attitude) of a camera relative to a known scene from an image of that scene.

Some pose estimation methods are based on detecting landmarks whose positions in space are known and on inferring the pose from the matched observation coordinates and object coordinates.

Another approach to pose estimation is Scene Coordinate Regression (SCR). A regression model predicts scene coordinates (in our case, geographic coordinates) for each image pixel. As previously, we infer the pose from the correspondences between observation and object coordinates. For instance, Shotton *et al.* demonstrated the pose estimation of an Red Green Blue Depth (RGB-D) camera using a random forest [198].

Another approach to pose estimation is Absolute Pose Regression (APR). We estimate the pose directly by regression from the observation. Kendall *et al.* demonstrated the pose estimation of a camera embedded on a Unmanned Aerial Vehicle (UAV) using a neural network [106], as did Muller *et al.* [161]. In the context of our thesis, direct pose regression is complicated by the fact that we do not have a reference radar observation of the scene of interest. However, we have pairs of colocated multimodal observations of another location so we might build a third modality into which the other two can be converted. We could then learn state regression from this third intermediate modality. The training and evaluation of this system is illustrated in figure 1.5.

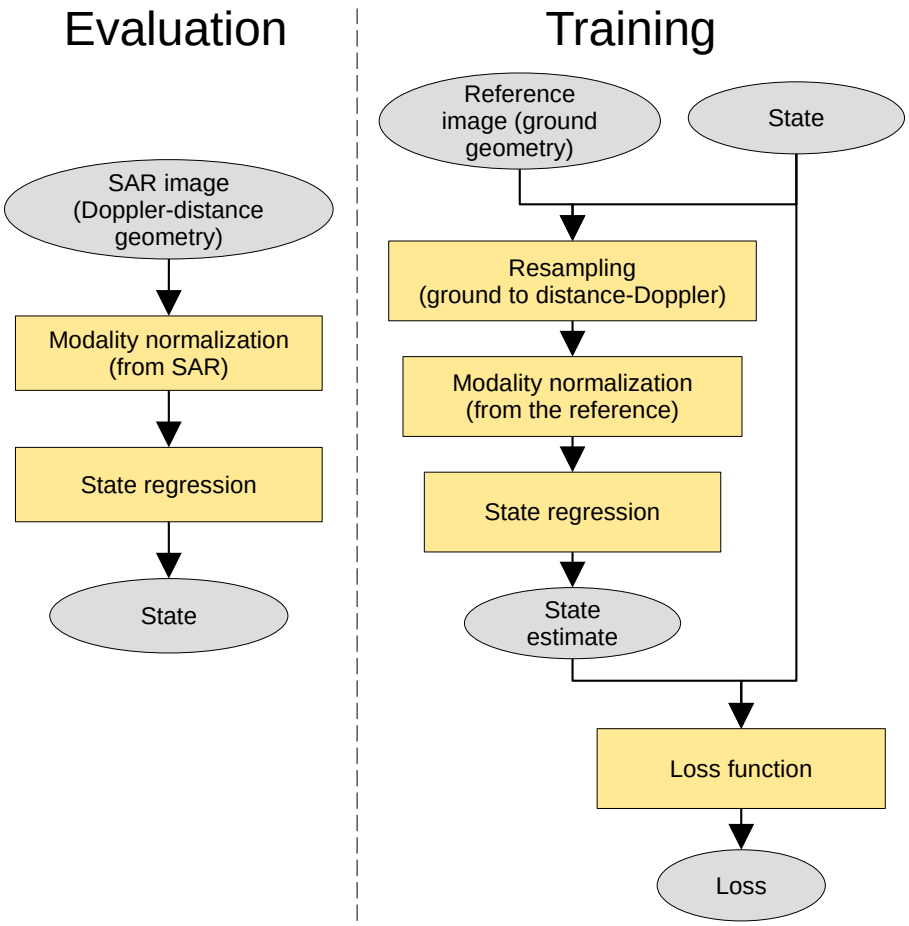


Figure 1.5 – Sketch of a multimodal absolute pose regressor.

1.3.4 Radar-based automotive navigation

The topic of radar aided automotive navigation has been receiving much attention in the last decade. The problem is to determine the geographic pose of a ground vehicle from radar measurements. The proposed methods involve using the radar for measuring distances directionally. Point clouds or occupancy grids are produced in this way and are aligned using such algorithms as Iterative Closest Point (ICP) or feature detection and matching. The methods developed for the navigation of ground vehicles are reminiscent of TERCOM. However, without resorting to interferometry, due to their large ground footprint, airborne radars cannot produce DEM at a resolution fine enough for applying the same methods used for the navigation of ground vehicles.

1.3.5 Multimodal geospatial data alignment

A typical problem of computer vision is to determine a geometric transformation between two images of identical or different modalities. For instance, one may want to determine the homography that relates the respective coordinate domains of two photographs (provided that the camera centers are the same or that the imaged scene is contained in a plane). As another example, we may want to determine an affine transform that relates the respective raster coordinates of a SAR image and an optical image.

Examples of classes of coordinate mappings are the translations, the affine transforms, the homographies and the finite elements spaces. We will review in chapter 2 the classes of coordinate mappings that are appropriate for SAR images.

If the two geospatial products have different modalities, we are required to convert each one into a geospatial product having the original geometry but a third common modality. This may consist in detecting object instances, contours or densely extracting multimodal descriptors.

Two kinds of methods exist for estimating the geometric transformation given two pieces of geospatial data having the same modality but different geometries.

Feature based geometric model fitting may be resorted to when the common modality is a set of located objects, so that we have two sets of located objects. It involves determining a geometric model from a matching between these objects. If the matching is already known, fitting a geometric model is a simple regression. If the matching is unreliable, meaning that some of the correspondences are correct and some are not, we may simultaneously determine a geometric model and a matching that are compatible and such that the cardinality of the matching is maximal. A popular heuristic to this combinatorial problem is RANdom SAmple Consensus (RANSAC) [67]. It consists in forming model hypotheses by fitting the model to small random subsets of correspondences (the smaller the subset, the higher the probability that it contains inliers only) and in scoring each model hypothesis according to the total number of compatible correspondences. If we do not have any correspondences but we have an initial estimate, we may determine the matching and the model with a fixed point strategy. We alternate between estimating the matching given the model and the model given the matching. This is the Iterative Closest Point (ICP) algorithm. However, the initial model or matching must be precise enough for this scheme to converge.

Area based geometric model fitting consists in determining the geometric model such that the average similarity of the matched locations is maximal [49]. When the two images or pieces of geospatial data have the same modality, a simple measure of local similarity such as the square difference of intensities can be used. When they have different modalities, we must resort to a more complicated measure of similarity. Popular multimodal measures of local similarity include Mutual Information (MI) [224, 242]. Other measures of local similarity are based on contour detection or on local feature descriptors.

Generally speaking, feature based methods suffer from the fact that they do not exploit the full content of the scene. The kinds of objects that they rely on may happen to be absent from a particular scene even though the scene is unambiguously identifiable by other means. Their algorithmic structure is also more complicated, offering more diverse failure modes. In practice, detecting object instances is generally more difficult than comparing patches of different modalities. Feature based methods are more intuitive. We humans prove that two locations are the same by matching landmarks. The nervous system of animals has been optimized for 600 million years to detect paths, preys, predators, etc. However, trying to reproduce biological systems, biological modes of operation and human cognition is sometimes a suboptimal heuristic for designing a machine.

The kinds of objects relied upon by feature based methods are typically saliency points and semantic objects. Saliency points are defined radiometrically and geometrically. Successfully detecting stable saliency points from images of the same modality having nearly the same viewpoints and nearly the same imaging conditions has been demonstrated and is routinely used [85]. The topic of multimodal detection of saliency points is being actively investigated but is not mature yet [220]. The modality gap between optical and SAR images makes designing a multimodal SAR-optical interest point detector challenging. In [96], Hansch *et al* proposed to detect interest points in an optical image using a well-known detector and to learn to predict them in a SAR image. But they did not specify a full image alignment pipeline.

Detecting instances of semantic objects in SAR images is made difficult by SAR artefacts (e.g. strong reflectors, layover, viewpoint dependent fluctuations) and by the little amount of data with which to train and evaluate the algorithms. Approaches based on bounding box prediction and classification were relatively unsuccessful. Most of the results reported in the literature deal with detecting ships on the ocean, which is a relatively favorable setting [39, 90, 132, 189]. Approaches based on pixel labelling were more successful. Semantic segmentation of SAR images was demonstrated with success on the task of building, road and water pixel classification [231, 212].

Feature based alignment methods often rely on image patch descriptors for matching points. Many learned or handcrafted SAR-optical image patch descriptors have been proposed [239, 253, 255, 245, 95, 94, 93].

1.3.6 Remote sensing content-based image retrieval

Content-Based Image Retrieval (CBIR) is the search of relevant images from a database given a query image. The similarity of a query image and an image from the database is typically measured using global descriptors. These are representations of a whole image. The similarity of two global descriptors is measured using a simple metric such as the L2 norm. When the image content is simple enough, traditional handcrafted or learned descriptors can be used as global descriptors [252]. When the image content is too complex, the global descriptors are typically defined by first detecting local features on the image, computing local descriptors for each local feature and aggregating the local descriptors.

The local descriptors may be handcrafted (e.g. Scale Invariant Feature Transform (SIFT) [145, 251]) or learned [98]. Popular aggregation methods include Bag of Visual Words (BOVW) [203, 120] and Vector of Locally Aggregated Descriptors (VLAD) [105]. BOVW consists in discretizing the space of local descriptors into a finite number of regions and determining how many local descriptors fall into each region. VLAD consists in discretizing the space of local descriptors into a finite number of regions and computing for each region the cumulative deviation, to the region center, of the local descriptors that fall into that region. The aggregation may be followed by a reduction of dimensionnality, for instance using Principal Components Analysis (PCA). In order to obtain better memory and computational efficiency, binary hashes are popular as global descriptors [250, 244, 215]. They may be compared using the Hamming distance. Few works tackle the problem of multimodal remote sensing image retrieval [265]. Instances of these work include [244, 215]. CBIR can intervene for the purpose of navigation. Given a set of predicted observations annotated with their respective poses, we determine which one is most similar to the actual observation.

1.3.7 Bridging the modality gap

Determining a mapping between the coordinate domains of two images is further complicated when they have different modalities. Then, in order to facilitate the alignment, we can adapt each image to a new third modality while keeping the two original geometries.

Contours For some combinations of modalities, contours that are detected separately largely coincide. Modality adaptation can then consist in detecting contours.

Image translation Image translation models are deep regression models from one modality to another [166, 100, 185]. They are often based on Generative Adversarial Models (GAN). They were notably applied to conversion between optical and SAR modalities [71, 227] and between different SAR modalities such as Sentinel-1 and TerraSAR-X [7].

Multimodal descriptors Handcrafted multimodal descriptors based on local self-similarity [255, 254, 245] and on contour histograms [2, 239, 240] have been proposed. Learned multimodal descriptors have been proposed [248, 14, 218, 152]. More details on multimodal descriptors will be given in chapter 3.

Object instances Detecting semantic object instances (e.g. roads, buildings) is a means of modality adaptation. The various techniques for detecting such objects can then be applied to modality adaptation.

Other methods In the original DSMAC, the two images are binarized (i.e. quantified on two levels) in a process that is invariant to contrast [36, 99].

1.4 Contributions

In this section, we expose the main contributions of our thesis.

1.4.1 Kinematic state estimation by aligning contours between distance/Doppler geometry and ground geometry

We can determine the kinematic state of an aerial device by aligning one cloud of distance-Doppler points and one cloud of ground points, with or without prior correspondences between the two.

Markiewicz *et al* proposed a method for estimating a rigid transform between the pseudo-ground plane and the ground plane [148]. The pseudo-ground plane is the virtual plane predicted to be the ground plane under the assumption that the actual state of the device is the estimated state. The method that they proposed involves detecting a point cloud from a SAR image in the pseudo-ground plane, detecting a point cloud from an optical image in the ground plane and aligning the two point clouds by fitting a rigid plane transform using an iterative closest point algorithm. The method that they propose enables to determine the nadir and the geographic orientation of the ground projection of the velocity vector. However, the altitude, the norm of the velocity vector and the depression angle of the velocity vector must be known. This is because this method considers a SAR image in the pseudo-ground plane instead of the distance-Doppler plane. Constructing a SAR image in the pseudo-ground plane requires more assumptions regarding the kinematic state than constructing a SAR image in the distance-Doppler plane.

In chapter 2, we describe a model that relates distance-Doppler coordinates and ground coordinates. The parameters of this model are the kinematic parameters. We propose an algorithm to fit this model, determining thereby the kinematic state, with or without known correspondences. To obtain a navigation system, the proposed geometric model fitting algorithm could be combined with a multimodal object or contour detector.

1.4.2 Kinematic state estimation relying on a multimodal similarity metric

We can estimate the kinematic state of an aerial device from a radar observation of the ground by determining the virtual kinematic state whereby a SAR image is most similar to a reference geospatial product.

Navigation based on SAR imaging of the ground is a chicken and egg problem. On the one hand, we require a SAR image for estimating the kinematic state of the device. On the other hand, we require an estimate of the kinematic state in order to build a SAR image (actually, the situation is more complicated: as we shall see further, different focalization image planes require different subsets of the kinematic state parameters to be known). As a consequence, navigation algorithms that are based on SAR imaging (few are documented in the public literature) are usually fixed point algorithms. We mean that they rely on an initial estimate of the kinematic state to build a SAR image from which they deduce a second estimate of the kinematic state. The first estimate is typically provided by inertial navigation. The intermediate SAR image may be geometrically distorted, depending on

the plane in which it is constructed. Only one fixed point iteration is typically performed. The first estimate must be precise enough so that the SAR image might be exploited by pattern recognition algorithms.

We propose a navigation algorithm that consists in determining a virtual kinematic state whereby a SAR image is most similar to the reference geospatial data. It does not require an initial estimate (only lower and upper bounds on the kinematic state parameters). It can rely on reference geospatial products of various modalities. Being an area-based method, it exploits all the content of the imaged scene.

1.4.3 Adaptiveness to various geospatial data

We can design an aerial navigation system that relies on various kinds of geospatial data and that can easily be extended to new kinds of geospatial data.

We may desire that a navigation system have the ability to exploit various kinds of geospatial data so that the user can select the most appropriate piece of geospatial data for a particular mission based on the circumstances and on a trade-off between availability, acquisition date, image quality and expected performances. For instance, cloud cover may prevent the user from using a particular source of optical images and force him to fall back to SAR images. Some geographic areas may be better suited by SAR rather than optical reference images. The navigation system should exploit a raw piece of geospatial data as produced by a provider, without requiring human editing.

In chapters 3, we describe an approach to build a navigation system that can exploit and be easily extended to various geospatial data modalities. It is based on dense image correlation using learned multimodal descriptors. We perform alignment experiments involving various combinations of geospatial data modalities such as OpenStreetMap (vector), Pléiades (optical), Digitale Orthophotos (DOP) (optical), Microsoft Bing aerial imagery (optical), I-Master (SAR), KOrean Multi-Purpose SATellite (KOMPSAT)-5 (SAR) and Miranda (SAR).

1.4.4 State estimation by scene coordinates regression

We can determine the kinematic state of the device by predicting geographic coordinates from image patches in Doppler frequency - distance geometry.

Scene coordinates regression is a family of pose estimation methods based on predicting scene coordinates from sensory observations [198]. We determine the state of the aerial device from correspondences between sensory and scene coordinates. This involves learning to predict scene coordinates from sensory observations and so requires training data. In this work, we only demonstrate state estimation using an image from the same sensor for training and inference, which violates the assumption that the modality of the reference and the modality of the observation differ. Further work should aim to learn to predict scene coordinates from the observation modality while training on the reference modality. This may involve adversarial domain adaptation [222] or image translation [100, 266, 7, 71].

1.5 Data

1.5.1 What is a geospatial product?

A geospatial product is the representation of information about a geographic area. A geospatial product is produced by an instrument (e.g. a radar imaging satellite, an optical imaging UAV) or by a human (e.g. land register). The modality of a geospatial product is the particular process by which it is constructed. Instances of modalities of geospatial products include KOMPSAT-5 (a SAR imaging satellite), Pléiades (an optical imaging satellite), OpenStreetMap (a global vector geographic database), etc. We can restrict a geospatial product to a subset of its ground footprint, thus obtaining a smaller geospatial product. A geospatial product can be viewed as a function that maps a geographic point to another quantity such as a set of annotations, a radar reflectivity, an optical irradiance, etc. A geographic coordinates system is a bijection between a subset of $\mathbb{R}^2$ and an area of the ground. A geospatial product can then be viewed as a mapping between geographic coordinates and objects of interest.

A raster geospatial product is a geospatial product that is represented as a cube of numerical values. Two axes of this cube correspond to the geographic axes. The remaining axis identifies the plane of data. A vector geospatial product is a geospatial product that is represented as a set of geometric primitives (e.g. polygons, lines). Each geometric primitive is associated with a set of annotations (e.g. street number, land use).

Our context involves such geospatial products as the SAR image produced during the mission and the reference description of the ground.

1.5.2 Multimodal stacks

We call stack of geospatial products a set of colocated geospatial products. We have built two stacks of geospatial products. The first is located in Germany and the second is located in the United Kingdom.

— The first stack is composed of five different modalities, which are OpenStreetMap, DOP, Pléiades, Miranda and KOMPSAT-5. It covers five distinct geographic locations in Germany, near the cities of Karlsruhe and Bonn. The imaged areas are countryside, sparsely urbanized, with instances of waterways and overhead power lines. A sample geographic location is represented in figure 1.6.

— The second stack is composed of three different modalities, which are OpenStreetMap, Microsoft Bing and I-Master. It covers seven distinct geographic locations in the United Kingdom. The imaged areas are countryside. There are instances of waterways. A sample geographic location is represented in figure 1.7.

In our experiments, some geospatial products serve as observations while other serve as references. We also call the observation a query. The reference geospatial product is required to cover a large area representing the location uncertainty of the aerial device. For each stack, the observation and reference modalities are reported in table 1.2.



(a) Pléiades



(b) OpenStreetMap



(c) KOMPSAT-5



(d) Miranda



(e) DOP

Figure 1.6 – A sample location from the first stack. The port of Karlsruhe seen in different modalities.

Stack	Observation modalities	Reference modalities
First stack	Miranda, Pléiades	OpenStreetMap, DOP, KOMPSAT-5
Second stack	I-Master	OpenStreetMap, Microsoft Bing

Table 1.2 – Observation and reference modalities.

1.5.3 Modalities

OpenStreetMap

OpenStreetMap is an open world wide geographic database together with a rich ecosystem of tools to manipulate its content. Any individual can contribute to OpenStreetMap. Each object contained in an OpenStreetMap archive has one of three natures:

Node A geographic point identified by its geographic coordinates.

Way A sequence of nodes.

Relation A set of other objects (i.e. nodes, ways, relations).

Each object is also annotated with textual key-value pairs. These annotations provide such information as the street number, the kind of building or wayerway, the street name, etc.

We have extracted from the OpenStreetMap global archive a few subsets corresponding to our regions of interest in Germany and in the United Kingdom.

Pléiades

Pléiades is a constellation of two optical remote sensing satellites with a panchromatic resolution of 70 cm [77]. We have purchased five true color Red Green Blue (RGB) orthoimages from the company SkyWatch [2]. Their Ground Sampling Distance (GSD) is 50 cm/pixel. The total area covered is 25 km^2.

DOP

DOP stands for *Digitale OrthoPhotos*. It is a collection of airborne optical images covering the entire german territory [3]. As part of the European Union (EU) open data directive, the states of Rhineland-Palatinate and North Rhine-Westphalia provide DOP images under an open license. The state of Baden-Württemberg currently sells DOP images but plans to provide an open access by june 2024. The resolutions of the images that we have accessed or purchased are 20 cm/pixel and 40 cm/pixel. We have produced large rasters whose GSD is 1 m/pixel.

2. SkyWatch `https://skywatch.com/`
3. Bundesamt für Kartographie und Geodäsie `https://gdz.bkg.bund.de/`

Image	Height above ground	WGS84 coordinates (deg)	Bandwidth	Acquisition date
Miranda A	930 m	N49.018834 E8.324621	1 GHz	August 2015
Miranda B	500 m	N50.618731 E7.125846	1 GHz	September 2015
Miranda C	1400 m	N50.545578 E7.161991	600 MHz	April 2015
Miranda D	800 m	N50.567383 E7.268762	600 MHz	June 2018
Miranda E	800 m	N50.578989 E7.112091	600 MHz	April 2015

Table 1.3 – Parameters of our Miranda images.

Microsoft Bing

Microsoft Bing Maps [4] is an online geographic search engine owned and operated by Microsoft. Using the piece of software QGIS [5] and Microsoft Bing Maps, we have produced a few rasters of optical imagery that cover the locations involved in our experiments. Their GSD is 2 m/pixel. The original imagery comes from Vexcel Imaging [6] and was acquired using an airborne camera. We do not know the original resolution nor the details of the processing pipeline that our rasters come from.

Miranda

Miranda is a Ka band SAR imaging radar developed by Fraunhofer Institute for High Frequency Physics [35, 68, 210, 87]. We have worked with five images. They are vertically monopolarized. They were acquired with a FMCW waveform, a radar carrier frequency of 35 GHz and a bandwidth of either 600 MHz or 1 GHz. They are geocoded and their GSD is 12.5 cm/pixel. Their range and azimuth resolution both are approximately 15 cm [68, 35]. The total area covered by our images is 18 km^2. The parameters of our five Miranda images are summarized in table 1.3.

KOMPSAT-5

KOMPSAT-5 is a Korean SAR imaging earth observation satellite [129, 258, 130]. We have purchased two images. Their imaging mode is stripmap Enhanced Standard (ES), their processing level is L1D (precisely geocoded with a Digital Elevation Model) and their

4. Microsoft Bing Maps `https://www.bing.com/maps`
5. QGIS - `https://qgis.org/`
6. Vexcel Imaging `https://vexceldata.com/`

GSD is 1.1 meters. The azimuth and range resolution both are 2.5 m. One image was vertically polarized while the other was horizontally polarized. The images were acquired on the 7th of august, 2017.

I-Master

I-Master is a Ku band, compact, lightweight airborne surveillance radar developed by Thales UK that is capable of StripMap and Spotlight SAR imaging [211, 235]. We have worked with seven StripMap images corresponding to distinct geographic locations in the United Kingdom. Their GSD is 1 m/pixel. Note that I-Master is capable of a finer resolution than that we have worked with. The total area covered by our images is 50 km^2. The images were acquired in 2018.

1.5.4 Preparation

Manual registration

Each geospatial product has an intrinsic coordinate system. Given two overlapping geospatial products, we need a way to relate the intrinsic coordinates of one geospatial product to the intrinsic coordinates of another geospatial product. As well, we need a way to relate the intrinsic coordinates of one geospatial product to geographic coordinates. In any case, we need a coordinate mapping between two coordinate domains. A coordinate mapping is a function from $\mathbb{R}^2$ to $\mathbb{R}^2$. Note that building our multimodal stack did not involve any resampling operation, which would produce altered geospatial products. It only involved determining the coordinate mappings between geographically overlapping geospatial products.

We build a particular coordinate mapping given a class of coordinate mappings and a set of sample coordinates-to-coordinates correspondences. We find, in the class of coordinate mappings, the mapping that best fit the correspondences.

We have worked with two classes of coordinate mappings. The first class is the set of affine transforms from $\mathbb{R}^2$ to $\mathbb{R}^2$. Its parameters are six real numbers. The second class is the set of natural neighbors interpolants. Its parameters are a finite part of $\mathbb{R}^2 \times \mathbb{R}^2$. Natural neighbor interpolants can represent much more flexible transformations than affine transforms. This is especially useful for our Miranda images because they exhibit layover. Their focusing assumed that the ground was flat but the scene actually contained non negligible relief such as hills. As a consequence, no affine transform precisely relates the Miranda raster coordinates to the Universal Transverse Mercator (UTM) coordinates so we had to resort to a natural neighbor interpolant.

In order to facilitate the process of aligning two geospatial products or the process of georeferencing a geospatial product, we have developed a tool. Its graphical interface is represented in figure 1.8. The user sets fixed control points on one geospatial product, moves them over the other geospatial product and selects a class of coordinate mappings. The model is fit to the pairs of coordinates defined by the user. The user can

modulate the opacity to examine the superposition of the two geospatial products. The output of the alignment process is a class of coordinate mappings and a set of coordinate correspondences.

Masks

Sometimes, we want to mask out parts of a geospatial product. For instance, a stack of geospatial products may be incoherent in some location because the content of the scene has changed between the acquisitions of two products. We may whish to mask out such parts of the geospatial products when training an algorithm.

Of course, we do not know before the mission the areas over which the reference and the observation geospatial products are incoherent, if any. This is because the observation geospatial product is not available before the mission. But we can know the incoherent areas of the samples pairs of observation and reference geospatial products of the other geographic locations that we use for training. So we use the incoherence masks only when the associated geospatial products are supposed to be available during mission preparation.

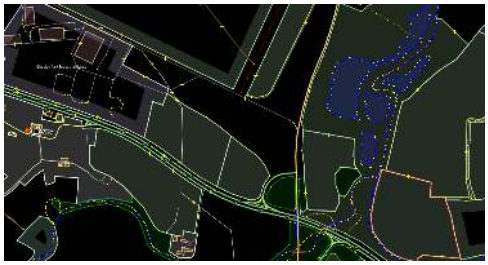
For each stack of geospatial products, we have built a mask that represents the incoherent areas.

1.6 Conclusion

In this chapter, we have introduced the topics related to our research problem. We have outlined the topic of inertial navigation, which is the collection of techniques for estimating a trajectory relying only on physical phenomena contained within the aerial device, the topic of radar imaging, by the means of which we endeavor to navigate. We have explained that our purpose is to design a system for estimating the kinematic state of an aerial device from a radar observation of the ground and we have introduced additional constraints that a system has to satisfy. The most important constraints are that we do not have an observation of the ground with the same sensor prior to the mission, that we cannot synthesize such an observation and that forming a SAR image requires part of the antenna motion relative to the ground to be known depending on the particular geometry of the image. We have reviewed the existing scientific state of the art. There does not exist a system that fully complies with all our requirements. We have described the contributions of this thesis. We shall introduce an estimation method based on aligning contours between the geometry of instantaneous Doppler frequency and distance and the ground geometry. We shall also propose an approach based on a learned multimodal similarity metric. Finally, we have described the two geospatial data stacks that the experiments of this thesis rely upon.



(a) Microsoft Bing



(b) OpenStreetMap



(c) I-Master

Figure 1.7 – A sample location from the second stack. A countryside area in the United Kingdom seen in different modalities.

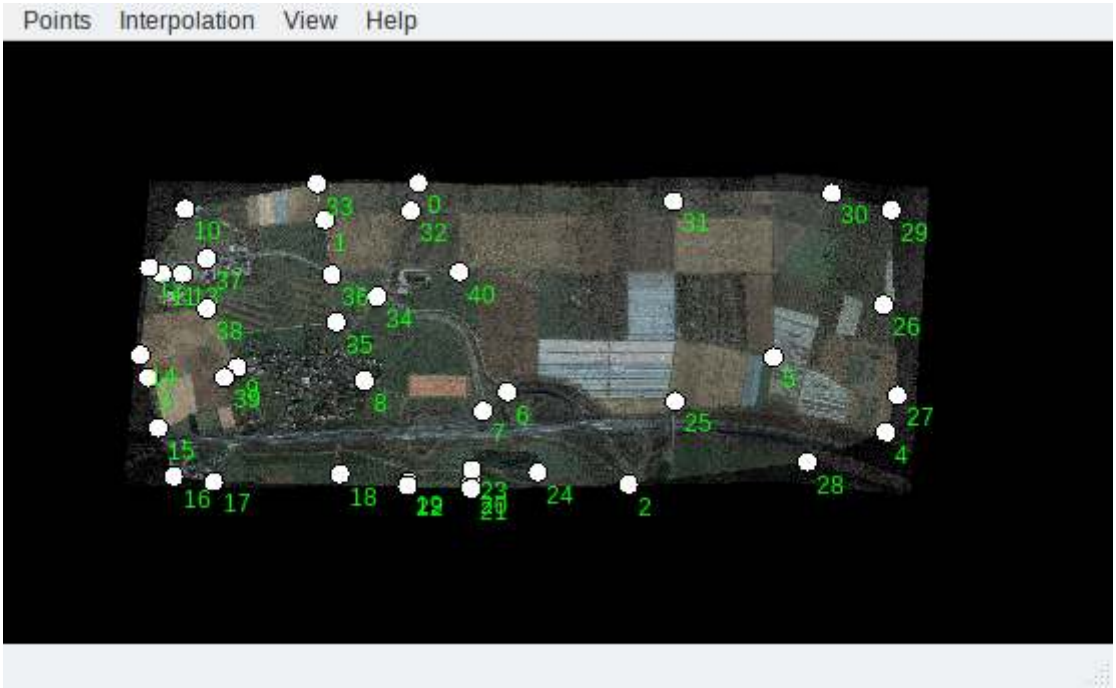


Figure 1.8 – User interface of our alignment tool.

Chapter 2

State estimation by matching contours

In this chapter, we describe the geometry of the problem at hand and we define the coordinates systems associated with SAR images. We show how to estimate the kinematic state given correspondences between different coordinates domains. For this purpose, we introduce a bayesian optimization model that requires point-to-point correspondences. We also introduce an algorithm inspired from Iterative Closest Point (ICP) that requires only a set-to-set correspondence. Finally, we propose a complete kinematic state estimation pipeline based on multimodal contour detection.

2.1 Geometric background

In this section, we introduce three image coordinate systems for the observation and the reference geospatial data. We describe how to compute image coordinates given spatial cartesian coordinates and how to convert between the three image coordinate systems. We shall see that the relation between different coordinate systems depends on the kinematic state. Conversely, in the other sections, we shall determine the kinematic state by fitting a coordinate transform.

2.1.1 Definitions

SAR images involve the following coordinate systems:

ground In this coordinate system, a physical point is identified by its geographic coordinates in a locally flat geographic coordinate system. This may be a local East North Up (ENU) coordinate system. We denote x_1^{gr} and x_2^{gr} the ground coordinates.

TCA-CD In this coordinate system, a physical point is identified by the time at which the distance between it and the antenna is minimal, called Time of Closest Approach (TCA) or zero-Doppler time, and by the distance at that time, called Closest Distance (CD) or zero-Doppler slant range. We denote the TCA x_1^{ca} and the CD x_2^{ca}.

IDF-ID In this coordinate system, a physical point is identified by the distance between it and the antenna, at a reference instant of time, and by its Doppler frequency at that time. These coordinates are respectively called Instantaneous Dis-

tance (ID) and Instantaneous Doppler Frequency (IDF). We denote the ID x_1^{rd} and the IDF x_2^{rd}.

raster In this coordinate system, a physical point is identified by its row and column pixel indices. There exists a bijective affine transform between the raster coordinates and another coordinate system such as IDF-ID, TCA-CD or ground.

The TCA-CD and IDF-ID coordinates are illustrated in figure 2.1. The notations used in this chapter are summed up in table 2.1.

Symbol	Domain	Description
$\boldsymbol{o}$	$\mathbb{R}^3$	The position of the device.
$\boldsymbol{v}$	$\mathbb{R}^3$	The velocity of the device.
$\boldsymbol{A}$	$\mathbb{R}^{3\times 2}$	A matrix whose two columns define the two ground plane axes. We assume that the columns are orthonormal.
$\boldsymbol{A_1}$	$\mathbb{R}^3$	The left column of $\boldsymbol{A}$.
$\boldsymbol{A_2}$	$\mathbb{R}^3$	The right column of $\boldsymbol{A}$.
$\boldsymbol{N}$	$\mathbb{R}^3$	The normal to the ground $\boldsymbol{N} = \boldsymbol{A_1} \times \boldsymbol{A_2}$.
$\boldsymbol{b}$	$\mathbb{R}^3$	The spatial cartesian coordinates of a point on the ground taken as the origin of the ground cartesian coordinates.
s	$\{-1, 1\}$	The sign of the triple product between the velocity $\boldsymbol{v}$, the antenna pointing direction and the normal to the ground $\boldsymbol{N}$. If the antenna looks left and the normal points upward then $s = 1$. If the antenna looks right and the normal points downward then $s = 1$. Else, $s = -1$.
$\boldsymbol{p}$	$\mathbb{R}^3$	The spatial cartesian coordinates of a point in space.
x_1^{gr}	$\mathbb{R}$	The first ground coordinate of a point in space (e.g. easting).
x_2^{gr}	$\mathbb{R}$	The second ground coordinate of a point in space (e.g. northing).
x_1^{ca}	$\mathbb{R}$	The time of closest approach to a point in space.
x_2^{ca}	$\mathbb{R}$	The closest distance to a point in space.
x_1^{rd}	$\mathbb{R}$	The distance at the reference instant of time to a point in space.
x_2^{rd}	$\mathbb{R}$	The Doppler frequency, at the reference instant of time, of a point in space.
$\boldsymbol{I_3}$	$\mathbb{R}^{3\times 3}$	The three-dimensional identity matrix.
λ_c	$\mathbb{R}_+$	The radar carrier frequency.

Table 2.1 – Notations.

2.1.2 Projection from space to image coordinates

In this section, we define how to compute the image coordinates of a point given its three-dimensional cartesian coordinates.

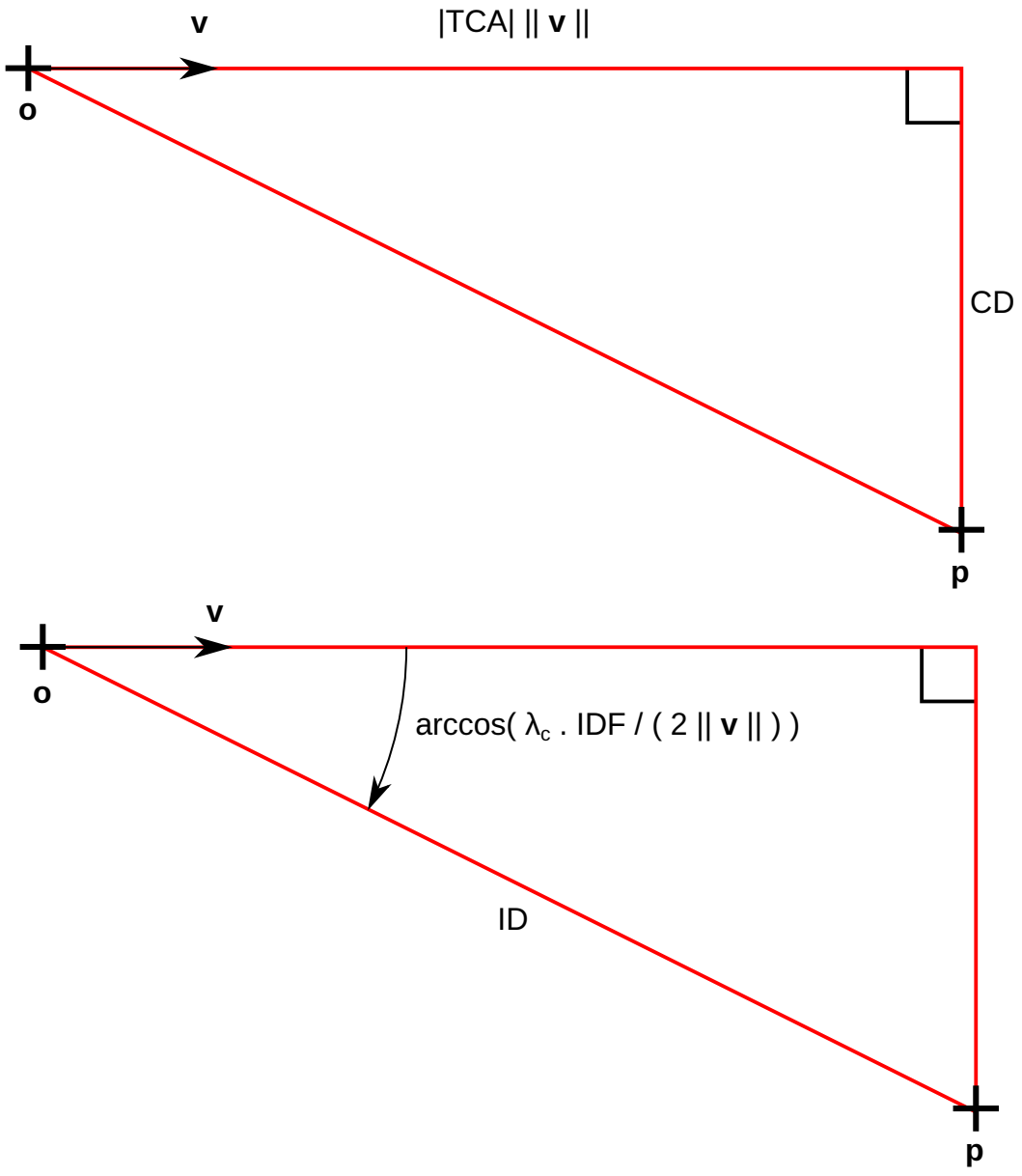


Figure 2.1 – Illustration of TCA-CD and IDF-ID coordinates in the slant range plane. See table 2.1 for the notations.

Projection from space to ground

$$\psi^{gr} : \mathbb{R}^3 \to \mathbb{R}^2, \boldsymbol{p} \to \boldsymbol{x^{gr}} \tag{2.1}$$

$$\boldsymbol{x^{gr}} = \boldsymbol{A}^T(\boldsymbol{p} - \boldsymbol{b}) \tag{2.2}$$

Projection from space to TCA-CD

$$\psi^{ca} : \mathbb{R}^3 \to \mathbb{R}^2, \boldsymbol{p} \to \boldsymbol{x^{ca}} \tag{2.3}$$

$$\begin{cases} x_1^{ca} = \frac{1}{\|\boldsymbol{v}\|^2}\boldsymbol{v}^T(\boldsymbol{p} - \boldsymbol{o}) \\ x_2^{ca} = \left\|(\boldsymbol{I_3} - \frac{1}{\|\boldsymbol{v}\|^2}\boldsymbol{v}\boldsymbol{v}^T)(\boldsymbol{p} - \boldsymbol{o})\right\| \end{cases} \tag{2.4}$$

Projection from space to IDF-ID

$$\psi^{rd} : \mathbb{R}^3 \to \mathbb{R}^2, \boldsymbol{p} \to \boldsymbol{x^{rd}} \tag{2.5}$$

$$\begin{cases} x_1^{rd} = \|\boldsymbol{p} - \boldsymbol{o}\| \\ x_2^{rd} = \frac{2}{\lambda_c}\boldsymbol{v}^T\frac{(\boldsymbol{p}-\boldsymbol{o})}{\|\boldsymbol{p}-\boldsymbol{o}\|} \end{cases} \tag{2.6}$$

2.1.3 Transformation between different coordinate systems

In this section, we define how to convert between different coordinate systems.

Transformation from ground to TCA-CD

Transforming from ground to TCA-CD is straightforward.

$$\phi^{gr \to ca} : \mathbb{R}^2 \to \mathbb{R}^2, \boldsymbol{x^{gr}} \to \boldsymbol{x^{ca}} \tag{2.7}$$

$$\begin{cases} x_1^{ca} = \frac{1}{\|\boldsymbol{v}\|^2}\boldsymbol{v}^T(\boldsymbol{A}\boldsymbol{x^{gr}} + \boldsymbol{b} - \boldsymbol{o}) \\ x_2^{ca} = \left\|(\boldsymbol{I_3} - \frac{1}{\|\boldsymbol{v}\|^2}\boldsymbol{v}\boldsymbol{v}^T)(\boldsymbol{A}\boldsymbol{x^{gr}} + \boldsymbol{b} - \boldsymbol{o})\right\| \end{cases} \tag{2.8}$$

Transformation from TCA-CD to ground

$$\phi^{ca \to gr} : \mathbb{R}^2 \to \mathbb{R}^2, \boldsymbol{x^{ca}} \to \boldsymbol{x^{gr}} \tag{2.9}$$

Transforming TCA-CD coordinates to ground coordinates involves finding the intersection between the ground plane, the zero-Doppler plane and a sphere. The center of this sphere is $\boldsymbol{c}$:

$$\boldsymbol{c} := \boldsymbol{o} + x_1^{ca}\boldsymbol{v} \tag{2.10}$$

The radius of the sphere is x_2^{ca}.

We denote $\boldsymbol{c'}$ the projection of $\boldsymbol{c}$ onto the ground plane, in ground coordinates.

$$\boldsymbol{c'} := \boldsymbol{A}^T(\boldsymbol{c} - \boldsymbol{b}) \tag{2.11}$$

We define $\boldsymbol{u_1}$ and $\boldsymbol{u_2}$.

$$\boldsymbol{u_1} := \frac{\boldsymbol{A}^T\boldsymbol{v}}{\|\boldsymbol{A}^T\boldsymbol{v}\|} \tag{2.12}$$

$$\boldsymbol{u_2} := \begin{pmatrix} 0 & -s \\ s & 0 \end{pmatrix} \boldsymbol{u_1} \tag{2.13}$$

We shall express the ground coordinates of the target in terms of $\boldsymbol{c'}$, $\boldsymbol{u_1}$ and $\boldsymbol{u_2}$. Define H as the signed height of the center of the circle:

$$H := \boldsymbol{N}^T\left(\boldsymbol{c} - (\boldsymbol{A}\boldsymbol{c'} + \boldsymbol{b})\right) \tag{2.14}$$

$$H = \boldsymbol{N}^T(\boldsymbol{c} - \boldsymbol{b}) \tag{2.15}$$

Define β as the angle between the ground normal and the velocity vector:

$$\beta := \arccos\left(\frac{\boldsymbol{N}^T\boldsymbol{v}}{\|\boldsymbol{v}\|}\right) \tag{2.16}$$

The coordinates of the target in terms of $\boldsymbol{c'}$, $\boldsymbol{u_1}$ and $\boldsymbol{u_2}$ are:

$$\begin{cases} \alpha_1 := \frac{H}{\tan(\beta)} \\ \alpha_2 := \sqrt{(x_2^{ca})^2 - \frac{H^2}{\sin^2\beta}} \end{cases} \tag{2.17}$$

Finally, the ground coordinates of the target are:

$$\boldsymbol{x^{gr}} = \boldsymbol{c'} + \alpha_1\boldsymbol{u_1} + \alpha_2\boldsymbol{u_2} \tag{2.18}$$

Transformation from ground to IDF-ID

Transforming from ground to IDF-ID is straightforward.

$$\phi^{gr\to rd} : \mathbb{R}^2 \to \mathbb{R}^2, \boldsymbol{x^{gr}} \to \boldsymbol{x^{rd}} \tag{2.19}$$

$$\begin{cases} x_1^{rd} = \|\boldsymbol{A}\boldsymbol{x^{gr}} + \boldsymbol{b} - \boldsymbol{o}\| \\ x_2^{rd} = \frac{2}{\lambda_c}\frac{(\boldsymbol{A}\boldsymbol{x^{gr}}+\boldsymbol{b}-\boldsymbol{o})^T\boldsymbol{v}}{\|\boldsymbol{A}\boldsymbol{x^{gr}}+\boldsymbol{b}-\boldsymbol{o}\|} \end{cases} \tag{2.20}$$

Transformation from IDF-ID to ground

$$\phi^{rd\to gr} : \mathbb{R}^2 \to \mathbb{R}^2, \boldsymbol{x^{rd}} \to \boldsymbol{x^{gr}} \tag{2.21}$$

Transforming IDF-ID coordinates to ground coordinates involves finding the intersection between the ground plane, a half cone and a sphere. First, let us determine the intersection between the sphere and the half cone, which is a circle. The center of this circle is:

$$\boldsymbol{c} := \boldsymbol{o} + \frac{\lambda_c x_1^{rd} x_2^{rd}}{2\|\boldsymbol{v}\|^2}\boldsymbol{v} \tag{2.22}$$

The circle is contained in the plane containing $\boldsymbol{c}$ and whose normal is $\boldsymbol{v}$. Its radius is R.

$$R := x_1^{rd}\sqrt{1 - \left(\frac{\lambda_c x_2^{rd}}{2\|\boldsymbol{v}\|}\right)^2} \tag{2.23}$$

We have to determine the intersection between this circle and the ground plane.

As for the transformation from closest approach to ground, we define a frame $(\boldsymbol{c'}, \boldsymbol{u_1}, \boldsymbol{u_2})$ on the ground plane, whose origin is the projection of $\boldsymbol{c}$ onto the ground plane, and which is aligned with the projected velocity.

$$\boldsymbol{c'} := \boldsymbol{A}^T(\boldsymbol{c} - \boldsymbol{b}) \tag{2.24}$$

$$\boldsymbol{u_1} := \frac{\boldsymbol{A}^T\boldsymbol{v}}{\|\boldsymbol{A}^T\boldsymbol{v}\|} \tag{2.25}$$

$$\boldsymbol{u_2} := \begin{pmatrix} 0 & -s \\ s & 0 \end{pmatrix}\boldsymbol{u_1} \tag{2.26}$$

As before, we shall express the ground coordinates of the target in terms of $\boldsymbol{c'}$, $\boldsymbol{u_1}$ and $\boldsymbol{u_2}$. The signed height of the center of the circle is still defined as:

$$H := \boldsymbol{N}^T\left(\boldsymbol{c} - (\boldsymbol{A}\boldsymbol{c'} + \boldsymbol{b})\right) \tag{2.27}$$

$$H = \boldsymbol{N}^T(\boldsymbol{c} - \boldsymbol{b}) \tag{2.28}$$

The angle between the ground normal and the velocity vector is still:

$$\beta := \arccos\left(\frac{\boldsymbol{N}^T\boldsymbol{v}}{\|\boldsymbol{v}\|}\right) \tag{2.29}$$

The coordinates of the target in terms of $\boldsymbol{c'}$, $\boldsymbol{u_1}$ and $\boldsymbol{u_2}$ are now:

$$\begin{cases} \alpha_1 := \frac{H}{\tan(\beta)} \\ \alpha_2 := \sqrt{R^2 - \frac{H^2}{\sin^2\beta}} \end{cases} \tag{2.30}$$

Finally, the ground coordinates of the target are:

$$\boldsymbol{x^{gr}} = \boldsymbol{c'} + \alpha_1\boldsymbol{u_1} + \alpha_2\boldsymbol{u_2} \tag{2.31}$$

Transformation from IDF-ID to TCA-CD

Transforming from IDF-ID to TCA-CD involves the geometry of a right triangle.

$$\phi^{rd \to ca} : \mathbb{R}^2 \to \mathbb{R}^2, \boldsymbol{x^{rd}} \to \boldsymbol{x^{ca}} \tag{2.32}$$

$$\begin{cases} x_1^{ca} = \frac{x_1^{rd} x_2^{rd} \lambda_c}{2\|\boldsymbol{v}\|^2} \\ x_2^{ca} = x_1^{rd} \sqrt{1 - \left(\frac{\lambda_c x_2^{rd}}{2\|\boldsymbol{v}\|}\right)^2} \end{cases} \tag{2.33}$$

Transformation from TCA-CD to IDF-ID

Transforming from TCA-CD to IDF-ID relies on the geometry of a right triangle.

$$\phi^{ca \to rd} : \mathbb{R}^2 \to \mathbb{R}^2, \boldsymbol{x^{ca}} \to \boldsymbol{x^{rd}} \tag{2.34}$$

$$\begin{cases} x_1^{rd} = \sqrt{\left(x_1^{ca}\right)^2 \|\boldsymbol{v}\|^2 + \left(x_2^{ca}\right)^2} \\ x_2^{rd} = \frac{2x_1^{ca}\|\boldsymbol{v}\|^2}{\lambda_c \sqrt{\left(x_1^{ca}\right)^2 \|\boldsymbol{v}\|^2 + \left(x_2^{ca}\right)^2}} \end{cases} \tag{2.35}$$

2.1.4 Pseudo image coordinates

If we identify physical points with a first coordinate system, we can construct a second coordinate system determined by the bijection between the two coordinate systems. In particular, we can construct new coordinate systems, called pseudo coordinate systems, by using the coordinates transforms defined in the previous sections as bijections. For instance, we can define pseudo ground coordinates by using the coordinate transforms between ground and IDF-ID but plugging in arbitrary position and velocity parameters. If these position and velocity parameters were equal to the actual position and velocity parameters, then the pseudo ground coordinates just defined would be identical to the ground coordinates.

2.1.5 Illustration

Figures 2.2, 2.3 and 2.4 represent aerial photographs of the location of Thales at Élancourt, France, respectively in ground coordinates, TCA-CD coordinates and IDF-ID coordinates. They were obtained by resampling an optical image. The simulated imaging parameters are reported in table 2.2.

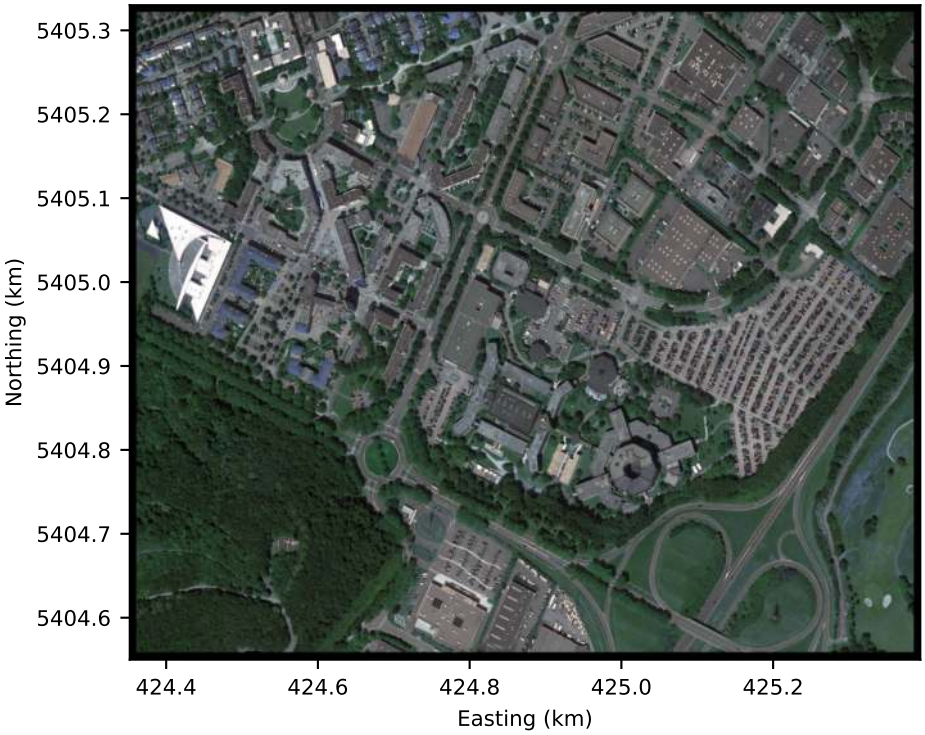


Figure 2.2 – The location of Thales at Élancourt in ground coordinates.

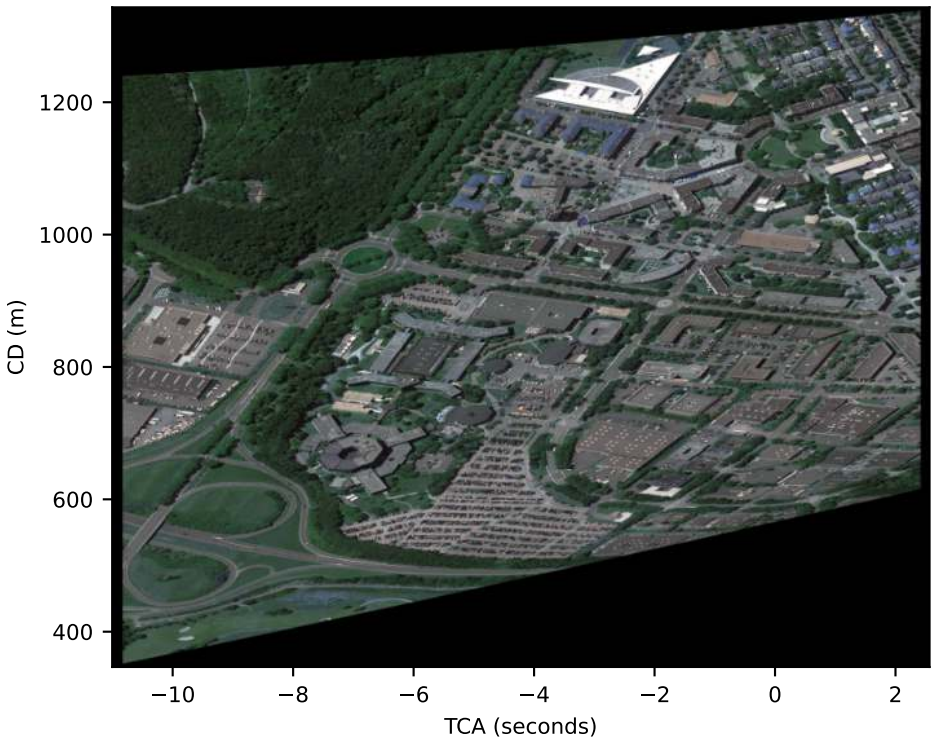


Figure 2.3 – The location of Thales at Élancourt in TCA-CD coordinates.

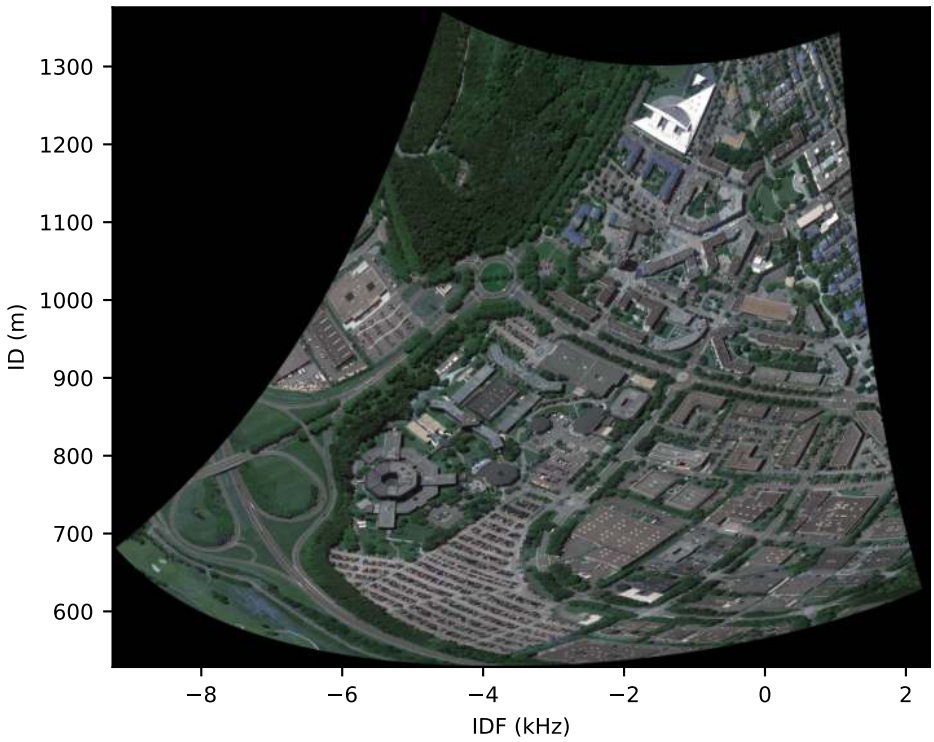


Figure 2.4 – The location of Thales at Élancourt in IDF-ID coordinates.

Parameter	Value
Imaged area	1025 m by 770 m
Center point (UTM 31N)	424861 ; 5404985
Platform altitude	500 m
Platform horizontal speed	50 m/s
Platform vertical speed	20 m/s
Radar frequency	30 GHz

Table 2.2 – Resampling parameters.

2.2 Determining the kinematic state with known correspondences

In order to construct a SAR image in a particular coordinate system, we must know, for each point of the coordinate domain, the range between the corresponding physical point and the antenna as a function of azimuth time η, which is:

$$R(\eta, \boldsymbol{p}) = \|\boldsymbol{o} + \eta\boldsymbol{v} - \boldsymbol{p}\| \tag{2.36}$$

The expressions of the slant range as a function of azimuth time and image coordinates are:

$$\begin{aligned} R^{ca}(\eta, \boldsymbol{x^{ca}}) &= \sqrt{\left(x_2^{ca}\right)^2 + (\eta - x_1^{ca})^2 \|\boldsymbol{v}\|^2} \\ R^{rd}(\eta, \boldsymbol{x^{rd}}) &= \sqrt{\left(x_1^{rd}\right)^2 \left(1 - \left(\frac{\lambda_c x_2^{rd}}{2\|\boldsymbol{v}\|}\right)^2\right) + \left(\eta - \frac{x_1^{rd} x_2^{rd} \lambda_c}{2\|\boldsymbol{v}\|^2}\right)^2 \|\boldsymbol{v}\|^2} \\ R^{gr}(\eta, \boldsymbol{x^{gr}}) &= \|(\boldsymbol{o} + \eta\boldsymbol{v}) - (\boldsymbol{A}\boldsymbol{x^{gr}} + \boldsymbol{b})\| \end{aligned} \tag{2.37}$$

The only parameter of R^{ca} is $\|\boldsymbol{v}\|$. The parameters of R^{rd} are $\|\boldsymbol{v}\|$ and λ_c. The parameters of R^{gr} are $\boldsymbol{o}$, $\boldsymbol{v}$, $\boldsymbol{A}$ and $\boldsymbol{b}$. In our context, parameters λ_c, $\boldsymbol{A}$ and $\boldsymbol{b}$ are known. Constructing a SAR image in TCA-CD coordinates or IDF-ID coordinates requires to know the platform speed $\|\boldsymbol{v}\|$ but constructing a SAR image in ground coordinates requires to know the full kinematic state. It is easier to form a geometrically correct SAR image in IDF-ID coordinates or TCA-CD coordinates than in ground coordinates. In this work, we assume that a SAR image in IDF-ID or TCA-CD coordinates can be built and used to determine the kinematic state of the device. A more detailed justification of the hypothesis that such a SAR image is available is developed in appendix A. In the following, we describe an algorithm for estimating the kinematic state $(\boldsymbol{o}, \boldsymbol{v})$ from known correspondences between the ground coordinates and some coordinate system X, where X designates either TCA-CD or IDF-ID.

Assume that we have two collections of coordinate tuples $(P_i^X)_{1\leq i\leq n}$ and $(P_i^{gr})_{1\leq i\leq n}$. The first set of coordinate tuples belong to coordinate system X. The second set of

coordinate tuples are ground coordinates. In this section, we suppose that, for all i, point P_i^X and point P_i^{gr} correspond to the same physical point modulo some noise (in section 2.3, we will consider that the two collections of coordinate tuples have different sizes in general). A computer vision algorithm or a human annotator shall produce such pairs of coordinates. Consequently, we have:

$$P_i^X = \phi^{gr \to X}(P_i^{gr}, \boldsymbol{o}, \boldsymbol{v}) + \boldsymbol{\epsilon}_i \tag{2.38}$$

Coordinate transform $\phi^{gr \to X}$ is defined as follows (see section 2.1.3):

$$\begin{aligned} \phi^{gr \to ca}(\boldsymbol{x^{gr}}, \boldsymbol{o}, \boldsymbol{v}) &= \begin{pmatrix} \frac{1}{\|\boldsymbol{v}\|^2} \boldsymbol{v}^T (\boldsymbol{A x^{gr}} + \boldsymbol{b} - \boldsymbol{o}) \\ \left\| (\boldsymbol{I_3} - \frac{1}{\|\boldsymbol{v}\|^2} \boldsymbol{v}\boldsymbol{v}^T)(\boldsymbol{A x^{gr}} + \boldsymbol{b} - \boldsymbol{o}) \right\| \end{pmatrix} \\ \phi^{gr \to rd}(\boldsymbol{x^{gr}}, \boldsymbol{o}, \boldsymbol{v}) &= \begin{pmatrix} \left\| \boldsymbol{A x^{gr}} + \boldsymbol{b} - \boldsymbol{o} \right\| \\ \frac{2}{\lambda_c} \frac{(\boldsymbol{A x^{gr}} + \boldsymbol{b} - \boldsymbol{o})^T \boldsymbol{v}}{\|\boldsymbol{A x^{gr}} + \boldsymbol{b} - \boldsymbol{o}\|} \end{pmatrix} \end{aligned} \tag{2.39}$$

Noise $\boldsymbol{\epsilon}_i$ is a two-dimensional normal random variable with zero mean and covariance:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{pmatrix} \tag{2.40}$$

If $i \neq j$ then variables $\boldsymbol{\epsilon}_i$ and $\boldsymbol{\epsilon}_j$ are independent.

We wish to retrieve the most likely $(\boldsymbol{o}, \boldsymbol{v})$. This corresponds to the following optimization problem:

$$\underset{(\boldsymbol{o},\boldsymbol{v})}{\text{maximize}}\, p(\boldsymbol{o}, \boldsymbol{v} | P_1^X, .., P_n^X, P_1^{gr}, .., P_n^{gr}) \tag{2.41}$$

Using a Bayesian reasonning, we obtain the following equivalent problem.

$$\underset{(\boldsymbol{o},\boldsymbol{v})}{\text{maximize}}\, \frac{p(\boldsymbol{o}, \boldsymbol{v}, P_1^X, .., P_n^X | P_1^{gr}, .., P_n^{gr})}{p(P_1^X, .., P_n^X | P_1^{gr}, .., P_n^{gr})} \tag{2.42}$$

The denominator is independant of the optimization variable so an equivalent problem is as follows.

$$\underset{(\boldsymbol{o},\boldsymbol{v})}{\text{maximize}}\, p(\boldsymbol{o}, \boldsymbol{v}, P_1^X, .., P_n^X | P_1^{gr}, .., P_n^{gr}) \tag{2.43}$$

Using Bayesian reasonning:

$$\underset{(\boldsymbol{o},\boldsymbol{v})}{\text{maximize}}\, p(P_1^X, .., P_n^X | \boldsymbol{o}, \boldsymbol{v}, P_1^{gr}, .., P_n^{gr}) p(\boldsymbol{o}, \boldsymbol{v} | P_1^{gr}, .., P_n^{gr}) \tag{2.44}$$

Since the ground coordinates and the kinematic state are independent:

$$\underset{(\boldsymbol{o},\boldsymbol{v})}{\text{maximize}}\, p(P_1^X, .., P_n^X | \boldsymbol{o}, \boldsymbol{v}, P_1^{gr}, .., P_n^{gr}) p(\boldsymbol{o}, \boldsymbol{v}) \tag{2.45}$$

Equivalently, we minimize the negative log-likelihood.

$$\underset{(\boldsymbol{o},\boldsymbol{v})}{\text{minimize}} - \log p(P_1^X, .., P_n^X | \boldsymbol{o}, \boldsymbol{v}, P_1^{gr}, .., P_n^{gr}) - \log p(\boldsymbol{o}, \boldsymbol{v}) \tag{2.46}$$

We assume that the observations are independent and that P_i^X and P_j^{gr} are independent whenever $i \neq j$. We obtain the following optimization problem.

$$\underset{(\boldsymbol{o},\boldsymbol{v})}{\text{minimize}} \left(- \sum_{i=1}^{n} \log p(P_i^X | \boldsymbol{o}, \boldsymbol{v}, P_i^{gr}) \right) - \log p(\boldsymbol{o}, \boldsymbol{v}) \tag{2.47}$$

Assuming normally distributed errors, we may write the two terms of the previous cost function as:

$$\begin{aligned} E_{\text{prior}}(\boldsymbol{o}, \boldsymbol{v}) :=& \frac{1}{\sigma_o^2} \|\boldsymbol{o} - \boldsymbol{o}_{\text{prior}}\|^2 + \frac{1}{\sigma_v^2} \|\boldsymbol{v} - \boldsymbol{v}_{\text{prior}}\|^2 \\ E_{\text{obs}}(\boldsymbol{o}, \boldsymbol{v}) :=& \sum_{i=1}^{n} \left(P_i^X - \phi_{gr \to X}(P_i^{gr}, \boldsymbol{o}, \boldsymbol{v}) \right)^T \boldsymbol{\Sigma}^{-1} \left(P_i^X - \phi_{gr \to X}(P_i^{gr}, \boldsymbol{o}, \boldsymbol{v}) \right) \end{aligned} \tag{2.48}$$

Parameters σ_o and σ_v represent the weight of the prior kinematic state $(\boldsymbol{o}_{\text{prior}}, \boldsymbol{v}_{\text{prior}})$. The final optimization problem is:

$$\underset{(\boldsymbol{o},\boldsymbol{v})}{\text{minimize}}\, E(\boldsymbol{o}, \boldsymbol{v}) \tag{2.49}$$

The cost function $E := E_{\text{prior}} + E_{\text{obs}}$ is differentiable over the kinematic states of interest.

2.3 Determining the kinematic state without known correspondences

The optimization model that we have described in the previous section allows to estimate the kinematic state from known correspondences between TCA-CD or IDF-ID coordinates on the one hand and ground coordinates on the other hand. In this section, we describe an algorithm for estimating the same parameters but without knowing point-to-point correspondences between the two sets of points. Instead, the input of this algorithm is one set-to-set correspondence. This algorithm is inspired from Iterative Closest Point (ICP) [21, 42] and builds upon these observations:

— if we know the kinematic state, we can determine correspondences between the two domains.

— if we know correspondences between the two domains, we can determine the kinematic state.

Knowing neither the kinematic state nor correspondences between the two domains, we alternatively estimate one while hypothesizing the other by block coordinate descent. Determining a kinematic state given correspondences is performed using the method exposed in the previous section. Determining correspondences between the two coordinate

domains given a kinematic state is performed as follows: at iteration k, for all i, we associate P_i^X with its likeliest correspondence $P_{m_{i,k}}^{gr}$ given kinematic state $(\boldsymbol{o}_k, \boldsymbol{v}_k)$.

$$m_{i,k} := \arg\min_j \left(\phi_{gr\to X}(P_j^{gr}, \boldsymbol{o}_k, \boldsymbol{v}_k) - P_i^X\right)^T \boldsymbol{\Sigma}^{-1} \left(\phi_{gr\to X}(P_j^{gr}, \boldsymbol{o}_k, \boldsymbol{v}_k) - P_i^X\right) \quad (2.50)$$

Computing $m_{i,k}$ is tantamount to a nearest neighbor search. We can accelerate this search using such a spatial data structure as a kd-tree.

Our ICP scheme is formally defined in algorithm 1. We did not prove its convergence but we have observed it empirically provided that the initial estimate is close enough to the solution. This algorithm extends and was inspired by the work of Markiewicz [148] who proposed an ICP algorithm to estimate an affine transform between a SAR image and a ground reference.

Input: Prior kinematic state $(\boldsymbol{o}_{\text{prior}}, \boldsymbol{v}_{\text{prior}})$.
Input: Points $(P_i^X)_{1\leq i\leq n_X}$.
Input: Points $(P_i^{gr})_{1\leq i\leq n_{gr}}$.
Input: Number of iterations K.
Output: Estimated kinematic state $(\boldsymbol{o}_K, \boldsymbol{v}_K)$.
$(\boldsymbol{o}_0, \boldsymbol{v}_0) \leftarrow (\boldsymbol{o}_{\text{prior}}, \boldsymbol{v}_{\text{prior}})$;
for *k from* 1 *to* K **do**
  for *i from* 1 *to* n_{gr} **do**
    Compute $\phi_{gr\to X}(P_i^{gr}, \boldsymbol{o}_{k-1}, \boldsymbol{v}_{k-1})$;
  end
  for *i from* 1 *to* n_X **do**
    Compute $m_{i,k}$ according to equation 2.50. ;
  end
  Estimate $(\boldsymbol{o}_k, \boldsymbol{v}_k)$ from $(P_i^X)_{1\leq i\leq n_X}$, $(P_{m_{i,k}}^{gr})_{1\leq i\leq n_X}$, $\boldsymbol{o}_{\text{prior}}$ and $\boldsymbol{v}_{\text{prior}}$ by numerical optimization of E (equation 2.49). ;
end

Algorithm 1: Estimating the kinematic state without known correspondences.

2.4 Experiments with iterative closest point

We now describe experimental results associated with our ICP based state estimation algorithm.

2.4.1 Protocol

Our experimental protocol is as follows:

1. We have a SAR image in IDF-ID coordinates. We have other geospatial data of the same geographic location, in ground coordinates.

2. We manually select a few curvilinear structures appearing both in the SAR image and in the reference geospatial data. For instance, these structures may be roads or field boundaries. They approximately cover all parts of the SAR image. We draw them separately on the SAR image and on the reference geospatial data so as to produce two sets of points: one set of IDF-ID coordinates and one set of ground coordinates.
3. We define a prior kinematic state which is the ground-truth kinematic state estimated by GNSS plus some noise. The noise is uniform with 1 km range on each position axis orthogonal to the ground, uniform with 200 m range in altitude, uniform with 10 deg range in elevation angle (i.e. the angle between the velocity vector and the horizontal), uniform with 10 deg range in orientation angle (i.e. the angle between the ground projection of the velocity vector and the geographic axes), uniform with 10 m/s range in speed.
4. We align the two point clouds using our ICP algorithm, thereby estimating the kinematic state. We compare the kinematic states estimated by our algorithm and estimated by GNSS.
5. We repeat steps 3 to 4, 250 times, and construct histograms of the estimation errors before and after correction by our algorithm.

2.4.2 Implementation

Our C++ implementation relies on the numerical optimization library *Ceres*[1] and on the nearest neighbors search library *nanoflann*[2] which implements a kd-tree. Notably, Ceres implements automatic differentiation of the cost function. It computes the gradient of the cost function by applying the rules of differentiation.

2.4.3 Data

Our experiments are based on data acquired with the I-Master airborne SAR imaging radar developed by Thales. As presented in chapter 1.5, I-Master is a Ku band, compact, lightweight airborne surveillance radar capable of StripMap and Spotlight SAR imaging [211, 235]. In these experiments, we have worked with three StripMap images corresponding to distinct geographic locations in the United Kingdom. We call A, B and C the three geographic sites. The SAR images used in our experiments are represented in figure 2.5. The trajectory of the aircraft during imaging was measured using IMU/GNSS but is incoherent with our SAR images, as already explained. As illustrated in figure 2.7, the projection of the IDF-ID observed points on the ground, based on the GNSS/IMU trajectory, and the reference points are shifted by 40 meters (case C) to 90 meters (case A). We have drawn the reference contours using QGIS and Microsoft Bing aerial imagery.

1. `https://github.com/ceres-solver/ceres-solver/`
2. `https://github.com/jlblancoc/nanoflann/`

Case	Number of observation points	Number of reference points
A	23 643	15 499
B	23 044	22 502
C	28 214	17 402

Table 2.3 – Number of observation and reference points. These points are sampled from the contours manually drawn over the observation and the reference respectively.

They are represented in figure 2.6. The lines were discretized into point clouds. The number of observation and reference points are reported in table 2.3.

2.4.4 Results and interpretation

The cumulative distribution of the state errors before and after estimation are reported in figures 2.9 (scene A), 2.10 (scene B) and 2.11 (scene C). For each scene, we report the errors of the following kinematic state parameters (before and after estimation):

— **X** easting,
— **Y** northing,
— **Z** altitude,
— **orientation** angle between the ground projection of the velocity vector and the east direction,
— **elevation** angle between the velocity vector and the up direction,
— **speed** norm of the velocity vector.

Given the distribution of initial errors, the probability that the ICP algorithm converges to a solution is 86 % for scene A, 92 % for scene B and 99 % for scene C. The execution time of aligning the two point clouds with 30 ICP iterations is about 2 seconds with an *Intel Core i7-12700* CPU without parallelism. The estimation is very precise (when we converge, we converge to the same solution) but seemingly inaccurate due to the fact that our ground-truth trajectory and our images do not precisely agree, as previously explained.

Figure 2.8 depicts the reference points and the observation points projected according to the initial and final states. We observe that the reference points and the observation points projected according to the final state coincide.

However, note that our test cases are relatively easy since the reference and the observation point clouds only contain common structures. In a real world setting, the reference point cloud may contain more structure than the observation point cloud.

2.5 Discussion

— In our model, the noise happens in coordinate domain X (i.e. TCA-CD or IDF-ID). We have made this assumption because, in our context, we observe X coordinates and not ground coordinates.

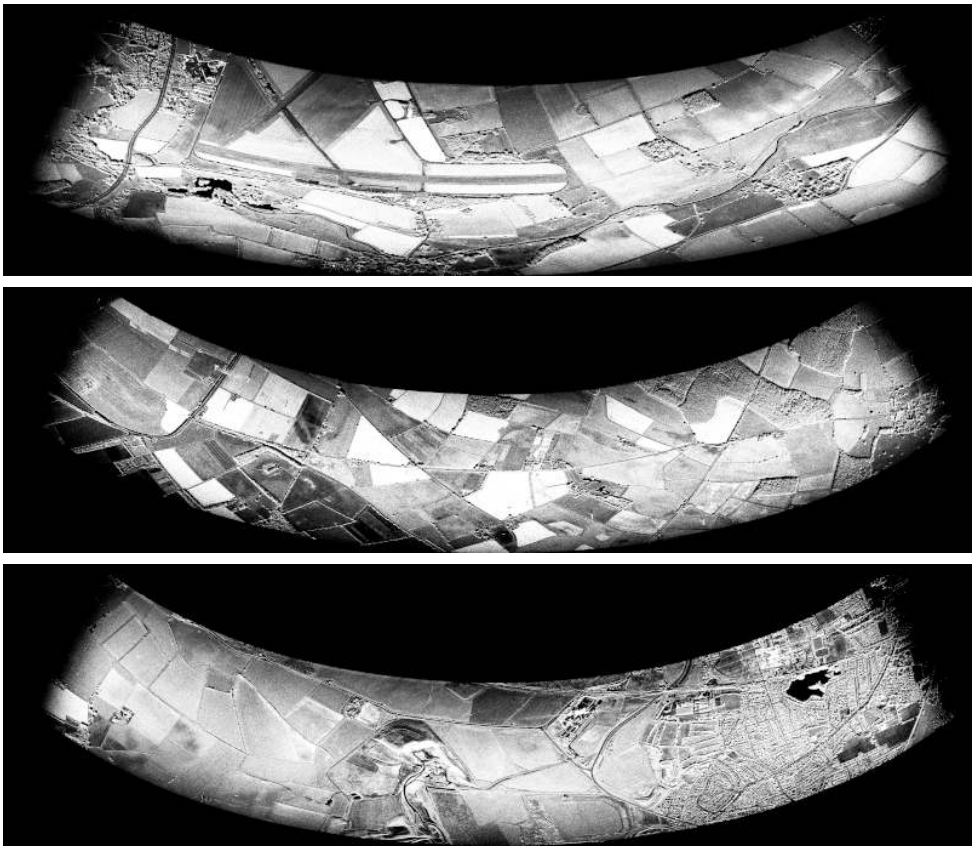


Figure 2.5 – The I-Master SAR images used in our experiment. From top to bottom: cases A, B and C. Horizontal axis is Doppler frequency. Vertical axis is distance.



Figure 2.6 – The reference contours for cases A, B and C. The background image is from Microsoft Bing.



Figure 2.7 – Shift between the reference points and the projection of the observed points using the GNSS trajectory estimate. Green: lake boundary drawn on the optical reference. Blue: lake boundary drawn on the SAR observation and projected onto the optical reference according to the ground-truth position and velocity of the device.



Figure 2.8 – Result for scene B. Red: projected SAR points before ICP. Green: projected SAR points after ICP. Blue: reference ground points.

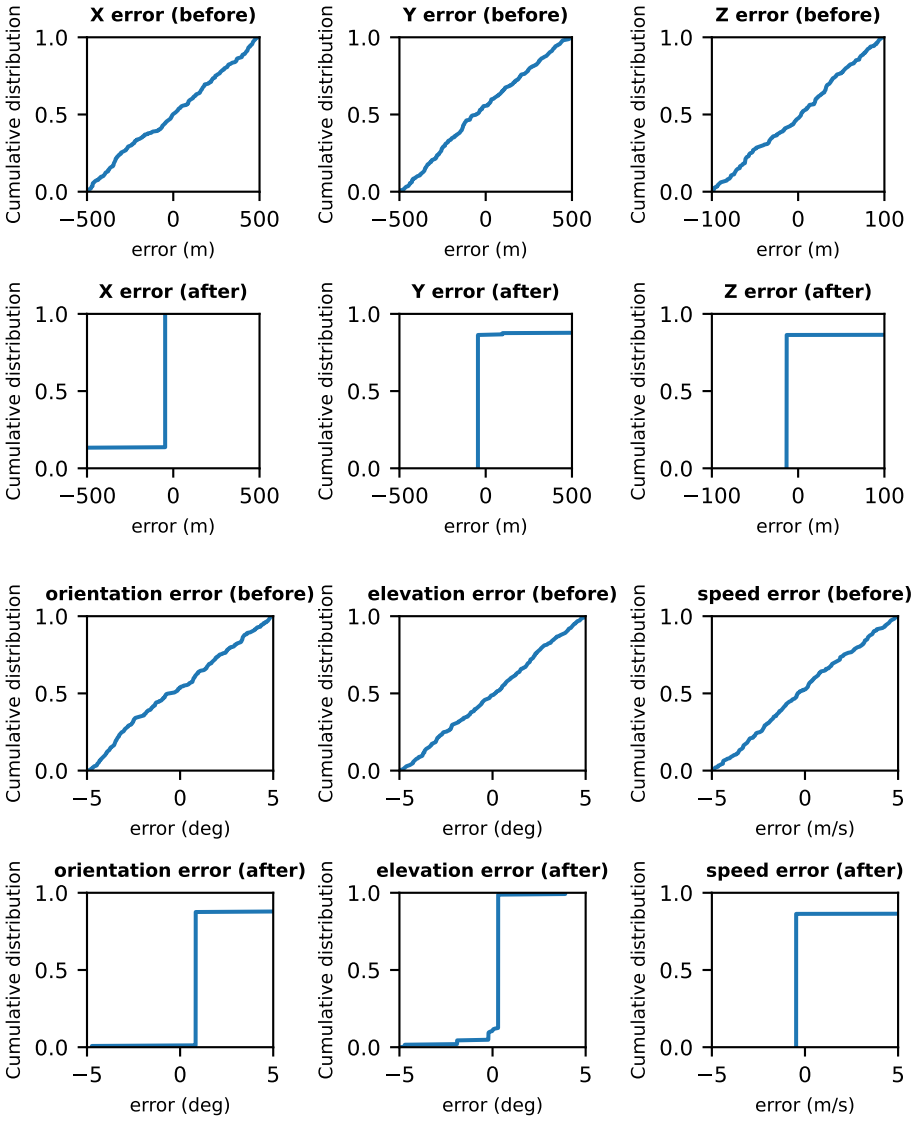


Figure 2.9 – Estimation errors cumulative distribution for scene A.

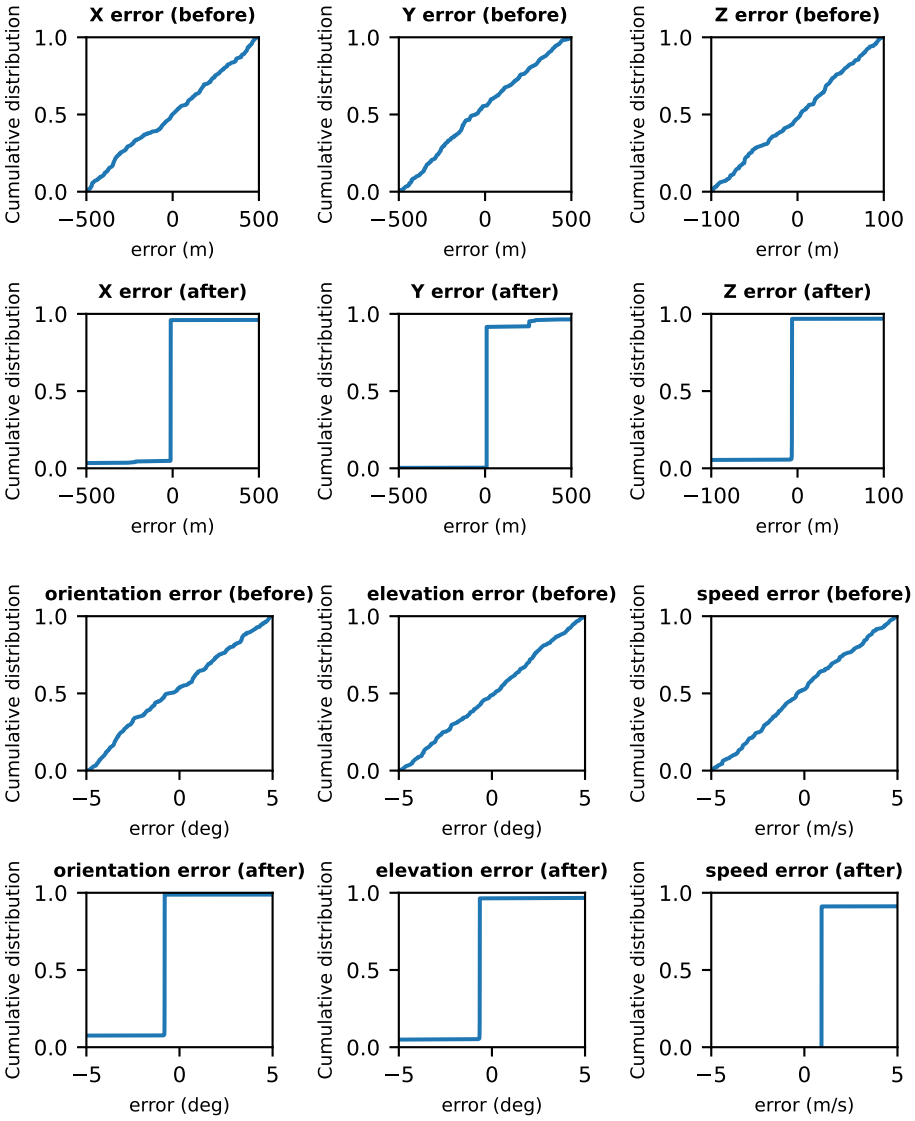


Figure 2.10 – Estimation errors cumulative distribution for scene B.

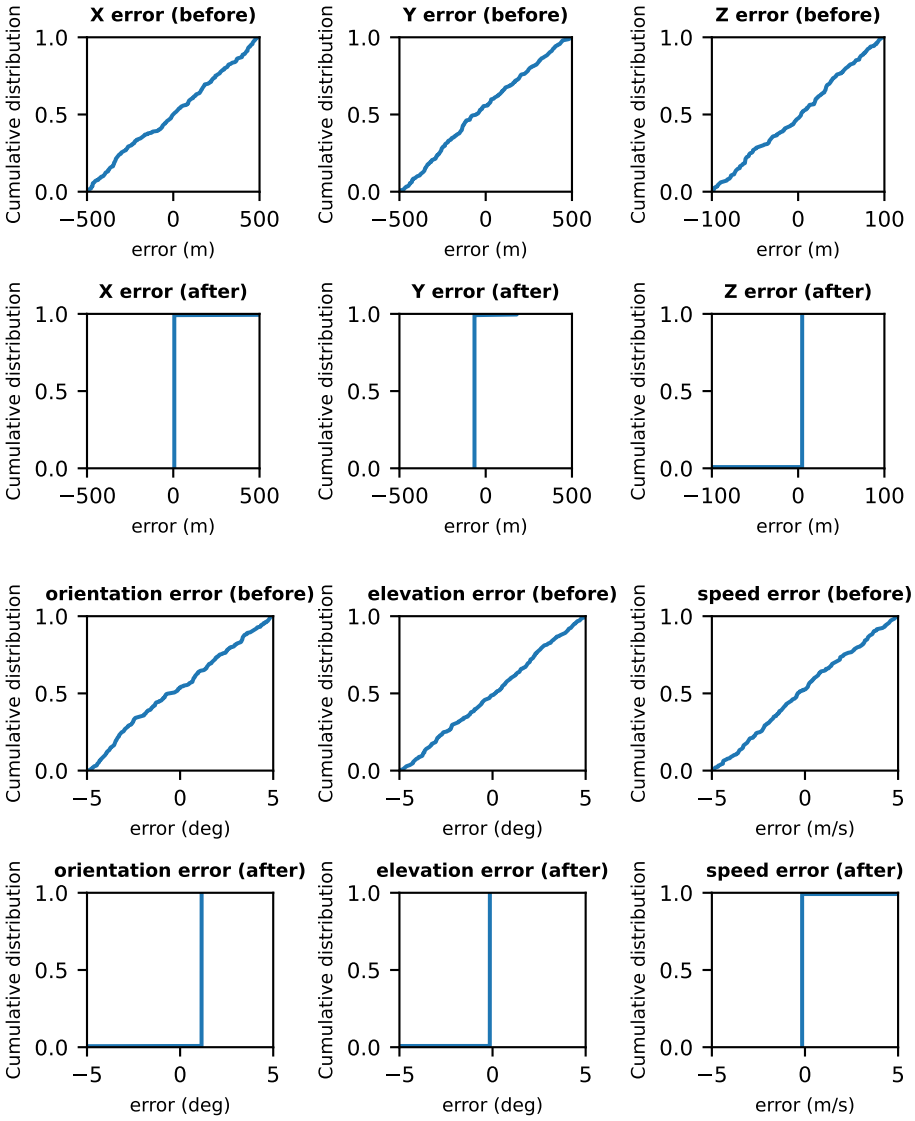


Figure 2.11 – Estimation errors cumulative distribution for scene C.

— We have not specified how to detect the observed points. A full navigation pipeline should include a means of detecting the same curvilinear structures simultaneously in the SAR image and in the reference geospatial product despite the modality mismatch. This process might rely on contour detection, semantic segmentation or Automatic Target Recognition (ATR).
— Given initial estimates of the altitude, of the depression angle (the angle between the horizontal and the velocity vector) and of the norm of the velocity vector, we could resample the SAR image from IDF-ID coordinates to pseudo-ground coordinates. The pseudo-ground coordinates coincide with the ground coordinates when the kinematic state estimate is the actual kinematic state. Performing pattern recognition may be easier in pseudo-ground coordinates than in IDF-ID coordinates. After having acquired points in pseudo-ground coordinates, we can transform them back to IDF-ID coordinates. So doing, the coordinates no longer depend on the initial kinematic state estimate.
— We have modeled the ground as flat. Modifying our algorithm so that it accommodates terrain relief and roundness is straightforward. The reference points should be defined by their spatial cartesian coordinates and not by their ground coordinates. The relief may be represented as a Digital Elevation Model (DEM). However, relief may also degrade the quality of the SAR image and so hinder pattern recognition.
— In the method that we have presented, there are two point clouds (i.e. one in either coordinate domain). In order to skip searching the nearest neighbors and save computation time, we could consider a point cloud in one domain and a distance map in the other domain. A distance map associates each coordinates tuple to its distance to the nearest point of the cloud. The distance map can be sampled in the form of a raster and differentiably interpolated.
— For the optimization algorithm to converge to the desired solution, the initial solution must lie in the attraction pool of the desired solution. Otherwise, it may converge to an undesired or trivial solution.

2.6 Complete estimation system

Building upon the ideas developed in the previous section, we now investigate a more complete state estimation algorithm, one that includes the detection of co-occuring patterns in the observation and in the reference. Specifically, we propose to detect co-occuring points from multimodal contour detection.

2.6.1 Overview of the estimation procedure

The state estimation procedure is sketched in figure 2.12. First, we train a neural network to predict, from an optical image, the contours that would be detected on a SAR image. This requires geocoded SAR and optical images. Note that we consider that the reference is an optical image only for the sake of the illustration clarity. Other geospatial

data modalities (e.g. vector, SAR) may suit as well. The first step need be performed once and for all. In order to prepare a mission, given a reference optical image, we produce a contour distance map. This map associates to every ground coordinates tuple the distance to the nearest contour. This second step need be performed during mission preparation. Finally, during the mission, we detect the contours on the SAR image and we estimate the kinematic state.

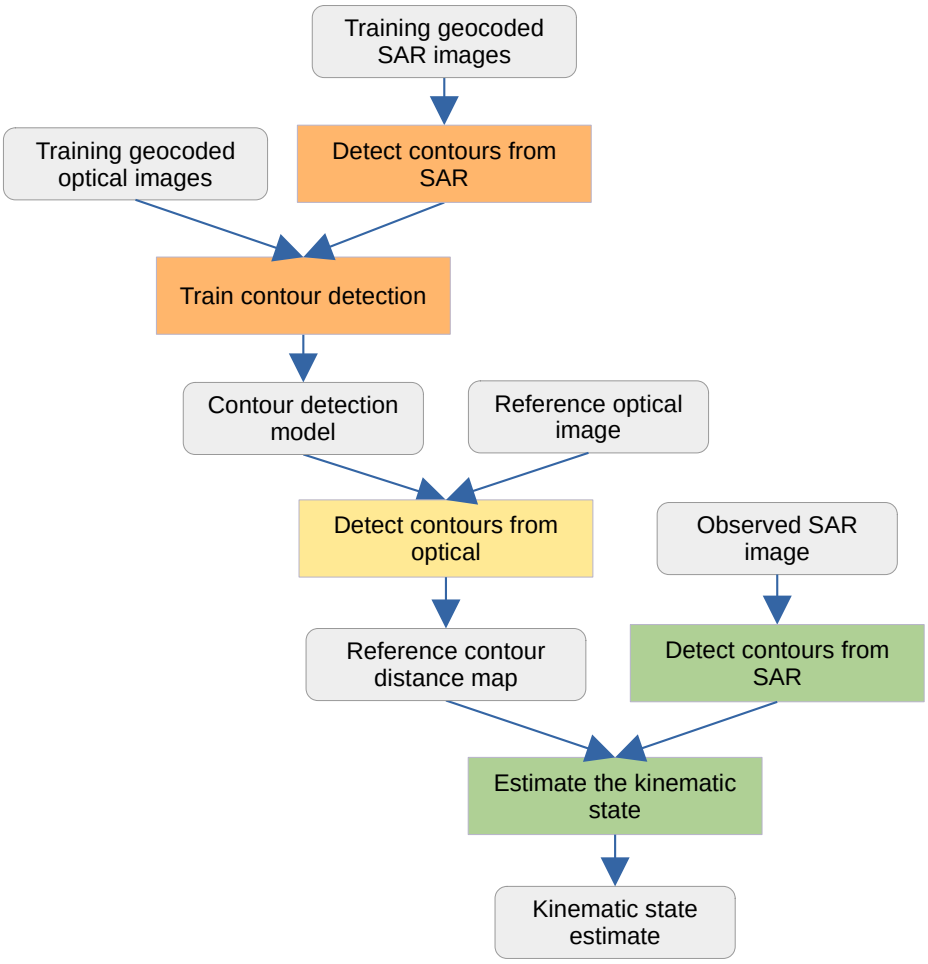


Figure 2.12 – Alignment procedure. Orange refers to an infrastructure task. Yellow refers to an offline mission-preparation task. Green refers to an online mission task.

2.6.2 Training to detect contours

Given a SAR image, we generate training distance maps as follows:

1. We detect contours on the SAR image using the ratio-of-averages (see section B.3.1).

2. We thin the contours using non maxima suppression.

3. We threshold the contours to obtain a binary-valued map.

4. We compute the distance transform of that binary map. In other words, for every pixel, we compute the distance to the nearest contour.

5. We upper bound the distance transformed map to a given distance. Upper-bounding the distances is required because the neural networks have a finite field of view.

6. We resample to some working resolution.

Note that we may enhance this simple contour detection technique (see section B.3). Having produced a distance map from the SAR image, we now resample the training optical images to the working resolution and cut out pairs of optical images and their SAR derived distance transforms. We get pairs of training patches. A sample training pair is represented in figure 2.13. Then, we train a neural network to predict the distance transform from the optical images. We select the standard U-Net architecture [183]. It is a fully convolutional neural network. The first part of the network reduces the resolution while increasing the number of channels. The second part of the network increase the resolution while reducing the number of channels. The two parts are connected with skip connections. As a cost function, we use the sum of squared differences. We optimize the weights using the stochastic gradient algorithm with a learning rate of 0.015. We train for six epochs with a batch size of 32. We use arbitrary rotations for data augmentation. The training procedure is represented in figure 2.14.

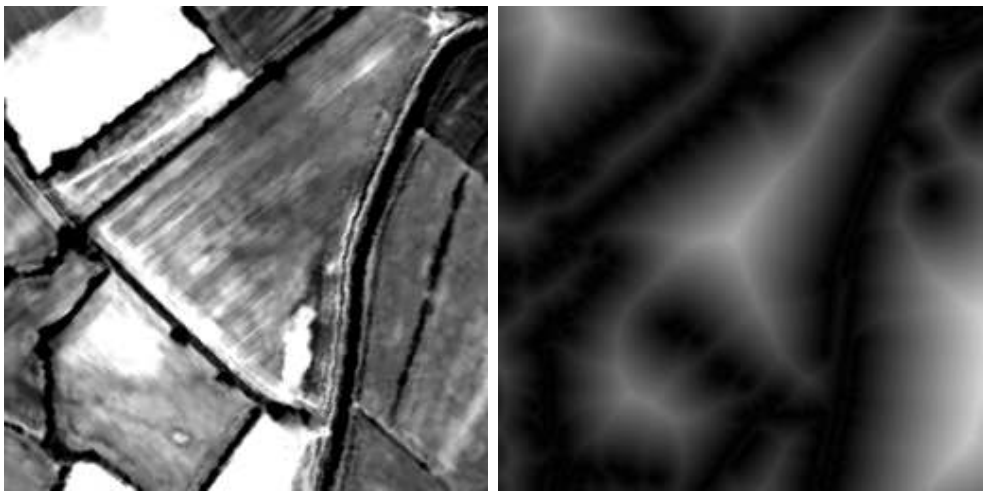


Figure 2.13 – Sample training samples. Left: optical image. Right: distance transform of contours detected from SAR.

2.6.3 Building the navigation data

Given an optical image representing the place of the mission, we sweep a window with consecutive overlap. At each window step we transform the optical patch into a distance map patch. We accumulate the distance map patches into a buffer. In order to avoid blocking artifacts, we accumulate with an apodization weights. An example map is represented in figure 2.15. In figure 2.16, we have plotted the value of the predicted contour distance along a line that is perpendicular to a contour. We observe that the predicted and theoretical contour distances are approximately equal.

2.6.4 Navigation

Given a SAR image in IDF-ID geometry, we must first identify the part of the image that contains actual echoes, since it depends on the actual device trajectory relative to the ground. We accomplish this by:

1. resampling the SAR image to a lower resolution,
2. computing a binary partitioning tree [19] (cf section B.4.1),
3. cutting the hierarchy of the tree at two components,
4. identifying the component with the highest mean intensity and
5. eroding the footprint of that class.

Other techniques such as graph cuts could perform that job as well. An instance of detecting the useful part of the SAR image is reported in figure 2.17.

We must now detect contours in the SAR image. For this:

1. We resample the useful part of the SAR image from IDF-ID to pseudo-ground coordinates (relying on a prior kinematic state estimate).
2. We detect contour points, in pseudo-ground coordinates, using the same algorithm used at training.
3. We subsample the set of contour points.
4. We transform back the contour points from pseudo-ground to IDF-ID coordinates.

We end up with a set of points in the IDF-ID domain. The reason behind the intermediate passage by the pseudo-ground domain is that pattern recognition algorithms (contour detection in this case) may have better performance in the pseudo-ground domain than in the IDF-ID domain. This is because the aspect of a pattern in pseudo ground coordinates is independent of the kinematic state of the device apart from a rotation and a translation.

Finally, we shall attempt to estimate the kinematic state as a solution to the following optimization problem:

$$\underset{\boldsymbol{o},\boldsymbol{v}}{\text{minimize}} \sum_{i=1}^{n} f\left(\phi_{rd\to gr}\left(P_i^{rd}, \boldsymbol{o}, \boldsymbol{v}\right)\right) \tag{2.51}$$

Here, f denotes the map of distance to contours computed at the stage of preparing the mission (see section 2.6.3). The cost function is differentiable but exhibits many local

minima so we first perform grid search (i.e. we sample the values of the cost function regularly in the space of kinematic states) and we secondly perform gradient-based iterative search from the previous optimal kinematic state. Since the computation time of grid search is exponential in the number of dimensions, we perform it only for a part of the kinematic state, specifically the translations. In figure 2.18, we report an example result of grid search optimization for translation. We show the projected points before and after optimization. The initial translation error is 1.8 km. The rest of the kinematic state is the true kinematic state.

2.6.5 Results

Aggregate results of 400 kinematic state estimations are represented in figure 2.19. The blue curve represents the distribution of the initial kinematic state estimation errors, which is uniform with 3 km range in translation, 400 m range in altitude and 4 m/s range in each component of the velocity vector. The orange curve represents the distribution of errors after grid search. The green curve represents the distribution of errors after iterative optimization. In these experiments, we observe an improvement of the translation estimates only. It is mostly accomplished by grid search.

In the experiments with ICP from section 2.4, we successfully estimate the full kinematic state, including the altitude and the velocity vector. However, in the experiments from this section, we do not improve the precision of the velocity vector and altitude estimates. A fundamental explanation is that the reference now contains structures that are absent from the observation, so the attraction pool of the desired solution of our optimization problem is smaller. An additional explanation is that we detect contours algorithmically and not manually, using a primitive algorithm which is noisy and densely detects false positives in forested areas. One may consider a less noisy detection algorithm such as LSD [76, 138] or learning a human contour annotation.

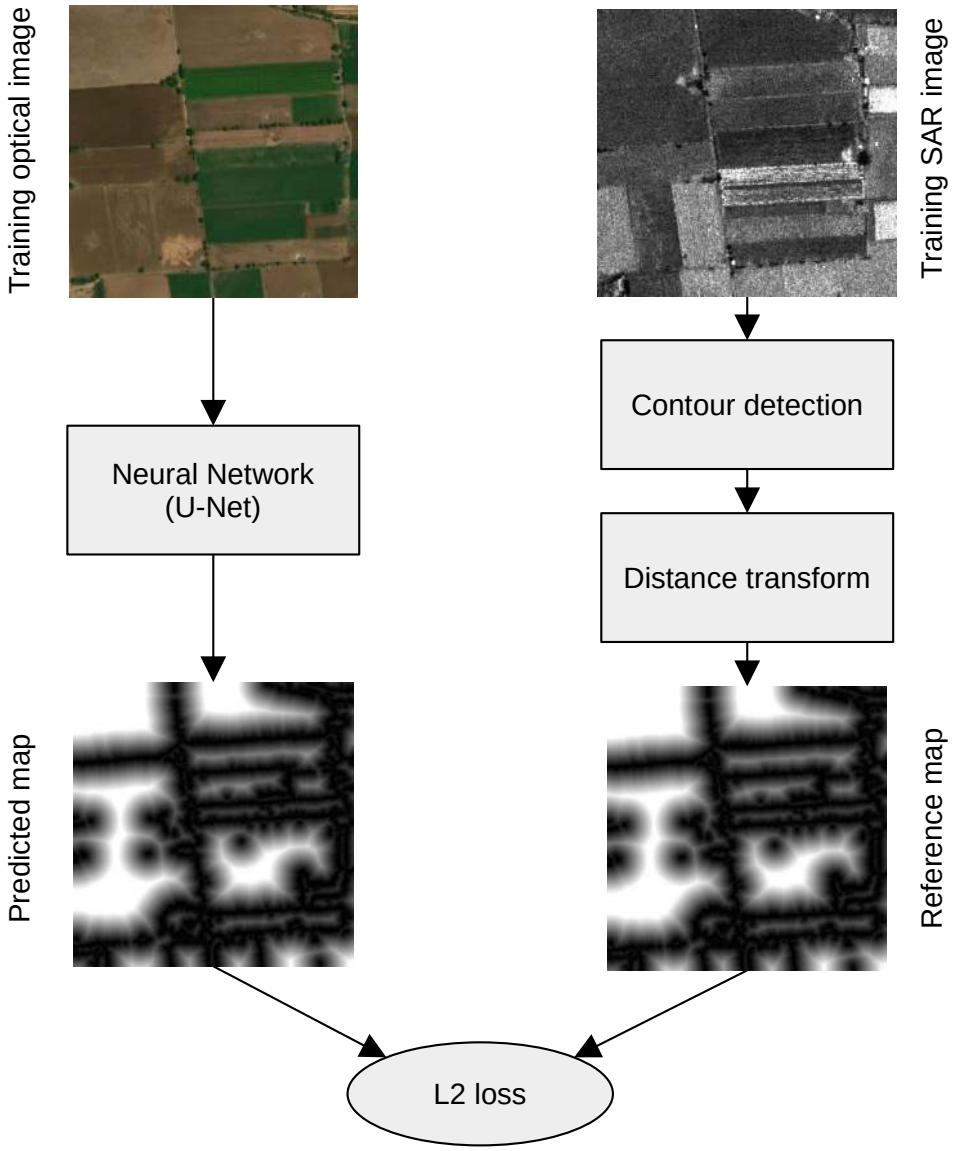


Figure 2.14 – Training the contour detection network. The illustrative optical image is from Microsoft Bing. The illustrative SAR image is from Capella Space.

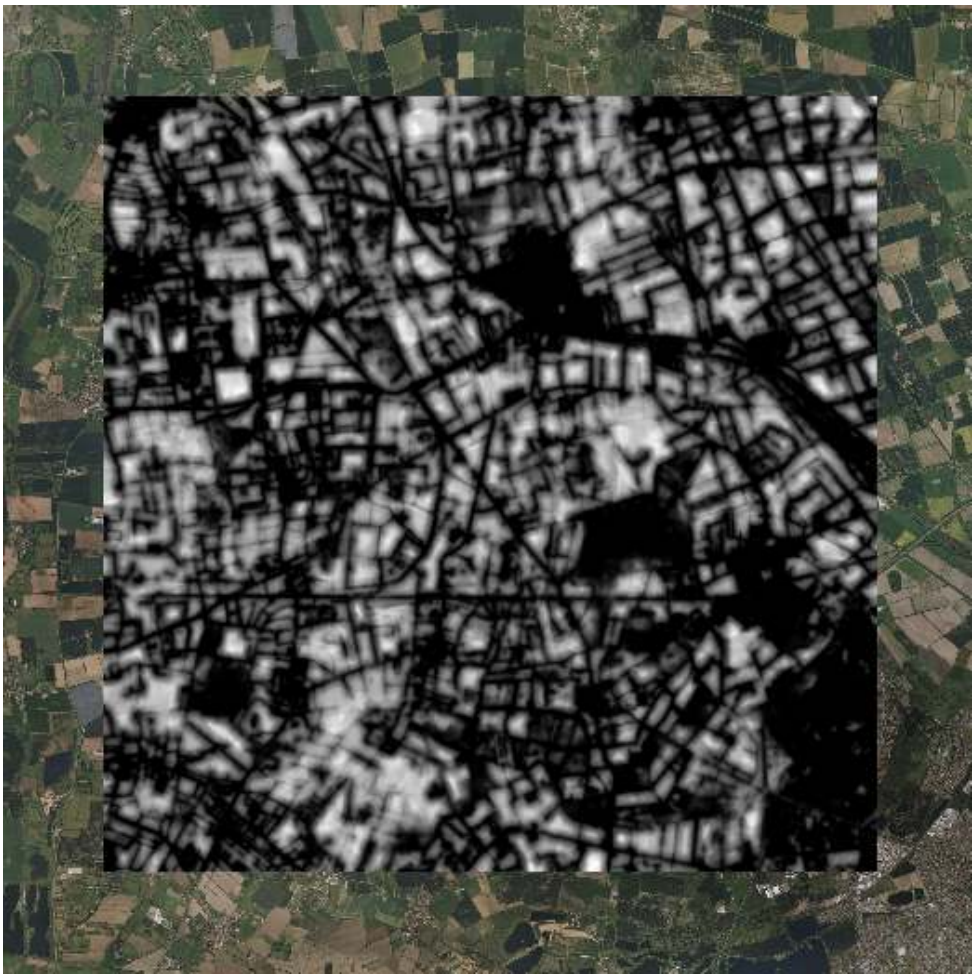


Figure 2.15 – A map of contour distance.

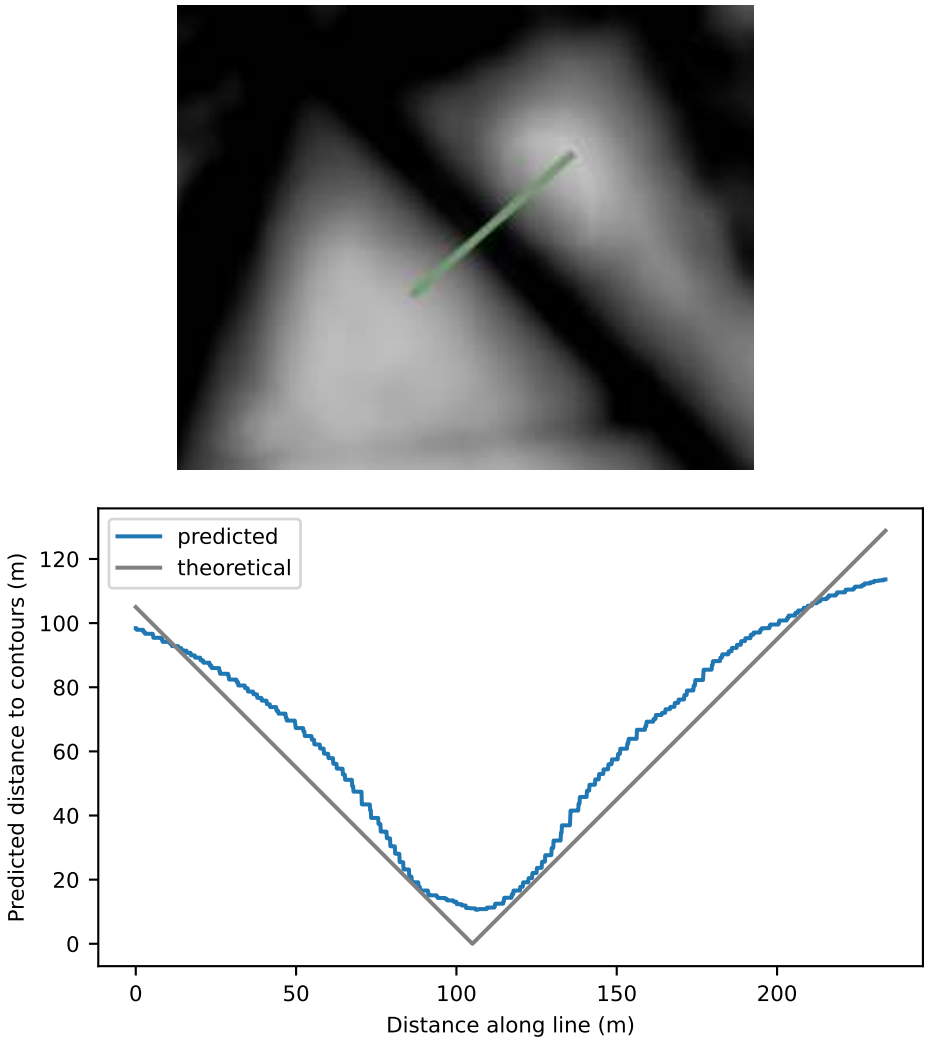


Figure 2.16 – A predicted contour distance profile. Top: a line over a predicted distance map. Bottom: the value of the predicted distance along this line.

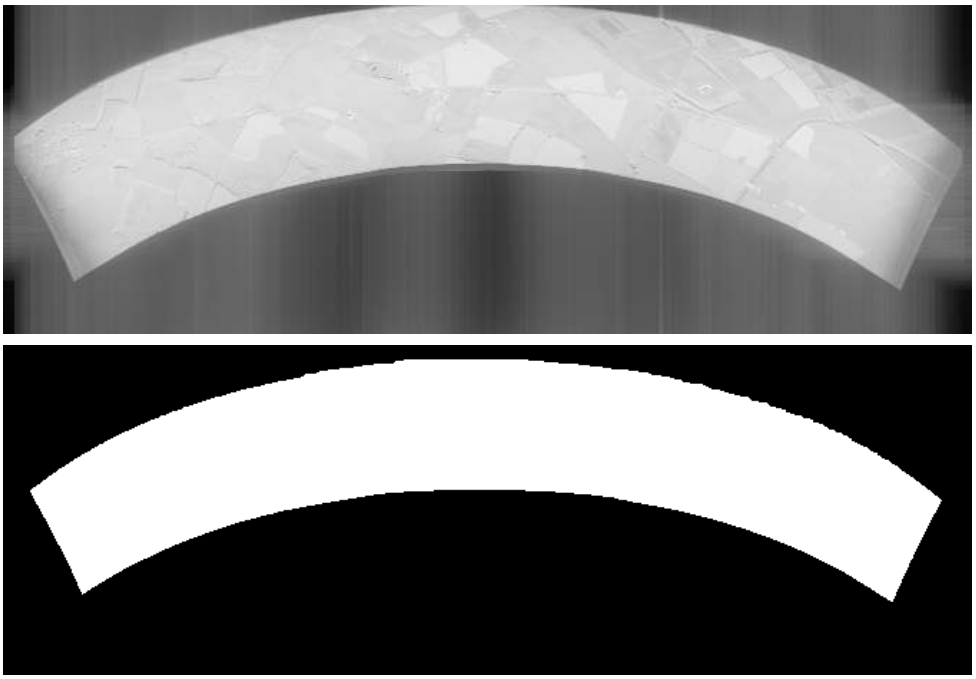


Figure 2.17 – Detecting the useful part of the SAR image using a binary partitioning tree.

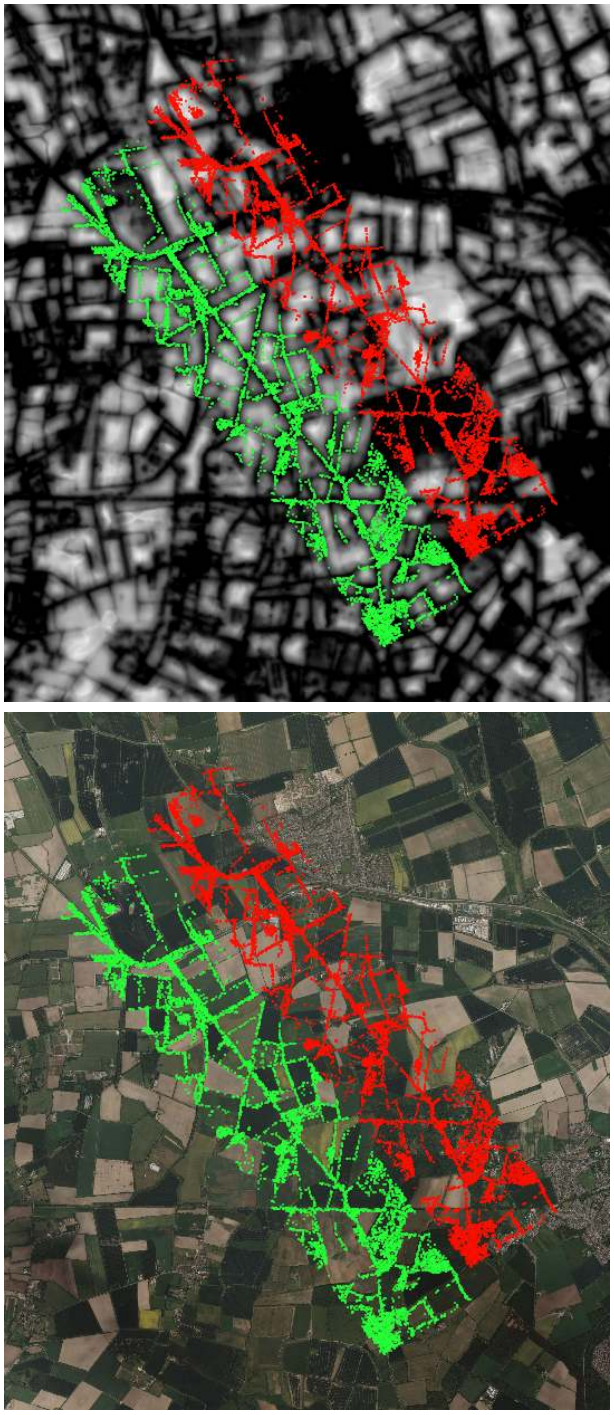


Figure 2.18 – Estimating a translation. Contours projection before alignment (red) and after (green), superposed over the distance map background (top) and the optical background (bottom).

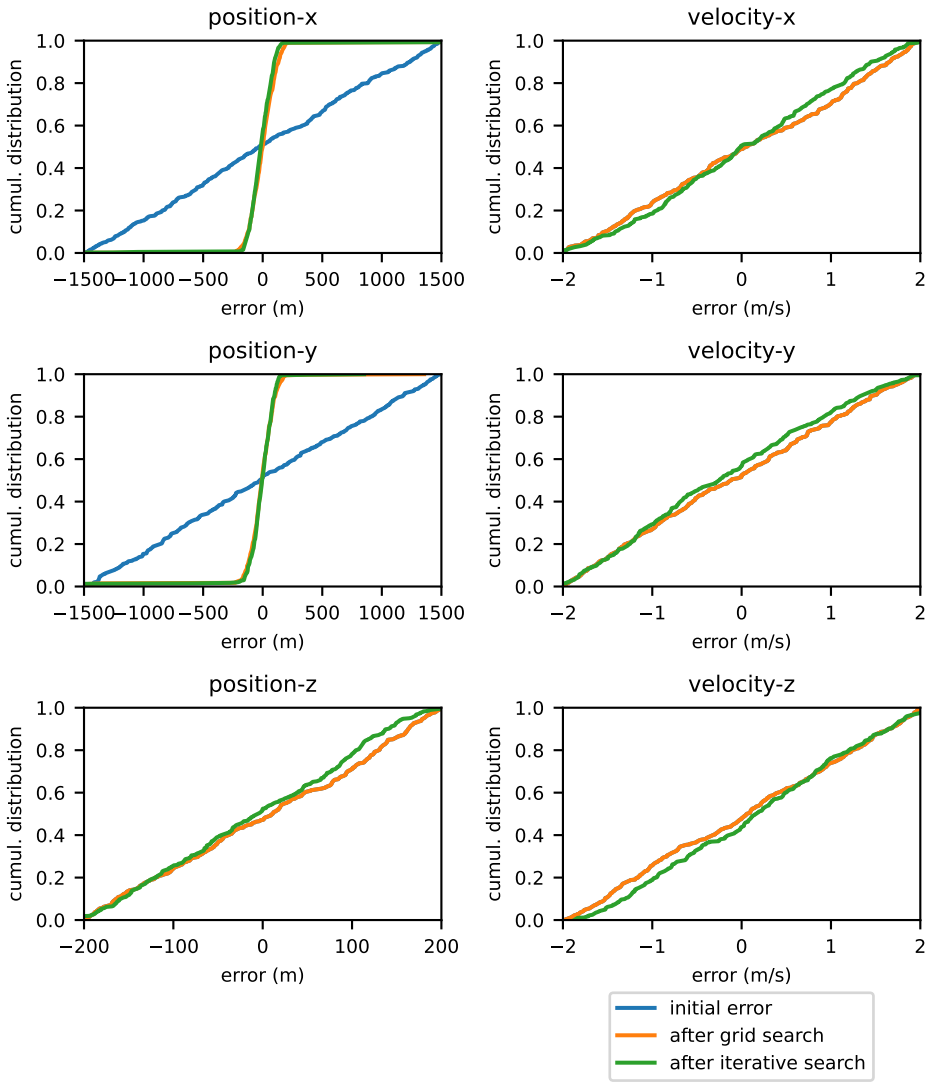


Figure 2.19 – Kinematic state estimation errors.

2.7 Conclusion

In this chapter, we have shown that we can convert between TCA-CD, IDF-ID and ground coordinates if we know the kinematic state. Conversely, we can determine the kinematic state from a set of correspondences between ground coordinates and either TCA-CD or IDF-ID coordinates. As well, given a set of ground coordinates and a set of TCA-CD or IDF-ID coordinates, if these two sets of points represent the same underlying physical structures, we can determine the kinematic state. We have finally illustrated a complete kinematic state estimation pipeline but it has experimentally only succeeded in estimating part of the kinematic state. In this chapter, we have bridged the modality gap by separately detecting points or contour. In the next chapter, we propose a kinematic state estimation method based on a multimodal similarity metric.

Chapter 3

State estimation based on a multimodal similarity metric

In the previous chapter, we addressed the fact that the reference and the observation have different modalities by considering that we could extract from them structures such as contours and point clouds. In this chapter, we present a kinematic state estimation system based on a multimodal similarity metric, which is a means of testing whether two points of two pieces of geospatial data of different modalities correspond to the same geographic point. We first define local descriptors, which are representations of local neighborhoods that can be compared easily. We report the performances of training local descriptors on our multimodal data. We then use these local descriptors for the purpose of navigation. We consider the estimation of a translation first and finally the estimation of the complete kinematic state. Parts of this chapter are inspired from our publication [149].

3.1 Local descriptors

In computer vision, a descriptor is a representation of an object. This representation is easier to process than the original object. Extracting information from an object is reduced to extracting information from its descriptor. Comparing two objects is reduced to comparing their respective descriptors. For instance, comparing two cities can be reduced to comparing their areas and their populations. In our context, a descriptor represents a patch of geospatial data and is the means of measuring the likelihood that these patches represent the same geographic point.

3.1.1 Description procedure

In this section, we describe how to produce a local descriptor given geospatial data and a geographic window. We decompose this process as follows:

1. constructing a local raster patch,
2. computing a descriptor from this raster patch.

The raster patch construction pipeline is handcrafted. We mean that it results from human engineering and not from an algorithmic search. The patch description pipeline may be handcrafted or learned. A geographic window defines an oriented local geographic neighborhood. It is the combination of:

— a center geographic point,
— an orientation angle relative to the geographic axes,
— a diameter,
— options for flipping the window about its axes. This serves for data augmentation.

3.1.2 Constructing raster image patches

We may also refer to the procedure of constructing a raster image patch as rendering (when the underlying geospatial data are vector) or resampling (when the underlying geospatial data are raster). Given geospatial data and a geographic window, producing a raster patch involves the following steps.

From raster data

For raster geospatial data, such as optical and SAR images, typical steps are:

1. **Despeckling.** When the underlying geospatial data are a SAR image, despeckling it may be beneficial to the discriminativeness of the descriptors. For learned descriptors, we have experimentally observed that it was better not to despeckle the SAR images. Contour-based handcrafted descriptors may benefit from despeckling. Note that the later resampling of the patch, at step 4, generally reduces the patch resolution and averages out some of the speckle. We provide an overview of the families of despeckling algorithms in section B.2.
2. **Conversion to grayscale.** In the case of a multiband image (e.g. RGB), we have experimentally observed that descriptors were more discriminative when the image was converted to grayscale. On the one hand, this operation destroys potentially useful information. On the other hand, it reduces the dimension and the variability of the patches. In our experiments, the later effect appeared to dominate over the former. We consequently opted to produce only monoband patches.
3. **Logarithm.** The SAR images have a high dynamic range. In order to flatten the histogram, patches of SAR images are converted to logarithmic intensity.
4. **Resampling.** We resample the piece of geospatial data covered by the window into a raster patch of fixed size. An affine transform relates the patch pixel coordinates and the geospatial data intrinsic coordinates.
5. **Normalization.** We have observed experimentally that more discriminative descriptors were achieved when the patch was normalized. We have compared various normalization techniques (e.g. histogram normalization, L2 normalization) and have found out that zero mean and unit variance normalization provided the best results in terms of descriptor discriminativeness.

In practice, we can factorize parts of this canonical pipeline such as despeckling and grayscale conversion.

From vector data

For vector geospatial data, such as OpenStreetMap, typical steps are:

1. **Rasterization.** We draw the vector data onto a raster patch. There is no need of any resampling since the patch directly has the requested geometry. In practice, the drawing may be performed using an existing software library such as libosmscout [1] or Mapnik [2]. Such tools enable to customize the drawing. The user can select which items to draw and customize their appearance. Notably, one can disable rendering text labels and icons, disable zoom-conditional drawing, replace dashed lines with continuous lines, remove line terminal markers, etc.
2. **Conversion to grayscale.** The patch may be converted to grayscale. The arguments presented for raster data apply here as well.

Since the patch is produced by an algorithm, it is already standardized so we do not normalize it further.

The original vector data may contain more information than is rendered into the raster patch. How to render the vector data so that the descriptors are most discriminative? We have observed that most of the useful information is contained in the geometric shapes whereas that the annotations contain little useful information. Some annotations, such as street numbers, do not contribute to recognizing a place observed through SAR imaging.

3.1.3 Handcrafted descriptors

After producing a raster patch representing some local geospatial data, one computes a more easily processable representation called a descriptor. In this section, we describe handcrafted local multimodal descriptors reported in the literature. They generally fall into two categories. Some are based on histograms of contour orientation and some are based on self-similarity.

Histograms of contour orientation

In computer vision, histograms of contour orientation are widely used as descriptors. Well-known instances include Scale Invariant Feature Transform (SIFT) [145] and Histogram of Oriented Gradient (HOG) [53]. Producing and using descriptors based on contour orientation histograms involves the following steps:

1. **Detecting contours in the raster patch.** This is usually accomplished with finite differences, possibly preceeded by a low-pass filter (i.e. Sobel filtering) and possibly succeeded by non maxima suppression (i.e. Canny edge detector). In our case, we require two edge detectors (one for each modality) that ideally detect the same contours despite the modality gap. If the patches are sufficiently smooth (i.e. the SAR images are despeckled or multi-looked) then finite differences can

1. `https://github.com/Framstag/libosmscout/`
2. `https://mapnik.org/`



Figure 3.1 – Rectangular, polar and log-polar masks.

be used. Otherwise, we may ressort to techniques such as the ratio of averages or phase congruency. These are presented in appendix B.3.

2. **Dividing the raster patch into cells.** We divide the raster patch into cells which generally do not overlap. Instances of rectangular, polar and log-polar grids are represented in figure 3.1.

3. **Computing a histogram of contour orientation for each cell.** The histograms of contour orientation are characterized by their number of bins. The orientation angles may be defined modulo π or modulo 2π. For multimodal descriptors, the orientation angles should be defined modulo π. The contribution of each point to a histogram bin may be proportional to the contour strength or constant. We may low-pass filter the histograms to attenuate the quantization noise.

4. **Concatenating and normalizing the histograms.** The histograms of all cells are concatenated and the concatenation is normalized, typically with L2 norm. The normalized concatenation of local orientation histograms is the descriptor.

5. **Comparing descriptors.** The similarity metric may typically be the L2 norm or the correlation coefficient.

We now describe some specific instances of descriptors that are based on contour orientation histograms. Sometimes, these descriptors are bundled with keypoint detection algorithms. In this case, we only discuss the description and not the keypoint detection since only the description is relevant in this chapter. SIFT [145] is one of the most widely used descriptor. It is based on a 4×4 rectangular grid with 8 orientation bins. SIFT is tailored for matching keypoints. HOG [53] is tailored for detecting objects. Dellinger *et al* proposed SAR-SIFT [62]. This is an adaptation of SIFT to SAR images. Contour detection relies on a separable ratio of averages. OS-SIFT [239] enables to match optical and SAR images. Contours from the optical image are extracted with Sobel filtering and contours from the SAR image are detecting using the ratio of averages. The grid is log-polar with 17 cells and each histogram has 8 bins. For matching optical and SAR images, Xiang *et al* proposed a descriptor with a phase congruency based contour detection [240].

Local self-similarity

Self-similarity was introduced in [196]. It consists in comparing each patch with itself and in comparing the correlation maps. In other world, we compute one auto-correlation map for each patch and we compare the auto-correlation maps using simple statistical metrics. It involves the following steps:

1. **Selecting a subpatch.** We define a subpatch from the patch that we want to describe. Typically, the subpatch is the central part of the patch.
2. **Computing the correlation map.** We compute the correlation between the subpatch and the rest of the raster patch. The particular correlation metric may be the normalized cross-correlation or the mean squared distance for instance.
3. **Pooling the correlation.** We define a grid on the raster patch such as the ones illustrated in figure 3.1. We reduce the correlation values over each cell to a single value. The reduction function may be the average or the maximum.
4. **Normalizing the descriptor.** We concatenate the reduced correlation of all pools and normalize the concatenation, for instance with soft-max.
5. **Comparing the descriptors.** We compare the descriptors using such a metric as the L2 distance or the normalized cross-correlation.

Local self-similarity was applied to the registration of SAR and optical images in several works [255, 254, 245]. They differ by the particular reduction function.

3.1.4 Learned descriptors

The descriptors that we have described in the previous section are handcrafted. In this section, we describe how to learn local descriptors. We first review the state of the art of learned multimodal local descriptors. We secondly describe the particular learned descriptors that we shall use in our registration process.

State of the art

The development of learned descriptors for computer vision has been especially driven by the tasks of face identification and local feature matching. Early methods such as eigenfaces [202, 221] and fisherfaces [17] relied on dimensionality reduction techniques such as principal components analysis (PCA) and linear discriminant analysis (LDA) to estimate a linear projection of normalized image patches into descriptors. A strong limitation of these methods is their inability to learn a complex transformation of input patches. This was alleviated by using multi-layer neural networks, as introduced in [44] for face identification and later applied to local feature matching in [101]. These two algorithms rely on a siamese or pseudo-siamese neural network trained to predict whether a pair of image patches is a match. Their descriptors are compared using the L2 distance. DeepFace [217] and MatchNet [248, 6, 260] introduced learned metrics, cross-entropy loss, hinge loss and more complex architectures. A critical part of the training of these

previous algorithms is the sampling of pairs of image patches. Given a set of positive pairs, many more negative pairs can be formed. While previous methods sample negative pairs randomly among those that can be formed, DeepDesc [29] proposed to focus on the hardest ones by retaining them from one epoch to the next. The methods mentioned so far all use a pairwise loss, that is to say each term of the cost function depends on a pair of descriptors and on a label indicating whether the pair is a match or not. Instead, with a contrastive loss, each term of the cost function depends on a reference descriptor called an anchor, a positive descriptor and at least one negative descriptor. Training descriptors with a contrastive loss was introduced in [233], FaceNet [193] (triplet margin loss) and PN-Net [14] (SoftPN loss). L2-Net [218] makes use of several negatives for each anchor instead of relying on a single one as commonly reported in the literature. With a contrastive loss, negatives can be selected according to a hard negative mining strategy. One such strategy is anchor swap. It was introduced in [13] and consists in inverting the anchor and the positive whenever the negative is closer to the initial positive than to the initial anchor. Another strategy is hard negative mining within a minibatch, which was introduced in [152] and saves computations by reusing descriptors.

Several descriptors for matching SAR and optical image patches have been reported in the literature. In general, they are straightforward implementations of the methods already mentioned. In [150, 93, 94], pseudo-siamese networks were used to describe SAR and optical image patches. Their descriptors are either compared with the inner product or with a learned metric. The generation of hard negative patches using a generative adversarial network (GAN) was reported in [92, 95]. These previous works focus on the task of matching ground control points for spaceborne SAR imagery, for which a rough alignment is already known as input, whereas, in our use case, an initial alignment is not required. SAR versus optical descriptors that can be compared using the L2 distance were reported in [32, 135]. The use of a standard Lp metric instead of one based on a neural network enables the use of data structures such as kd-trees [20] for efficiently retrieving neighbors in descriptor space.

Neural network architecture and training

The learned local descriptors that we use in this work are based on a pseudo-siamese neural network whose architecture is represented in figures 3.3 and 3.2. It is composed of two branches that have an identical architecture (cf figure 3.3) but different weights. The architecture of each branch is inspired from VGG [200]. VGG is a basic plain neural network, with a convolutional backbone, characterized by a doubling of the number of chanels and a halving of the spatial resolution every few layers. The layers are composed of a convolution operation and a ReLU activation function. We have tailored the architecture to our needs. The descriptor dimension is 125. A notable feature of our architecture is that it is composed of convolutional layers only. The typical densely connected head is replaced with a convolutional layer with a 1×1 kernel. This architecture is reminiscent of traditional handcrafted descriptors that are composed of a grid of sub-descriptors. This replacement enables a reduced number of parameters but prevents the descriptors from

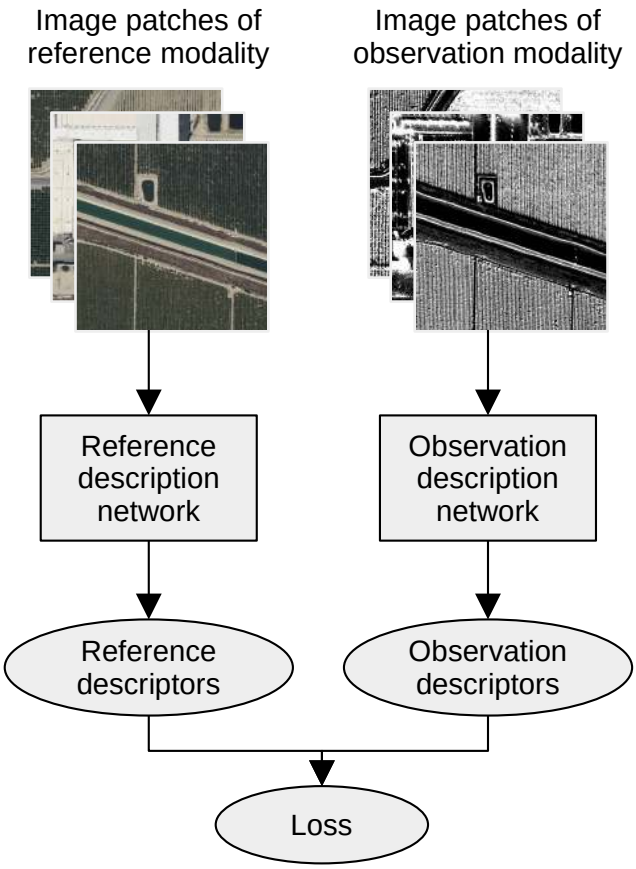


Figure 3.2 – Pseudo-siamese neural network architecture and training.

being invariant to rotation.

In order to train a description model, we require paired raster patches of both modalities. We generate them from the training geospatial data as mentioned in section 3.1.2. We sample locations uniformly (with stratified sampling) over the geographic area marked for training and, for each point, we produce paired image patches with different orientations and flips for data augmentation. We assign each piece of geographic data either to training or to testing. Sample pairs of raster patches are represented in figure 3.4.

The similarity of two descriptors is measured using the L2 norm. We train the neural network using a triplet margin contrastive loss l defined as follows.

$$l(d_i^Q, d_i^R, d_j^Q, d_k^R) := l^Q(d_i^Q, d_i^R, d_j^Q) + l^R(d_i^Q, d_i^R, d_k^R) \tag{3.1}$$

with

$$l^Q(d_i^Q, d_i^R, d_j^Q) := \max\left\{0, \delta + \|d_i^Q - d_i^R\|^2 - \|d_j^Q - d_i^R\|^2\right\} \tag{3.2}$$

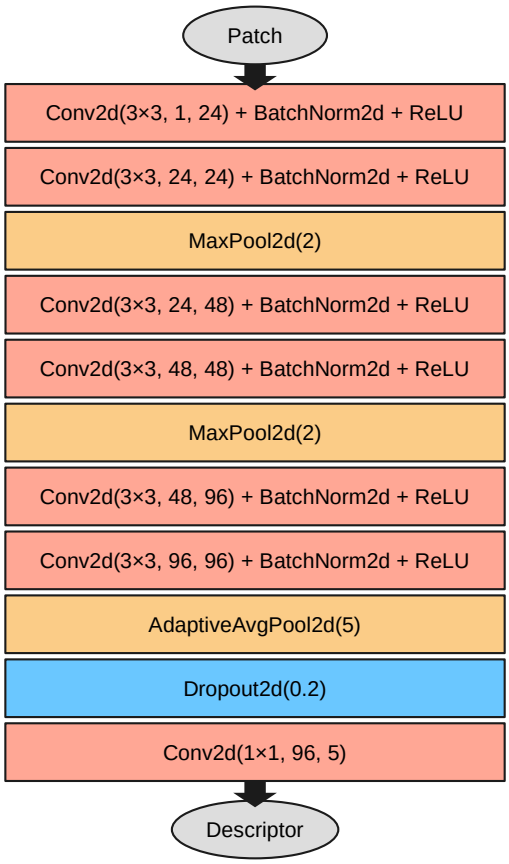


Figure 3.3 – One branch of our pseudo-siamese neural network architecture.

and

$$l^R(d_i^Q, d_i^R, d_k^R) := \max\left\{0, \delta + \|d_i^Q - d_i^R\|^2 - \|d_i^Q - d_k^R\|^2\right\} \tag{3.3}$$

Here, δ stands for some arbitrary margin that represents a minimal distance of non-colocated descriptors. We set $\delta = 1$. Letters Q and R respectively designate the query and the reference modalities and letter d stands for local features produced by the network. Finally, indices i, j and k identify geographic locations. We assume that $i \neq j$ and $i \neq k$.

The rest of the training parameters are set as follows. The learning rate is 0.01 with a decay of 0.95 every 4 epochs. We have considered to hard mine the negatives but, having observed no benefit, we have discarded this strategy. The negatives are sampled from the entire dataset and not from the minibatch. Each batch references 128 distinct geographic locations. Given that a geographic location is associated with a positive and a negative patch from the reference and from the query modalities, each batch is accordingly associated with $4 \times 128 = 512$ tuples of local features. The number of epochs is 40.



Figure 3.4 – Examples of pairs of raster patches.

3.2 Description models training and evaluation

In this section, we report the results of training the local descriptors described in the previous section, on the multimodal geospatial data presented in section 1.5.

3.2.1 Raster patch construction processes

We first describe how we produce a raster image patch given a piece of geospatial data and a desired patch geometry.

OpenStreetMap The patch construction process for OpenStreetMap is as follows. We first render a color raster patch having the desired geometry using libosmscout, a software library that provides the functionality of drawing OpenStreetMap data. We have customized the stylesheet that controls the drawing appearance and behaviour. We have notably disabled zoom-conditional drawing, we have removed labels and icons, we have made all the lines solid and we have removed the line terminal markers. We finally convert the patch to grayscale and normalize it to zero mean and unit variance.

Pléiades The patch construction process for Pléiades images is as follows. First, we convert the image to grayscale. Secondly, we resample it into a patch having the desired center, footprint diameter, rotation angle, horizontal and vertical flipping. Thirdly, we normalize the patch to zero mean and unit variance.

DOP The patch construction process for DOP is the same as Pléiades.

Bing The patch construction process for Bing is the same as Pléiades.

Miranda The patch construction process for Miranda images is as follows. First, we resample the image into a patch having the desired geometry. Secondly, we convert the patch from intensity to log-intensity. Thirdly, we normalize the patch to zero mean and unit variance.

KOMPSAT-5 The patch construction process for KOMPSAT-5 is the same as Miranda.

I-Master The patch construction process for I-Master is the same as Miranda.

3.2.2 Performance metrics

We now describe how to measure the discriminative power of a description model. Our metric is the false match rate for a true match rate of 95 %. This metric is commonly reported in the scientific literature on multimodal local descriptors [135, 151]. It is the probability that two geographically separate descriptors are closer, in descriptor space, than a threshold such that the probability of two geographically colocated descriptors being farther apart than this threshold is 5 %. We abbreviate the name of this metric as 95-score. Its mathematical definition is:

$$\mathbb{P}\left(\|d_i^Q - d_j^R\| < T \,|\, i \neq j\right) \tag{3.4}$$

with T such that:

$$\mathbb{P}\left(\|d_i^Q - d_j^R\| < T \,|\, i = j\right) = \frac{95}{100} \tag{3.5}$$

3.2.3 Performance of description models

For each combination of a piece of query geospatial data and a reference modality of the same multimodal stack, we train a description model using the other pieces of query geospatial data of the same modality combination. So doing, we can use the description models on locations unseen at training and perform cross-validation studies. We have chosen the raster patch sizes and the ground footprints depending on the combination of data modalities. We report them in table 3.1. Arbitrarily, for a given combination of data modalities, we use the same patch and footprint sizes for the reference modality and the query modality. The 95-scores of the trained description models are reported in table 3.2 for the first multimodal stack and in table 3.3 for the second multimodal stack. The ordered average scores for each modality combination are reported in table 3.4.

On the first stack, the results for Pléiades are generally better than the results for Miranda. The scores involving only optical images are generally better than the scores involving one optical image and one SAR image. The scores involving one optical image and one SAR image are generally better than the scores involving only SAR images. On the first stack, the SAR images appear to be intrinsically more difficult to match than the optical images. However, on the second stack, *I-Master+Bing* achieves the best score and *I-Master+OpenStreetMap* achieves the worst score. On the whole, the average scores reveal no obvious pattern. The best reference modalities are optical. KOMPSAT-5 is consistently in the worst half, which is coherent with the fact that its resolution is the worst. The worst combination is *I-Master+OpenStreetMap*. On this location, hedgerows and roads are often missing from OpenStreetMap, which may explain the poor scores. The Miranda images are easier to match with OpenStreetMap than with KOMPSAT-5. The band, viewpoint and resolution of Miranda and KOMPSAT-5 images differ to such an extent that it is easier to match Miranda with OpenStreetMap despite OpenStreetMap not being a physical measurement. For the combination *I-Master+Bing*, the piece of geospatial data *I-Master F* exhibits the worst scores. This piece of geospatial data represents a bridge over a river and exhibits strong multipath. For the combination *Miranda+KOMPSAT-5*, image *Miranda E* exhibits the poorest scores. This image contains well visible electric lines unseen at training. It also exhibits layover.

From these results, we first conclude that the modality gap does not inherently induce a higher difficulty in matching image patches. Secondly, we conclude that discrepancies, data incompletion, artefacts such as layover, poor resolution, high noise level may degrade the performance of patch matching.

3.3 Estimating a translation

3.3.1 Method

We wish to determine a mapping between the respective coordinate domains of two pieces of geospatial data such that the average similarity of the matched points is maximal. This writes as the following optimization problem:

$$\underset{T \in \mathcal{T}}{\text{minimize}} \int_{\Omega} s(x, f^Q, T(x), f^R) dx \tag{3.6}$$

$\mathcal{T}$ is a class of coordinate mappings, T is a coordinate mapping, Ω is the coordinate domain of the query geospatial data, f^Q and f^R are respectively the query geospatial data and the reference geospatial data. The function $s(x, f^Q, y, f^R)$ measures how dissimilar point x on f^Q and point y on f^R are. It may for instance be the L_2 norm. We decompose s as:

$$s(x, f, y, g) = s(d^Q(x, f^Q), d^R(y, f^R)) \tag{3.7}$$

Here, $d^Q(x, f^Q)$ and $d^R(y, f^R)$ are local descriptors respectively extracted from f^Q at x and from f^R at y. We suppose that the descriptors satisfy the following property for any transform T in $\mathcal{T}$:

$$d^R(T(x), f^R) = d^R(x, f^R \circ T) \tag{3.8}$$

If we approximate the integral of equation 3.6 with a finite sum, the problem becomes:

$$\underset{T \in \mathcal{T}}{\text{minimize}} \sum_{x \in \mathcal{X}} s(d^Q(x, f^Q), d^R(T(x), f^R)) \tag{3.9}$$

$\mathcal{X}$ denotes a set of integration points sampled over the coordinate domain of the query (i.e. sampled in Ω). The cost function can be evaluated for any coordinate mapping without evaluating the local descriptors $d^Q(x, f^Q)$ of the query geospatial data more than once. Similarly, to save execution time, we wish that evaluating the cost function many times requires only a single evaluation of the reference local descriptors $d^R(T(x), f^R)$. For this purpose, we precompute descriptors and interpolate them with nearest neighbors interpolation. In other words:

$$d^R(T(x), f^R) \approx d^R(\text{nn}(T(x), \mathcal{Y}), f^R) \tag{3.10}$$

Here, $\text{nn}(y, \mathcal{Y})$ is the nearest neighbor of y in $\mathcal{Y}$ and $\mathcal{Y}$ contains samples of points in the coordinate domain of the reference geospatial data. In an implementation, nearest neighbor searches can be precomputed and/or performed efficiently using adequate data structures. The problem becomes:

$$\underset{T \in \mathcal{T}}{\text{minimize}} \sum_{x \in \mathcal{X}} s(d^Q(x, f^Q), d^R(\text{nn}(T(x), \mathcal{Y}), f^R)) \tag{3.11}$$

We define $\mathcal{F}$ as points in the query coordinate domain together with their local descriptors:

$$\mathcal{F} = \left\{(x, d^Q(x, f^Q)); x \in \mathcal{X}\right\} \tag{3.12}$$

We define $\mathcal{G}$ as points in the reference coordinate domain together with their local descriptors:

$$\mathcal{G} = \left\{(y, d^R(y, f^R)); y \in \mathcal{Y}\right\} \tag{3.13}$$

Given $\mathcal{F}$ and $\mathcal{G}$, the cost function that we have introduced can be evaluated without extracting new descriptors.

The registration method that we have presented relies on sampling points in the coordinate domain of each piece of geospatial data and in extracting a local descriptor at each point. In the coordinate domain of the reference geospatial data, the points are sampled by uniform stratified sampling. In other words, the coordinate domain associated with the reference is partitioned into a grid and points are sampled within each cell with a uniform distribution. The cell size must be small enough so that the nearest neighbor approximation holds. We set it to one twentieth of the footprint diameter of the local descriptors. In the coordinate domain of the query geospatial data, the points are sampled in a grid. The cell size is chosen in light of these considerations:

— It should preferably be small enough so that the query geospatial data are entirely covered with footprints of local descriptors,
— The execution time of evaluating the cost function is proportional to the number of points sampled on the query geospatial data.

We may accordingly set the cell size of the query to half the footprint diameter of the local descriptors.

Our cost function may have many local minima so we simply optimize it with a grid search. We define a grid in the parameter space of $\mathcal{T}$ and sample the cost at the center of each cell. The cost function is not differentiable but could be made so using another interpolation method. However, since there may be many attraction pools, we would require a grid search anyway. As a class of coordinates transforms, we only consider translations here. Incidently, the parameter space of the coordinate mapping is a plane and we can vizualise the values of the cost function in the form of a two-dimensional image which we call a correlation map.

Interpolating reference descriptors saves execution time because the reference descriptors are used several times. One reference descriptor is used for several evaluations of the cost function. The reference descriptors and the nearest neighbor searches can be computed beforehand, which saves inference execution time at the detriment of space. Once they are computed, they can be used for any number of inferences. If the reference descriptors are precomputed, then interpolating them reduces the amount of space required. The inference algorithm is highly parallelizable and without conditional branches. The precomputation of the descriptors and nearest neighbor searches is also highly parallelizable.

3.3.2 Relation to image retrieval

Image retrieval is the task of searching in a database an image that is most similar to a query image. The similarity between a query image and a database image is measured using global image descriptors. The method described in the previous section can be thought of as image retrieval. In our method, the global descriptor associated with an image is the concatenation of all the local descriptors. However, in typical image retrieval methods, the local descriptors aggregation is more complex than concatenation and achieves smaller global descriptors. Well-known aggregation procedures are Bag of Visual Words (BOVW) [203, 251] and Vector of Locally Aggregated Descriptors (VLAD) [105, 98].

To the best of our knowledge, no image retrieval system has been experimentally demonstrated to work on such challenging multimodal data as ours. There is little work on multimodal remote sensing image retrieval [265, 244], and remote sensing image retrieval systems are generally tested on databases of a few tens of thousands of images. In our method, the distance between two global descriptors is the average point-to-point dissimilarity of the two images, as measured by local descriptors. This property might be lost if the local descriptors were aggregated by a complex process. The matching performance would probably be reduced, and the behavior of the algorithm would be less intuitive. Even if the local descriptors were aggregated by a complex process, we could still not simultaneously improve execution time and memory use, since our local descriptors are shared among global descriptors so that a map of global descriptors and a map of local descriptors occupy the same volume of memory (provided that the sampling step is the same and that the global and local descriptors have the same dimension). Conversely, our method would perform poorly as an image retrieval method since we generally expect image retrieval methods to not care about the location of the searched content within the image.

3.3.3 Experiments

We now report alignment experiments based on the method that we have presented and we analyse their results.

Protocol

The overall registration procedure is represented in figures 3.5 and 3.6. We align pieces of query geospatial data onto pieces of reference geospatial data. For each alignment, the size of the query is $(800\text{ m})^2$, that is to say 0.64 km^2, and the size of the reference is $(15\text{ km})^2$, that is to say 225 km^2. Unless the reference modality is KOMPSAT-5, we choose the reference so that the location of the query within the reference is random and uniformly distributed. If the reference modality is KOMPSAT-5, the reference is chosen so that the query is located at the center of the reference. This is because the area covered by our KOMPSAT-5 images is too small to select the reference randomly as with the other

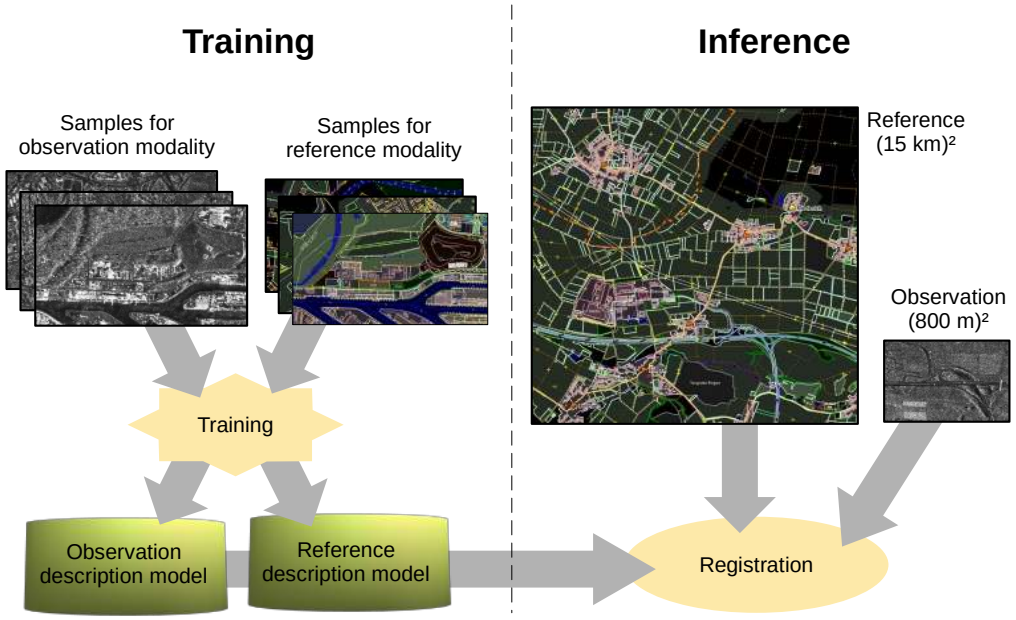


Figure 3.5 – Image registration using a learned multimodal similarity metric.

modalities. We reuse the description models whose training was presented in section 3.2. The description model used for an alignment has not been trained on the geospatial data from which the query geospatial data are extracted. In other words, a description model has not seen at training the geographic location used for inference. We precompute the reference descriptors. The reference stratified sampling cell size is 10 m. We compute the nearest neighbors during the inference instead of precomputing them.

We measure the quality of a superposition produced by our algorithm as follows. For each alignment experiment, we compute the magnitude of the average offset between the estimated and the ground-truth translations. For each combination of data modalities, we report the cumulative distribution of these magnitudes. We also report, for each combination of data modalities, the number of experiments for which the magnitude of the mean error is lower than 20 m. We call success rate the probability that an alignment experiment achieves a precision better than 20 m (we choose this threshold arbitrarily).

Results

We report in figure 3.8 the cumulative distributions of the magnitudes of the average errors. We report in table 3.5 the number of experiments for which the magnitude of the average error is less than 20 m. We report in figure 3.9 examples of alignments. An alignment takes between 1.5 and 2 seconds on an *Intel Core i7-12700* CPU with parallel evaluation of the cost function.

We also report in table 3.5 the success rates for mutual information (MI) and normal-

Query modality	Reference modality	Patch size (px)	Footprint size (m)
Miranda	OpenStreetMap	60	150
Miranda	KOMPSAT-5	60	200
Miranda	DOP	40	150
Pléiades	OpenStreetMap	80	200
Pléiades	KOMPSAT-5	60	200
Pléiades	DOP	40	150
I-Master	OpenStreetMap	40	200
I-Master	Bing	40	200

Table 3.1 – Patch size and footprint size for each modality combination.

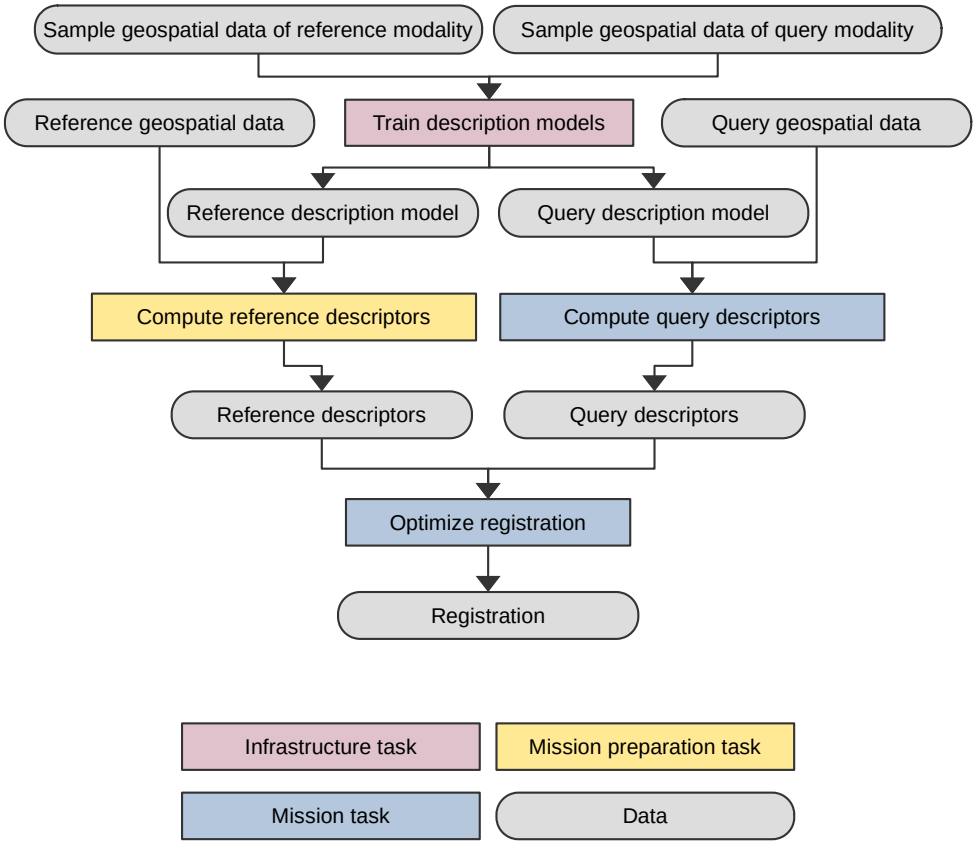


Figure 3.6 – Translation estimation procedure.

Excluded query geospatial product	**Reference modality**	**95-score**
Miranda A	OpenStreetMap	0.154
Miranda B	OpenStreetMap	0.028
Miranda C	OpenStreetMap	0.277
Miranda D	OpenStreetMap	0.327
Miranda E	OpenStreetMap	0.278
Miranda A	KOMPSAT-5	0.348
Miranda B	KOMPSAT-5	0.045
Miranda C	KOMPSAT-5	0.314
Miranda D	KOMPSAT-5	0.203
Miranda E	KOMPSAT-5	0.466
Miranda A	DOP	0.252
Miranda B	DOP	0.009
Miranda C	DOP	0.223
Miranda D	DOP	0.109
Miranda E	DOP	0.290
Pléiades A	OpenStreetMap	0.110
Pléiades B	OpenStreetMap	0.274
Pléiades C	OpenStreetMap	0.195
Pléiades D	OpenStreetMap	0.133
Pléiades E	OpenStreetMap	0.082
Pléiades A	KOMPSAT-5	0.335
Pléiades B	KOMPSAT-5	0.242
Pléiades C	KOMPSAT-5	0.220
Pléiades D	KOMPSAT-5	0.245
Pléiades E	KOMPSAT-5	0.219
Pléiades A	DOP	0.227
Pléiades B	DOP	0.285
Pléiades C	DOP	0.083
Pléiades D	DOP	0.015
Pléiades E	DOP	0.069

Table 3.2 – Descriptors performance metrics for the first stack.

Excluded query geospatial product	Reference modality	95-score
I-Master A	OpenStreetMap	0.069
I-Master B	OpenStreetMap	0.179
I-Master C	OpenStreetMap	0.239
I-Master D	OpenStreetMap	0.389
I-Master E	OpenStreetMap	0.675
I-Master F	OpenStreetMap	0.505
I-Master G	OpenStreetMap	0.218
I-Master A	Bing	0.010
I-Master B	Bing	0.089
I-Master C	Bing	0.042
I-Master D	Bing	0.156
I-Master E	Bing	0.106
I-Master F	Bing	0.450
I-Master G	Bing	0.016

Table 3.3 – Descriptors performance metrics for the second stack.

Query modality	Reference modality	Average score
I-Master	Bing	0.124
Pléiades	DOP	0.136
Pléiades	OpenStreetMap	0.159
Miranda	DOP	0.177
Miranda	OpenStreetMap	0.213
Pléiades	KOMPSAT-5	0.252
Miranda	KOMPSAT-5	0.275
I-Master	OpenStreetMap	0.368

Table 3.4 – Ordered mean scores for each modality combination.

ized cross-correlation (NCC). We have computed the success rates associated with these two traditional methods by producing 10 meters resolution rasters (for the reference and the query) and performing pattern matching using these similarity metrics. MI and NCC fail to obtain satisfying alignment success rates. MI offers much higher success rates than NCC when the reference is OpenStreetMap. For the other references, these two methods offer relatively similar success rates. Using the traditional methods, the best scores are achieved on the combination Pléiades-DOP, whose modality gap is the smallest.

The alignment algorithm achieves high robustness despite the modality gap and the fact that the area of the reference is more than 350 times larger than the area of the query. The combination of modalities that achieves the best median precision is Pléiades/DOP, which is coherent with the small gap between these two modalities. The worst combination of modalities in terms of the median precision is Miranda/KOMPSAT-5. This may derive from the comparatively poor resolution and dynamic range of our KOMPSAT-5 images. We can order the modality combinations as follows in terms of median precision:

— Pléiades + DOP < Miranda + DOP < Miranda + KOMPSAT-5
— Pléiades + DOP < Pléiades + KOMPSAT-5 < Miranda + KOMPSAT-5

Our optical images appear to be systematically more adequate for alignment than our SAR images. When the query is Pléiades or Miranda, in terms of precision, using OpenStreetMap as a reference is nearly as good as using an optical image and is significantly better than using our SAR images. When the query is I-Master, using OpenStreetMap as reference is worse than using our optical images. We have observed that our OpenStreetMap archive of the United Kingdom lacks many hedges, dirt roads and internal details of industrial areas. We assume that, if our OpenStreetMap archive contained these missing objects, the performance obtained using OpenStreetMap data would be similar to the performance obtained using the optical reference. A sample correlation map is represented in figure 3.7. The correlation peak is clearly delimited.

Examining the unsuccessful alignment experiments, we have identified the following failure modes: objects unseen at training, time inconsistency and poor focusing. Poor focusing may notably be caused by ground relief. One Pléiades query whose alignment failed contains fields covered by farming tarpaulins unseen at training. One Miranda query whose alignment failed contains electric lines. They were unseen at training and exhibited layover. One Miranda query whose alignment failed contains hills and is poorly focused. One I-Master query whose alignment failed contains a large open air mine which is not well reported in OpenStreetMap. One I-Master query whose alignment failed contains an industrial area with tanks which are not well reported in OpenStreetMap and exhibits multi-path. One I-Master query whose alignment failed contains only a bridge over the water with strong multipath echos.

In summary, estimating a translation between pieces of geospatial data of different modalities using local descriptors is very robust and requires one or two seconds in our setting. Poor image quality and discrepancies may degrade the probability of a successful matching. In future works, we may consider improving the computational efficiency either using learned binary descriptors [250] or by using a Fast Fourier Transform (FFT) based

descriptors correlation.

Query	Reference	Num. exp.	Success rate	... MI	... NCC
Pléiades	OpenStreetMap	74	100 %	33.0 %	1.1 %
Pléiades	DOP	74	100 %	63.5 %	45.9 %
Pléiades	KOMPSAT-5	94	94 %	3.2 %	1.1 %
Miranda	OpenStreetMap	46	98 %	11.1 %	1.9 %
Miranda	DOP	45	100 %	4.4 %	6.7 %
Miranda	KOMPSAT-5	48	78 %	14.6 %	14.6 %
I-Master	OpenStreetMap	180	93 %	9.9 %	0.6 %
I-Master	Bing	180	93 %	6.1 %	0.6 %

Table 3.5 – Number of alignment experiments and success rate for each combination of data modalities. Comparison with Mutual Information (MI) and Normalized Cross-Correlation (NCC).

3.4 Estimating the full kinematic state

We now describe how to estimate the full kinematic state using a multimodal similarity metric. The working principle of the estimation system is illustrated in figures 3.10 and 3.11. We determine a most likely state among a predefined set of virtual states. The likelihood of a virtual state is the similarity between the reference description of the ground and a virtual description of the ground predicted on the basis of this virtual state. The predicted description of the ground is a geocoded SAR image constructed on the assumption of this virtual state.

In order to implement this system, we need to address several issues.

— We should define how to measure the similarity between a SAR image and the description of the ground.

— We should define how to perform an estimation with an amount of computations that is compatible with the mission requirements. The ideal system sketched in figure 3.11 requires notably to form a geocoded SAR image for each test state. This is impractical and actually wastes computational resources.

We address these two issues in the next paragraphs.

3.4.1 Multimodal correlation

As seen in sections 3.1 and 3.2, we measure the similarity between a predicted SAR image and reference geospatial data using learned multimodal local descriptors.

— The similarity of the two pieces of geospatial data is their average local similarity over the intersection of their geographic domains.

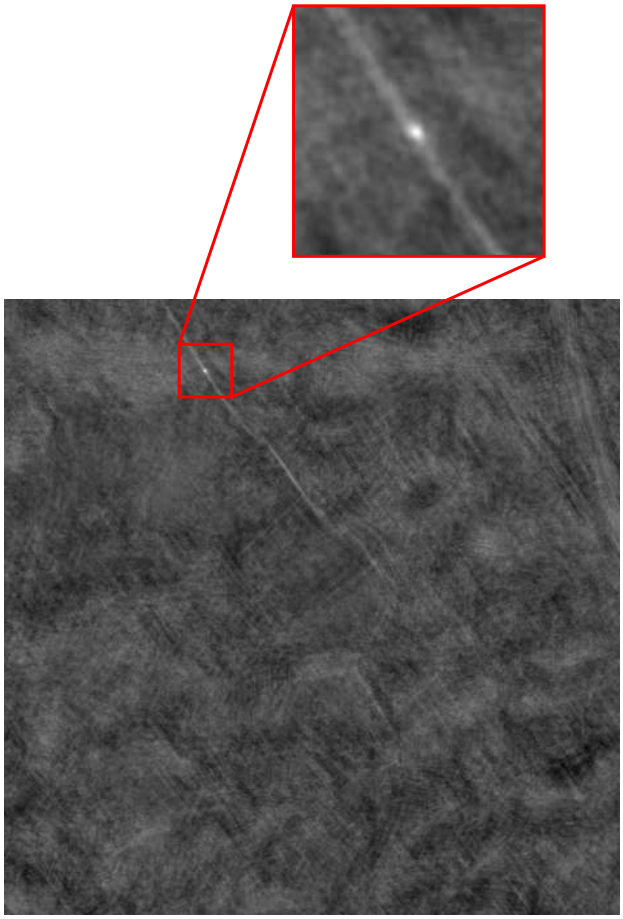


Figure 3.7 – A correlation map covering (15 km)^2 with a resolution of 10 m.

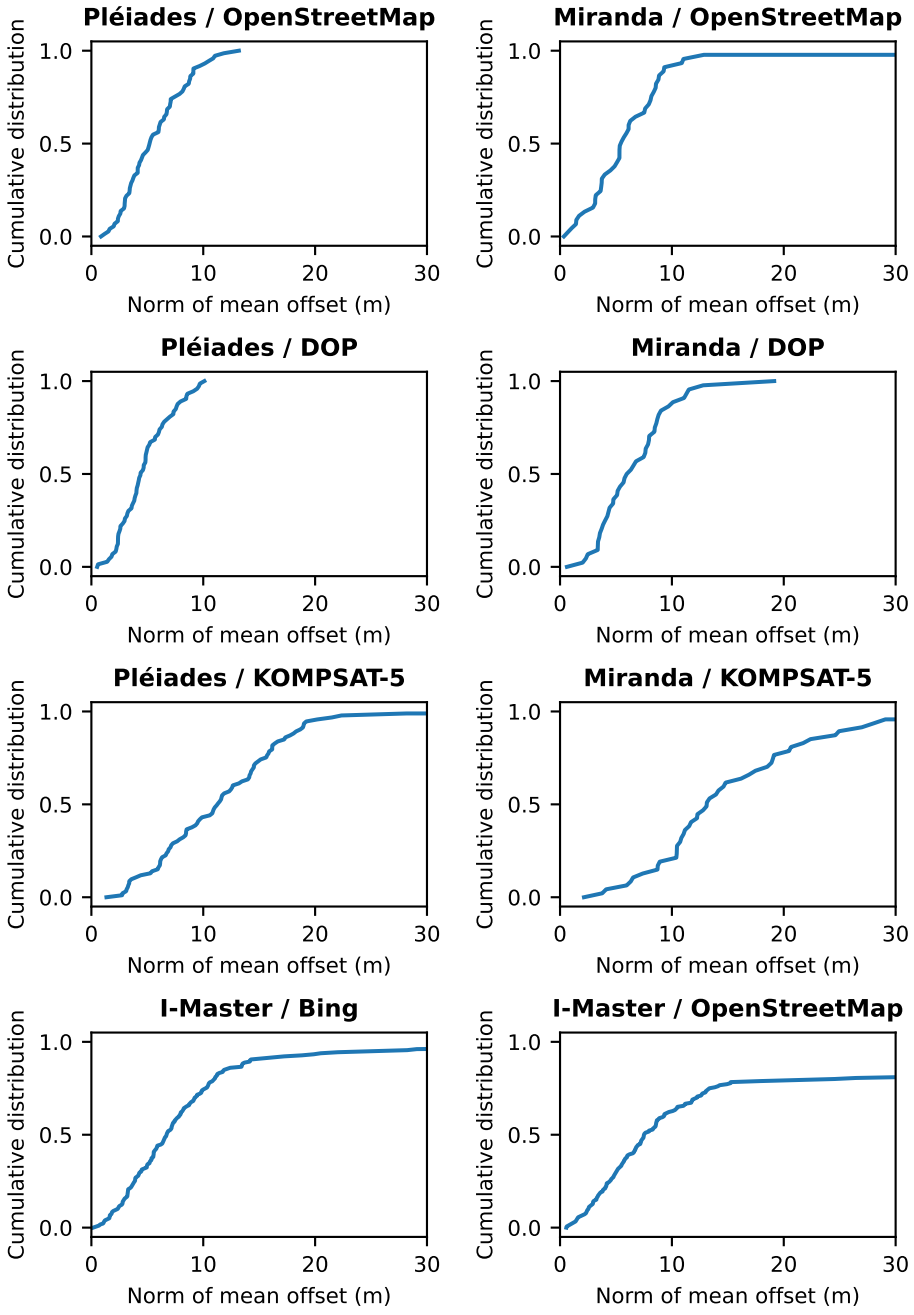


Figure 3.8 – Translation estimation precision.

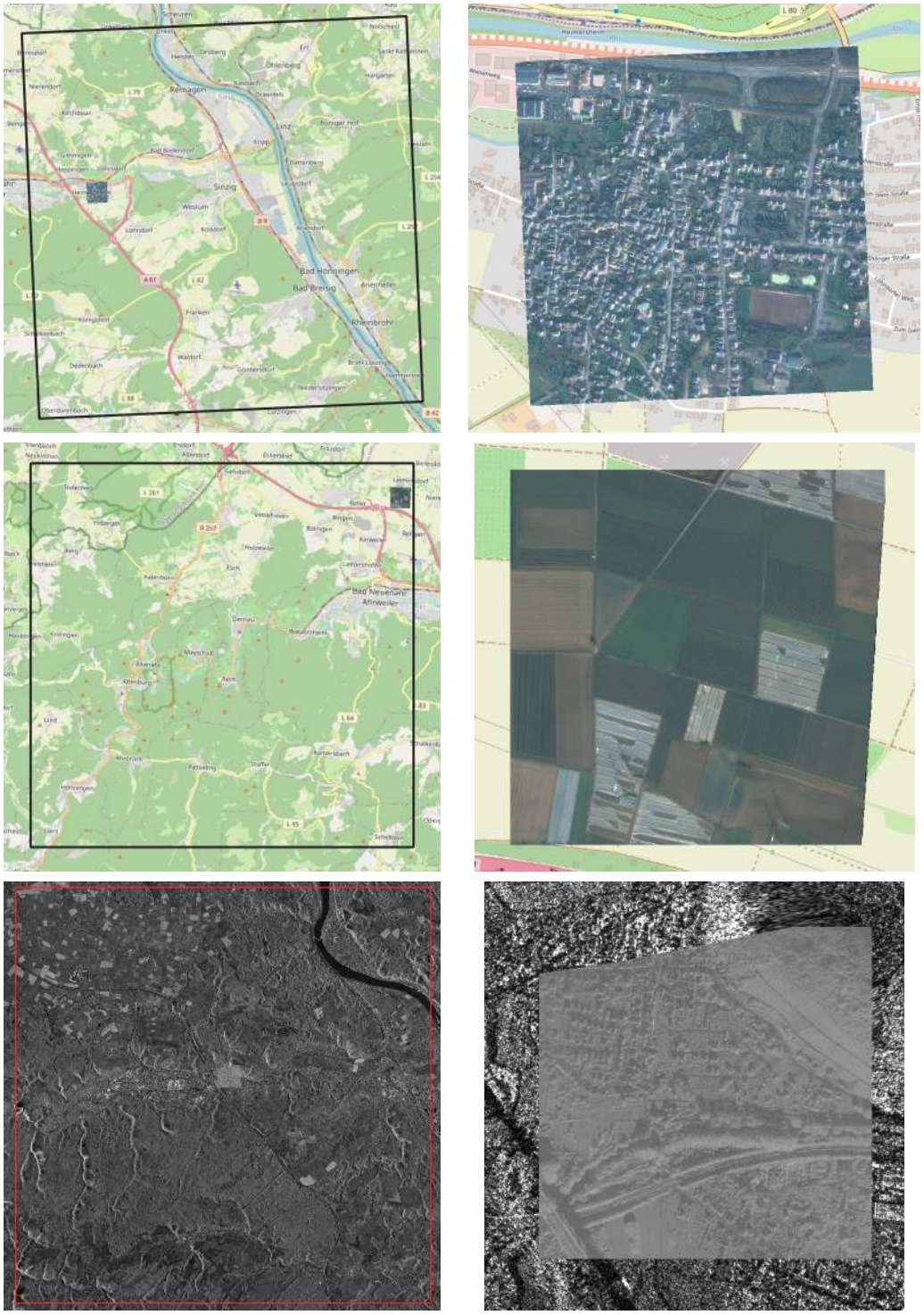


Figure 3.9 – Alignment examples. Each row represents a single alignment experiment. The first and second rows are Pléiades/OpenStreetMap combinations. The third row is a Miranda/KOMPSAT-5 combination. The images of the left column are zoomed in on the reference. The images of the right column are zoomed in on the observation.

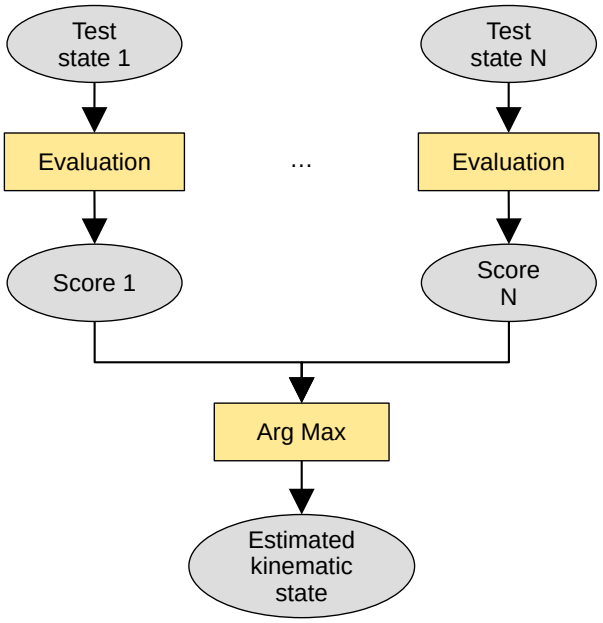


Figure 3.10 – Working principle of our proposed kinematic state estimation system.

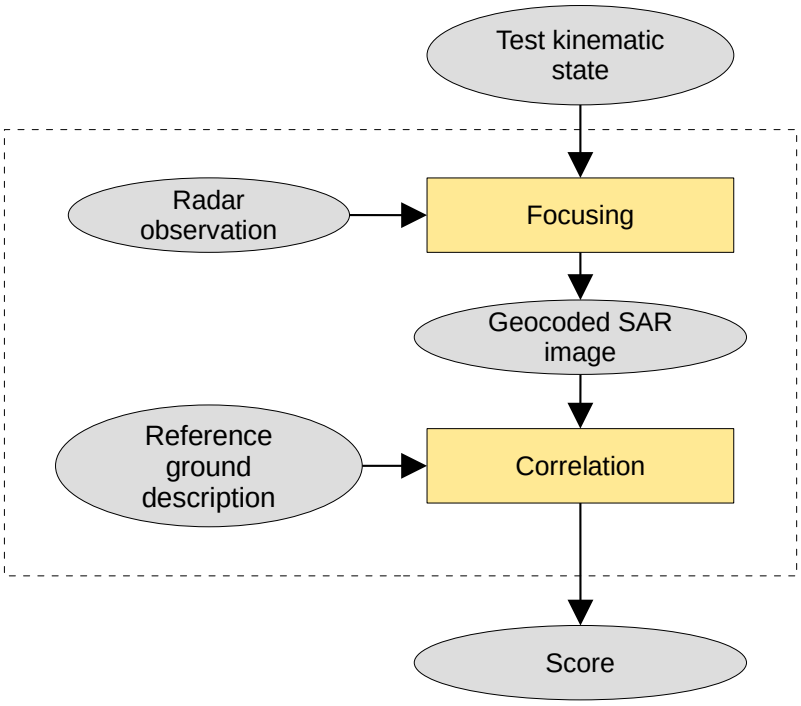


Figure 3.11 – Evaluation of a virtual kinematic state.

— The local similarity of the two pieces of geospatial data at some geographic point is the similarity of representations of the restrictions of the geospatial data to the vicinity of this point. We call such a representation a descriptor. It encodes the appearance of the geospatial data in the vicinity of a geographic point. This is illustrated in figure 3.12. In the metric space of descriptors, the descriptors associated with the same geographic location are close to one another.
— The similarity of two descriptors is measured using a simple mathematical metric such as the L2 norm.

In other words, the procedure by which we measure the similarity of two pieces of geospatial data is as follows. We sample points in the intersection of the two geographic domains. We extract a descriptor at each point for each modality (overall, there is one descriptor for each combination of a piece of geospatial data and a sample geographic point). We compute the average similarity of the paired descriptors.

3.4.2 Kinematic parameters

We identify a kinematic state with a position $\boldsymbol{o}$ and a velocity vector $\boldsymbol{v}$ in an East North Up (ENU) local tangent frame $(\boldsymbol{e_x}, \boldsymbol{e_y}, \boldsymbol{e_z})$. We parameterize the state as follows.

a is the origin ENU easting.

b is the origin ENU northing.

h is the origin ENU height above ground.

s is the norm of the velocity vector.

θ is the angle, measured in the horizontal plane, between the geographic axes and the projection of the velocity vector onto the horizontal plane. $\theta \in]-\pi, \pi]$.

ϕ is the angle, measured in the vectical plane that contains the velocity vector, between the velocity vector and its projection onto the horizontal plane. $\phi \in]-\pi/2, \pi/2[$.

In terms of these parameters, the kinematic state is:

$$\begin{cases} \boldsymbol{o}^T \boldsymbol{e_x} = a \\ \boldsymbol{o}^T \boldsymbol{e_y} = b \\ \boldsymbol{o}^T \boldsymbol{e_z} = h \\ \boldsymbol{v}^T \boldsymbol{e_x} = s\cos(\phi)\cos(\theta) \\ \boldsymbol{v}^T \boldsymbol{e_y} = s\cos(\phi)\sin(\theta) \\ \boldsymbol{v}^T \boldsymbol{e_z} = s\sin(\phi) \end{cases} \tag{3.14}$$

Conversely:

$$\begin{cases} a = \boldsymbol{o}^T \boldsymbol{e_x} \\ b = \boldsymbol{o}^T \boldsymbol{e_y} \\ h = \boldsymbol{o}^T \boldsymbol{e_z} \\ \theta = \operatorname{arctan2}\left(\boldsymbol{v}^T \boldsymbol{e_y}, \boldsymbol{v}^T \boldsymbol{e_x}\right) \\ \phi = \arcsin\left(\frac{\boldsymbol{v}^T \boldsymbol{e_z}}{\|\boldsymbol{v}\|}\right) \\ s = \|\boldsymbol{v}\| \end{cases} \tag{3.15}$$

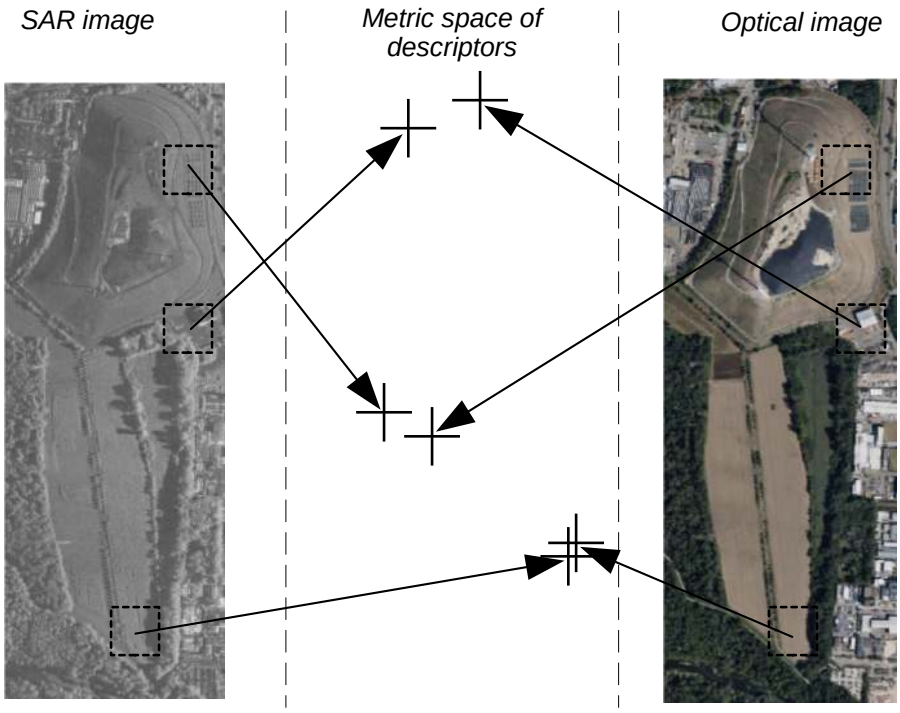


Figure 3.12 – Extracting descriptors from two pieces of geospatial data.

Here, $\boldsymbol{e_x}$, $\boldsymbol{e_y}$ and $\boldsymbol{e_z}$ are unit length vectors respectively oriented along the easting, northing and up directions. As well, θ is the angle between the ground x-axis and the projection of the velocity vector onto the ground.

3.4.3 Computational affordability

We now describe ways to make the proposed kinematic state estimation system computationally affordable.

Computing in advance

By computing in advance, we mean to perform a computation before the mission, to save the computation products onto the device and to use them during the mission.

1. We may extract in advance the reference local descriptors.

Factorizing

By factorizing computations, we mean to perform a redundant computation only once and reuse the computation products several times.

1. We may perform the radar echos pulse compression once.
2. Virtual kinematic states with the same speed s may share the same focusing operation. This is possible by constructing the SAR image in TCA-CD coordinates or in IDF-ID coordinates. We can consequently reduce the number of focusing operations to the number of unique parameters s among all the virtual kinematic states.
3. Virtual kinematic states with the same speed s, altitude h and elevation ϕ may share the same SAR descriptor extraction. We can consequently reduce the number of SAR descriptor extractions to the number of unique (s, h, ϕ) combinations of parameters among all the virtual kinematic states.

The combination of factorizations 2 and 3 is illustrated by figure 3.13.

Factorizing and approximating

By approximating, we mean to replace a computation with another one that produces approximately the same results with a lower workload.

1. We may approximate the reference descriptors with interpolation from a fixed set of reference descriptors, for instance using nearest neighbors. This enables sharing reference descriptors among different virtual kinematic states and consequently reducing the volume of data.

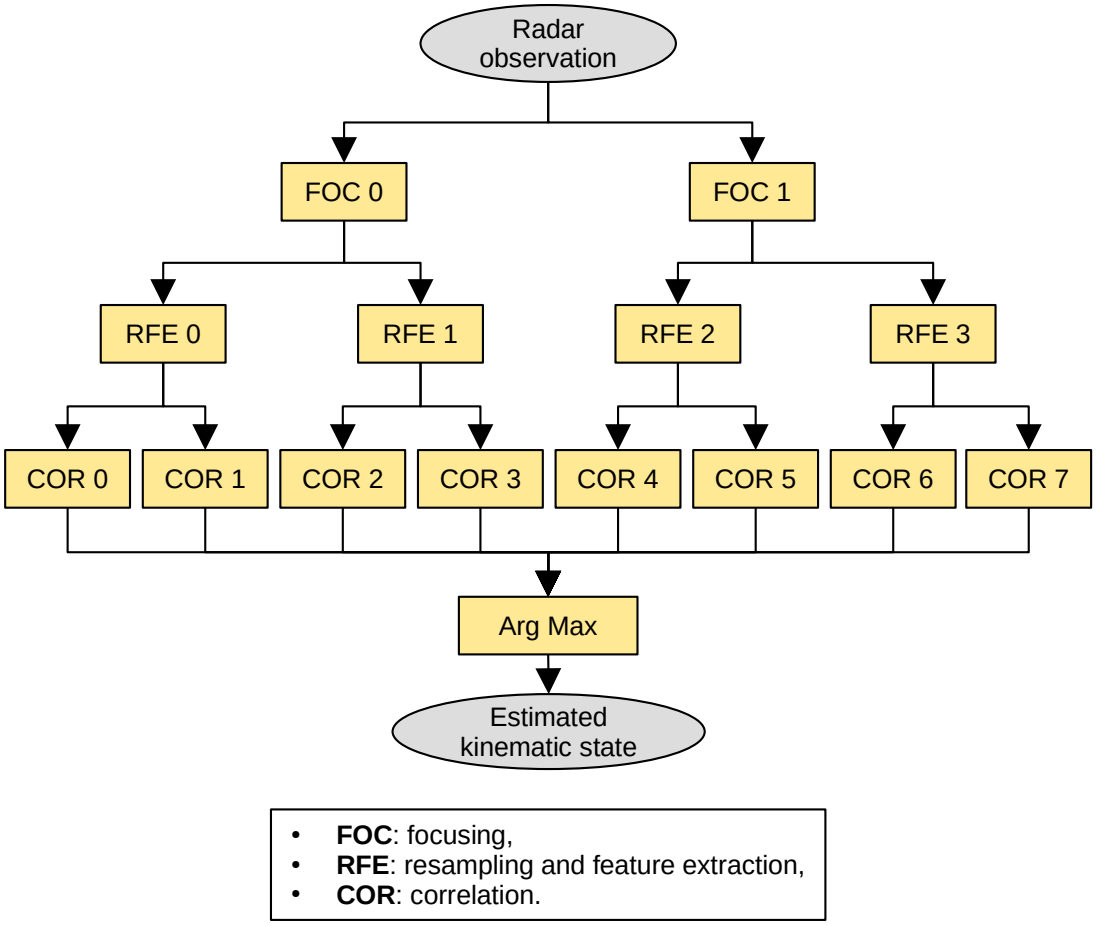


Figure 3.13 – Illustrative factorized workflow.

Prunning

By prunning a computation, we mean to skip it altogether.

1. We may deduce from another source of information that some virtual kinematic states are not possible. We may then skip computing their likelihood. Altimetry and prior assumptions (e.g. horizontal flight, known range of orientation angle) may provide such information. The information may appear before or during the mission.
2. In a multiresolution approach, we may process a batch of test kinematic states and recursively process another batch based on the current best virtual kinematic state.

Parallelizing

By parallelizing some computations, we mean to perform then simultaneously, provided that the hardware architecture allows it.

1. We may compute local descriptors in parallel.
2. We may evaluate virtual kinematic states in parallel.

3.4.4 Specifications of an operational system

We now bring together the ideas that we have presented and specify a state estimation system. It aims to solve the following optimization problem:

$$\underset{(a,b,h,\theta,\phi,s)}{\text{minimize}} \frac{\int_{\Omega(h,\phi,s)} \|f^{\text{SAR}}(x,h,\phi,s) - f^{\text{ref}}(\gamma(x,a,b,\theta),\theta)\|^2 dx}{\int_{\Omega(h,\phi,s)} dx} \tag{3.16}$$

Our notations are:

Ω designates the part of the pseudo-ground domain that contains the SAR image. It depends on h, s and ϕ. The pseudo-ground coordinates are identical to the ground coordinates up to a direct two-dimensional rigid transform. The first axis of the pseudo-ground coordinate system is the projection of the device velocity onto the ground. The origin of the pseudo-ground coordinate system is the projection of the device position onto the ground. The pseudo-ground coordinates are identical to the ground coordinates when the horizontal component of the velocity is eastward and the projection of the device position onto the ground is the ENU origin.

x designates pseudo-ground coordinates.

γ transforms pseudo-ground coordinates to ground coordinates. It is a plane rigid transform that depends on a, b and θ.

$f^{\mathbf{SAR}}$ produces a SAR image in pseudo-ground coordinates and extracts a multimodal local descriptor at the given pseudo-ground coordinates.

$f^{\mathbf{ref}}$ produces a multimodal local descriptor from the reference geospatial data at the given ground coordinates and orientation.

The workflow that we propose is presented in figure 3.14. It is composed of three parts.

1. We train the two description models. This requires overlapping samples of geospatial data of the reference and query modality. This stage is performed once for a given combination of data modalities.
2. We compile the navigation data. These are all the data that the device shall rely on to estimate its kinematic state. This stage is performed once for a given mission, prior to the mission. Its input are the reference geospatial data describing the region of interest, a trained description model and the definition of the virtual kinematic states. We proceed with the various computations, notably of the reference descriptors.
3. We finally proceed with the actual kinematic state estimation given a radar observation and the navigation data. This stage is performed during the mission as many times as required and allowed by external constraints.

We have detailed the first stage (i.e. training the description models) in section 3.2. We now describe the navigation data, the second and third stages in the next three sections.

3.4.5 The navigation data

The navigation data are composed of:

— A set of samples for each axis of the state space (i.e. northing, easting, altitude, orientation, elevation and speed). The virtual states to be evaluated are the cartesian product of these six arrays.
— A list of reference descriptors extracted from the reference geospatial data.
— For each combination of a speed, an altitude and an elevation angle, a list of pseudo-ground coordinates (which we shall call observation description points) at which to extract local descriptors from the SAR observation.
— The index of the reference descriptor for each combination of an observation description point, a northing, an easting and an orientation.
— The observation description model. For instance, these may be the architecture definition and the weights of a neural network.

Typically, the most voluminous parts of the navigation data are the array of indices of descriptors and the array of descriptors. The size of the first array is proportional to the total number of observation description points, to the number of northings, to the number of eastings and to the number of orientations. The size of the second array is proportional to the ground area that may be imaged and to the number of orientations because the descriptors that we use are not rotation-invariant.

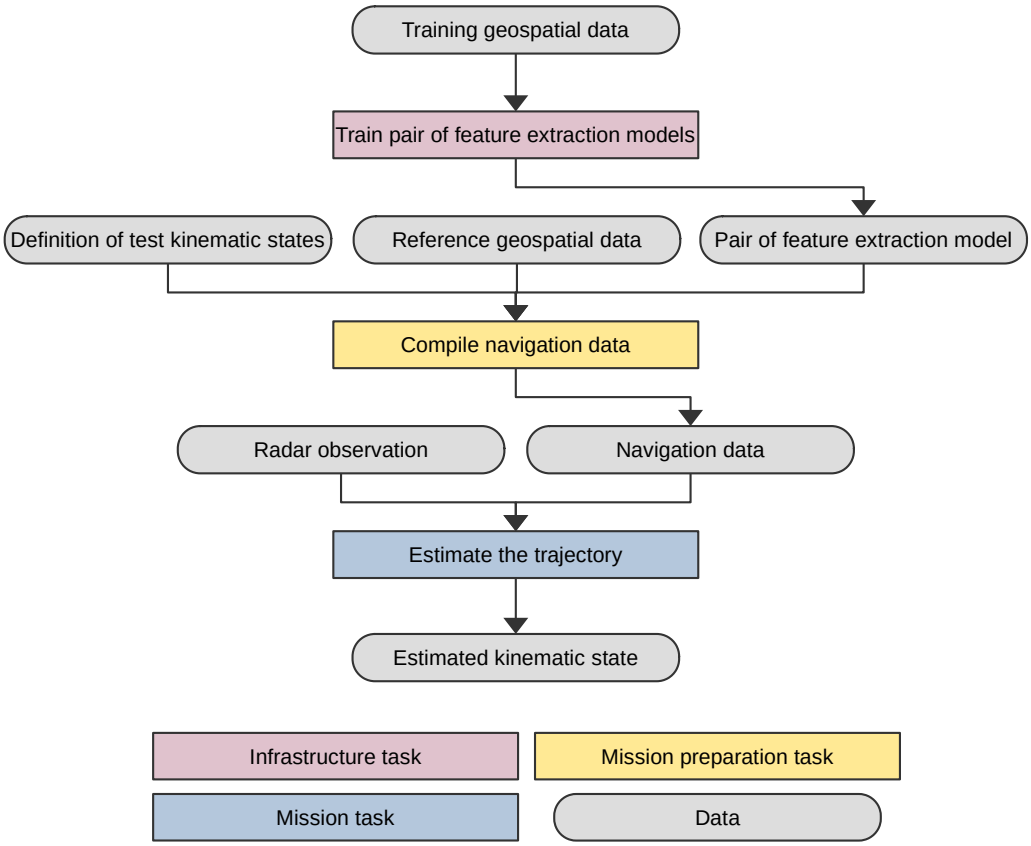


Figure 3.14 – Workflow of the kinematic state estimation system.

3.4.6 Compiling the navigation data

Producing the navigation data requires the following input:

— The definition of the set of virtual kinematic states. We define a center state and, for each axis of the state space (i.e. northing, easting, altitude, orientation, elevation and speed), the length of an interval and a discretization step. The set of virtual kinematic states is the cartesian product of the samples of each axis. The resolution of the state estimation for a given axis cannot be finer than the discretization step of this axis.
— The reference description model. We shall use it to compile the navigation data.
— The observation description model. We shall copy it to the navigation data.
— The reference geospatial data. We shall use these data to compute the reference local descriptors. The reference geospatial data and the reference description model have the same modality.
— Radar and platform parameters. This includes the radar frequency. This includes the instant of times at which the illumination shall start and stop, in the timeframe of the observation SAR image. This includes the squint angle of the antenna. This includes whether the antenna looks to the left or to the right of the trajectory. This includes the illumination beam roll angles, in the plane orthogonal to the device velocity, between the projected vertical and the limits of the projected beam, as illustrated in figure 3.15 in the case of an horizontal velocity.
— The observation descriptors sampling distance. This defines the size of a cell of a grid in the pseudo-ground plane. The observation descriptors are extracted from the observation at the center of each cell.
— The reference descriptors sampling distance. This defines the size of a cell on the ground. The reference descriptors are sampled uniformly within each cell (so that the overall sampling strategy is uniform stratified sampling).

The navigation data compilation is detailed in algorithm 2. Notably, in order to share the reference descriptors among several virtual state evaluations, we rely on some data structure. Let us call this data structure a layer. There is one layer per test orientation. A layer is composed of a set of ground points that are indexed by a spatial data structure for efficient nearest neighbors searches. Each of these ground points may or may not be associated with a reference to a descriptor. In the course of constructing the navigation data, when we need to extract a descriptor at some arbitrary ground point A, we find B the nearest neighbor of A among the ground points of the layer. If B is not already associated with a descriptor, we compute a local descriptor at B. Under this scheme, in practice, we have observed that each descriptor gets referenced a few hundred times in average. Sharing the reference local descriptors achieves a high compression rate. A layer is also composed of a bounding box of the ground coordinates of the points that are illuminated under at least one kinematic state.

Input: A collection of virtual kinematic states $((s_i, h_i, \phi_i, \theta_i, a_i, b_i))_{1 \leq i \leq N}$.
Input: A reference description model.
Output: The navigation data.
for *each orientation* θ **do**
    Initialize $layer[\theta].bbox$ to an empty bounding box in the ground coordinates domain ;
end
for *each combination of speed, altitude and elevation* (s, h, ϕ) **do**
    Determine the illuminated area in the pseudo-ground coordinates domain ;
    Determine $expoints[s, h, \phi]$ as the pseudo-ground points at which to extract observation descriptors ;
    for *each combination of orientation, northing and easting* (θ, b, a) **do**
        Extend $layer[\theta].bbox$ so that it includes the illuminated ground area under the hypothesis $(a, b, h, \phi, \theta, s)$;
    end
end
for *each orientation* θ **do**
    Construct point cloud $layer[\theta].points$ by uniform stratified sampling within $layer[\theta].bbox$;
    Mark each point as not associated with any descriptor ;
    Construct a kd-tree $layer[\theta].index$ for the points $layer[\theta].points$;
end
for *each hypothesis* $(s, h, \phi, \theta, b, a)$ **do**
    for *each pseudo-ground point* A *in* $expoints[s, h, \phi]$ **do**
        Compute ground point B from pseudo-ground point A under hypothesis $(s, h, \phi, \theta, b, a)$;
        Using $layer[\theta].index$, determine C the closest neighbor of B among $layer[\theta].points$;
        if C *is not associated with any descriptor* **then**
            Compute the reference descriptor at point C under hypothesis $(s, h, \phi, \theta, b, a)$;
            Mark point C as associated with this descriptor ;
            Save the descriptor to the navigation data ;
        end
        Save to the navigation data the reference to the descriptor to which M is associated ;
    end
end

Algorithm 2: Building the navigation data.

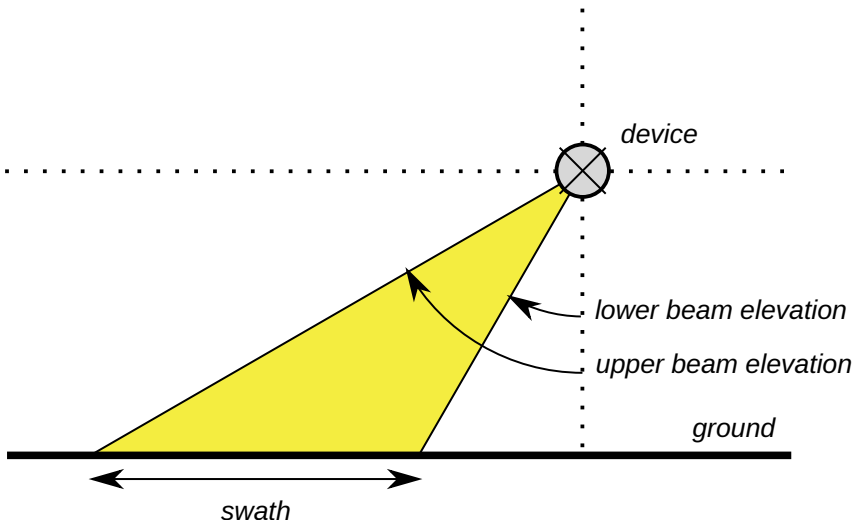


Figure 3.15 – Transverse illumination angles (in the case of an horizontal kinematic state).

3.4.7 Navigating

Estimating the kinematic state from the navigation data and from the radar observation is detailed in algorithm 3. It is composed of focusing, feature extraction and correlation operations. The algorithm lends itself to a parallel implementation since we can compute in parallel any *for* loop. They also contain no conditional branches, which is fit for execution on certain architectures (e.g. Graphics Processing Unit (GPU)). The execution time is:

— proportional to the number of unique speeds,
— proportional to the number of unique altitudes,
— proportional to the number of unique elevations,
— proportional to the number of unique orientations,
— proportional to the number of unique northings,
— proportional to the number of unique eastings,
— proportional to the total number of feature extraction points,

In addition to the navigation data, the required volume of working memory is determined by the size of the SAR image and by the array of observed descriptors. The array of observation descriptors is typically much less voluminous than the SAR image.

3.4.8 Theoretical discussion

The proposed method is simple, predictable and supports various reference geospatial data modalities. It uses all the information available in the observation SAR image, contrary to methods involving detecting and identifying specific object instances. The precision of the state estimation is determined by the state space sampling step. We could more precisely locate the optimum using a polynomial interpolation of the samples

of the cost function near the optimum. In the proposed method, the local descriptors are not required to be invariant to any class of coordinate transforms. Using local descriptors that are invariant to the transformation from IDF-ID to ground coordinates, we could further factorize descriptor extractions and decrease the execution time of the algorithm. As we have introduced it, our method requires several focalisation operations. However, it would require only one if we used an autofocus algorithm (see appendix A). We need to improve the optimization of the cost function. We currently perform grid search in a six-dimensional space. A better optimization scheme would enable to handle larger state uncertainty without a prohibitive growth of the computational burden. We may consider an approach based on generating and testing state hypotheses [67].

We identify the following failure modes:

Inconsistencies. It may happen that the reference geospatial data are incoherent with the actual ground at the time of the mission. For instance, in wartime, a location may get destroyed after the reference geospatial data are produced. An adversary may also purposedly make the location more difficult to recognize (e.g. modifying the landscape, using radar corner reflectors).

Scene ambiguity. If the mapping between the kinematic state and the radar observation is not injective, it may happen that the imaged area is indistinguishable from another area. By definition, no system can reliably estimate the kinematic state in the presence of ambiguity. However the fact that we have a reference description of the ground before the mission implies that we can evaluate during mission preparation the risk of imaging an ambiguous area. The smaller the imaged area, the higher the risk of imaging an ambiguous area.

Poor focusing. The presence of relief or flight non-linearities may hinder the SAR image quality and consequently the discriminativeness of the local features. We can predict the presence of relief during mission preparation. Autofocus algorithms can help restore image quality. We have not investigated the use of a Digital Terrain Model (DTM) for predicting features during mission preparation.

3.4.9 Implementation

In this section, we describe how the system that we described can be implemented.

Description model training tool

We require a tool to train the description models. It takes as input geospatial data of the reference modality, geospatial data of the observation modality and a mask indicating which areas to use for training. We require a mask because the observation and reference geospatial data may be incoherent in some regions, having been produced at different times for instance. The tool produces trained description models. We have already presented this topic in section 3.1.

Navigation data compilation tool

We require a tool to compile the navigation data that the device shall rely on for navigation. This tool takes as input the piece of reference geospatial data, the definition of the set of test kinematic states and the reference and observation description models. The output are opaque navigation data to be uploaded onto the device.

Navigation program

The navigation program is the online part of the state estimation system. It is the only part that is executed on the device or during the mission. Given raw radar data and navigation data, the navigation program estimates the device kinematic states by correlation between predicted and actual observations.

3.4.10 Experiments

We now report the results of estimating kinematic states using the method that we have presented in this chapter. These experiments are based on I-Master data as flight data and on Bing aerial imagery as a reference description of the ground. As explained in chapter 1.5, the I-Master images in IDF-ID geometry and the GNSS trajectories at our disposal disagree. The IDF-ID coordinates read on a SAR image and those computed from the GNSS trajectory differ. The difference corresponds to several tens or several hundreds of meters on the ground. We do not perform state estimation with image I-Master F since it only contains a bridge. We use already focused SAR images in IDF-ID coordinates. Our implementation is written in C++ and the computer with which we perform all the experiments is equiped with an *Intel Core i7-12700* CPU, 32 GiB of RAM and an *NVIDIA GeForce RTX 3080* GPU.

Each experiment consists in estimating some kinematic parameters knowing the rest of the kinematic state.

— As a first experiment, we estimate the northing and easting and set the other kinematic parameters from GNSS. The easting search interval is 15 km wide with a step of 10 m. The northing search interval is 15 km wide with a step of 10 m. The results are reported in table 3.6.

— Secondly, we estimate the altitude, northing and easting and set the other kinematic parameters from GNSS. The altitude search interval is 1 km wide with a step of 10 m. The easting search interval is 200 m wide with a step of 10 m. The northing search interval is 200 m wide with a step of 10 m. The results are reported in table 3.7.

— Thirdly, we estimate the orientation, easting and northing and set the other kinematic parameters from GNSS. The orientation search interval is 90 degrees wide with a step of 1 deg. The easting search interval is 200 m wide with a step of 10 m. The northing search interval is 200 m wide with a step of 10 m. The results are reported in table 3.8.

— Fourthly, we estimate the speed and set the other parameters from GNSS. The speed search interval is 30 m/s wide with a step of 0.1 m/s. The results are reported in table 3.9.
— Fifthly, we estimate the northing, easting, altitude, orientation and elevation and set the speed from GNSS. The orientation search interval is 8 deg wide with a step of 1 deg. The elevation search interval is 8 deg wide with a step of 1 deg. The easting search interval is 500 m wide with a step of 10 m. The northing search interval is 500 m wide with a step of 10 m. The altitude search interval is 200 m wide with a step of 10 m. The results are reported in tables 3.10 and 3.11.

Besides, we report in figure 3.16 a critical point of the cost function. The width of the correlation peak along each axis is reported in table 3.12.

The 72 estimation experiments that we have performed show that our method successfully estimates the kinematic state or part of it. We cannot precisely measure the estimation precision since the SAR images at our disposal, their respective GNSS trajectories and the earth geometry themselves do not agree to a sufficient precision. The altitude of the device, measured using GNSS, was about 3000 km. So the deviations between the state estimates and the GNSS ground-truth are a few percents of the altitude. We obtain the same results using either an OpenStreetMap or an optical reference. With either reference modality, the kinematic state estimates are approximately the same and so is the shape of the cost function near the correlation peak. A limitation of our experiments is that we have assumed to have an IDF-ID image. We have not experimentally proved in details that we could build such an image despite the unknown motion of the antenna. The computation time is higher than allowed by typical mission constraints (e.g. 1 second). However, in our experiments, the SAR image covered a ground area of about 10 km^2. The higher the area covered, the lower the probability of it being ambiguous. However, in typical mission regions, an area of 10 km^2 is more than necessary. Reducing it would also reduce the computation time. Or we could select at mission time a smaller subset of the image that we identify as unambiguous. Another approach to reduce the computation time and obtain an operational system may be to combine our system with either a parametric autofocus algorithm (see section 1.3.1) or another source of information about the kinematic state (such as an altimeter). For instance, we could estimate the speed with a parametric autofocus algorithm, the altitude with a radar altimeter, consider the elevation null and estimate the northing, easting and orientation using our method. Another line of improvement may be the optimization of the cost function. However, a correlation-based cost function necessarily has many convexity pools so a correlation-based method cannot avoid massive sampling of the state space. At best, it can perform it quickly.

In future works, we shall investigate descriptors that are invariant to the transformation between IDF-ID and ground coordinates (or between TCA-CD and ground coordinates) and that are binary. A 64-bit binary descriptor is 100 times faster to compare than a 125-dimensional floating point descriptor and is 60 times more compact. Invariance to the transformation between IDF-ID and ground coordinates would enable further

factorizations and may be achieved by modifying the descriptors training procedure.

3.5 Conclusion

In this chapter, we have investigated the use of a learned local multimodal similarity metric to estimate the kinematic state of the device. We have reviewed the local descriptors reported in the literature, either handcrafted or learned. We have demonstrated with extensive experiments that estimating a translation using learned multimodal local descriptors is very reliable and has a practical computation load despite the modality gap and the imbalance of the observation and reference areas. Finally, we have investigated how to determine the kinematic state completely. We determine a kinematic state on the hypothesis of which the SAR image in ground geometry is most similar to the reference piece of geospatial data. However, the computation cost of this approach is not practical yet. As a continuation of this work, achieving a better computation cost may involve using local descriptors that are binary and that are invariant to the transformation between IDF-ID and ground coordinates.

Input: The navigation data.
Input: The radar observation.
Output: The estimated kinematic state.
likeliest_state ← None ;
for *each speed* s **do**
  Construct a SAR image supposing speed s ;
  for *each combination of altitude, elevation* (h, ϕ) **do**
    Extract descriptors at points $expoints[s, h, \phi]$;
    for *each combination of orientation, northing and easting* (θ, b, a) **do**
      Compute the correlation between the observation and reference descriptors ;
      if *the correlation is better than the current best correlation* **then**
        likeliest_state ← $(s, h, \phi, \theta, b, a)$;
      end
    end
  end
end

Algorithm 3: Estimating the kinematic state.

Scene	**Reference**	**East. err.**	**North. err.**	**Exec. time**
A	Bing	-20 m	-60 m	6.4 s
A	OpenStreetMap	-10 m	-60 m	6.4 s
B	Bing	-10 m	130 m	4.7 s
B	OpenStreetMap	-10 m	130 m	4.3 s
C	Bing	-50 m	50 m	9.0 s
C	OpenStreetMap	-50 m	50 m	9.2 s
D	Bing	10 m	-60 m	10.4 s
D	OpenStreetMap	10 m	-50 m	10.4 s
E	Bing	10 m	-20 m	6.2 s
E	OpenStreetMap	10 m	-20 m	5.8 s
G	Bing	20 m	-70 m	3.8 s
G	OpenStreetMap	10 m	-70 m	3.6 s

Table 3.6 – Easting and northing estimation errors (optimizing translation only).

Scene	Reference	Alt. err.	Exec. time
A	Bing	30 m	17.1 s
A	OpenStreetMap	0 m	17.1 s
B	Bing	20 m	12.6 s
B	OpenStreetMap	-20 m	12.6 s
C	Bing	100 m	23.1 s
C	OpenStreetMap	110 m	23.0 s
D	Bing	20 m	23.1 s
D	OpenStreetMap	20 m	23.1 s
E	Bing	0 m	15.2 s
E	OpenStreetMap	-40 m	15.2 s
G	Bing	10 m	10.8 s
G	OpenStreetMap	10 m	10.7 s

Table 3.7 – Altitude estimation errors (optimizing altitude and small translation).

Scene	Reference	Orient. err.	Exec. time
A	Bing	0 deg	0.6 s
A	OpenStreetMap	0 deg	0.6 s
B	Bing	1 deg	0.5 s
B	OpenStreetMap	1 deg	0.5 s
C	Bing	0 deg	0.6 s
C	OpenStreetMap	0 deg	0.7 s
D	Bing	0 deg	0.7 s
D	OpenStreetMap	0 deg	0.7 s
E	Bing	0 deg	0.6 s
E	OpenStreetMap	0 deg	0.6 s
G	Bing	0 deg	0.5 s
G	OpenStreetMap	0 deg	0.5 s

Table 3.8 – Orientation estimation errors (optimizing orientation and small translation).

Scene	Reference	Speed err.
A	Bing	-0.2 m/s
A	OpenStreetMap	-0.1 m/s
B	Bing	0.3 m/s
B	OpenStreetMap	0.1 m/s
C	Bing	0.3 m/s
C	OpenStreetMap	0.1 m/s
D	Bing	-0.1 m/s
D	OpenStreetMap	0 m/s
E	Bing	0.2 m/s
E	OpenStreetMap	0.1 m/s
G	Bing	-0.1 m/s
G	OpenStreetMap	0 m/s

Table 3.9 – Speed estimation errors (optimizing all kinematic parameters but the speed).

Scene	Reference	East. err.	North. err.	Alt. err.
A	Bing	-40 m	-40 m	30 m
A	OpenStreetMap	0 m	-60 m	-10 m
B	Bing	0 m	130 m	20 m
B	OpenStreetMap	-20 m	130 m	-10 m
C	Bing	20 m	100 m	90 m
C	OpenStreetMap	30 m	100 m	90 m
D	Bing	0 m	-60 m	10 m
D	OpenStreetMap	-10 m	-50 m	20 m
E	Bing	10 m	-20 m	0 m
E	OpenStreetMap	40 m	0 m	-40 m
G	Bing	10 m	-70 m	10 m
G	OpenStreetMap	10 m	-70 m	10 m

Table 3.10 – Easting, northing and altitude estimation errors (optimizing all kinematic parameters but the speed).

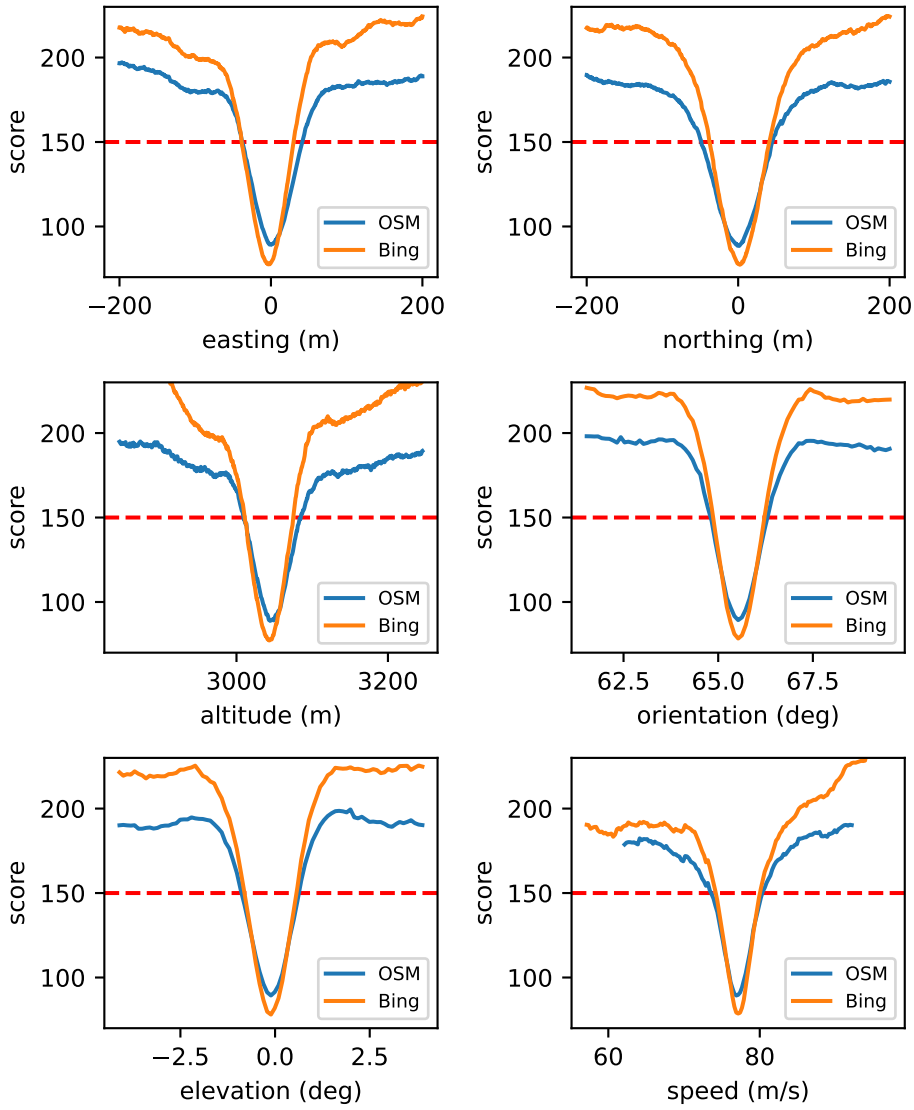


Figure 3.16 – Values of the cost function in the vicinity of a critical point along each axis of the space of kinematic states. Blue is using OpenStreetMap and orange is using Bing.

Scene	Reference	Elev. err.	Orient. err.	Exec. time
A	Bing	0 deg	0 deg	48.1 s
A	OpenStreetMap	0 deg	0 deg	48.2 s
B	Bing	0 deg	0 deg	41.1 s
B	OpenStreetMap	0 deg	0 deg	41.1 s
C	Bing	-1 deg	-1 deg	71.6 s
C	OpenStreetMap	1 deg	1 deg	70.8 s
D	Bing	0 deg	0 deg	68.7 s
D	OpenStreetMap	0 deg	0 deg	68.3 s
E	Bing	0 deg	0 deg	43.9 s
E	OpenStreetMap	0 deg	0 deg	44.0 s
G	Bing	0 deg	0 deg	28.8 s
G	OpenStreetMap	0 deg	0 deg	28.8 s

Table 3.11 – Elevation and orientation estimation errors (optimizing all kinematic parameters but the speed).

Axis	Peak width (OpenStreetMap)	Peak width (Bing)	Unit
speed	6.8	5.8	m/s
elevation	1.5	1.4	deg
orientation	1.5	1.4	deg
northing	93.8	78.6	m
easting	79.8	67.9	m
altitude	75.5	64.1	m

Table 3.12 – Width of the correlation peak along each axis of the space of kinematic states.

Chapter 4

State estimation based on scene coordinates regression

In this chapter, we investigate a kinematic state estimation system based on Scene Coordinate Regression (SCR). A neural network predicts the geographic coordinates corresponding to image patches in Instantaneous Doppler Frequency (IDF) / Instantaneous Distance (ID) geometry. Then, the kinematic state is determined from the correspondences between geographic coordinates and IDF-ID coordinates. We train the neural network on reference geocoded image patches resampled into the IDF-ID geometry. A strong limitation of our investigations is that we do not conduct experiments to demonstrate generalization from the reference geospatial product to the observation geospatial product. In particular, we did not succeed in demonstrating generalization between geospatial products of different modalities. This violates the hypothesis of our work that the reference and observation geospatial products have different modalities. However, we demonstrate the feasibility of rest of the pipeline. This chapter introduces SCR, describes its application to our problem, reports early experimental results and discusses paths toward satisfying all the constraints of our problem.

4.1 Scene coordinates regression

In [198], Shotton *et al* demonstrated training a random forest to predict the spatial coordinates of a point from simple depth and RGB comparison features. The pose of the RGB-D camera is then determined robustly from the predicted correspondences using RANSAC. The use of a neural network instead of a regression forest to predict spatial coordinates from an image patch was reported in [26]. In [134], the spatial coordinates are estimated not for a few image patches but for every image pixel. A weakness of these methods is the need of images of the scene of interest labeled with spatial coordinates. In [249], in the context of aerial navigation, it was proposed to train the scene coordinates regression on synthetic data.

4.2 Proposed method

Our proposed method is represented in figure 4.1. Given a geocoded image of the scene with the mission sensor, we train a neural network to predict geographic coordinates from image patches in IDF-ID geometry. The training image patches are synthesized by

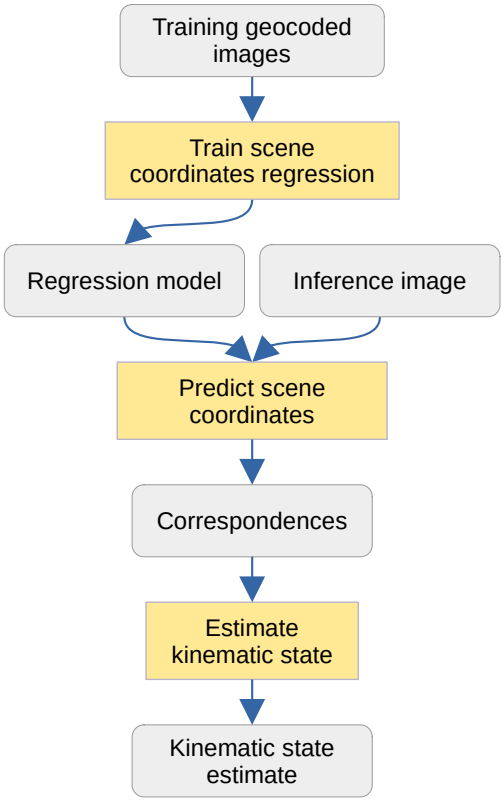


Figure 4.1 – Kinematic state estimation based on scene coordinates regression.

sampling kinematic states and resampling the reference geocoded training image (of the mission location) from ground geometry to IDF-ID geometry. Once the regression model is trained, at mission time, we predict geographic coordinates at a few locations of an image in IDF-ID geometry. From the correspondences between geographic coordinates and IDF-ID coordinates, we determine the kinematic state.

4.2.1 Generating training image patches

The process for generating a training patch is as follows:

1. We sample a kinematic state. For that, in a local tangent frame, we sample an altitude, a northing, an easting, a speed, an elevation angle and an orientation angle between the velocity vector and the ground geographic axes. Each of these six values are sampled uniformly in user-defined ranges that represent the initial kinematic state uncertainty.
2. We sample a patch window. The user defines the intervals of IDF and ID coordinates covered by the mission-time SAR image. Uniformly within these bounds, we select a range of IDF and ID covered by the image patch.

3. We resample the reference image into a synthetic observation patch, from ground geometry to IDF-ID geometry. We also associate to each patch the geographic coordinates of its center point. Our resampling relies on several subsamples per pixel and on linear interpolation from IDF-ID to ground.
4. We normalize each patch to zero mean and unit variance.

The parameters for the generation of the training image patches are:
— The input geocoded reference image.
— The WGS84 coordinates of a ground point at which to set the local tangent frame.
— In the local tangent frame, the intervals of northing, easting, altitude, speed, elevation angle and orientation angle from which to sample kinematic states uniformly.
— The IDF and ID bounds of the mission time SAR image.
— The length of the interval of IDF and ID coordinates covered by a patch.
— The patch width and height in pixels.

4.2.2 Regression model architecture and training

The neural network that we use in our experiments is ResNet-18 [86], a network that we expect to be easy to train. It makes use of skip connections, which facilitates the propagation of information, and it has the smallest capacity in the ResNet family, in terms of number of layers and parameters. The network input (in our case) is a single channel (intensity) image patch. The output coordinates are quantized. The model produces one probability histogram for the easting coordinate and another one for the northing coordinate. A refinement of our model may consist in directly producing real number coordinates, without quantization, using a cost function inspired from DSAC [26]. We train the model from scratch using the cross-entropy loss. Important parameters of the regression model are:
— The interval and number of bins for discretizing the northing coordinate.
— The interval and number of bins for discretizing the easting coordinate.

4.2.3 Kinematic state estimation

During the mission, the algorithm extracts patches from the observation image, in IDF-ID coordinates, and the regression model predicts their corresponding geographic coordinates. The patch windows may depend on the observation image since the part of the IDF-ID domain that contains radar echoes depends on the kinematic state, as seen in section 2.6.4. Secondly, given a set of correspondences between IDF-ID and geographic coordinates, the algorithm estimates the kinematic state as follows:

1. It produces an initial estimate from three correspondences (i.e. the minimum number of correspondences).
2. It refines the initial estimate with a numerical optimization scheme that takes all the correspondences into account.

We have already introduced the optimization-based refinement in section 2.2 so we now describe how to compute the initial estimate. Our notations are:

— $\boldsymbol{o}$ the device position,
— $\boldsymbol{v}$ the device velocity,
— $\boldsymbol{p}_i$ the spatial coordinates of ground point i,
— f_i the observed IDF of ground point i,
— d_i the observed ID to ground point i,
— λ the wavelength.

The position $\boldsymbol{o}$ is at the intersection of three spheres. The respective centers of these spheres are $\boldsymbol{p}_1$, $\boldsymbol{p}_2$ and $\boldsymbol{p}_3$. Their respective radii are d_1, d_2 and d_3. The intersection of the first two spheres is a circle in space. The intersection of that circle with the third sphere is composed of two intersection points. One of them is below the ground and the other is the position $\boldsymbol{o}$.

The procedure for computing $\boldsymbol{o}$ that we shall describe hereafter depends on solving circle-circle intersections. So we first detail a procedure for solving circle-circle intersections. Finding the intersection of two circles in a plane can be reduced without loss of generality to finding the intersection of a circle of unit radius centered at the origin and a circle of any radius R whose center lies on the x-axis at coordinate l. This is illustrated by figure 4.3. We denote α the angle of a point on the first circle. Applying the law of cosines, we get that:

$$R^2 = 1 + l^2 - 2l\cos\alpha \tag{4.1}$$

$$\cos\alpha = \frac{1 + l^2 - R^2}{2l} \tag{4.2}$$

The coordinates of the intersection points are:

$$\left(\cos\alpha; \pm\sqrt{1 - \cos^2\alpha}\right) \tag{4.3}$$

We define:

$$\begin{aligned} f_x(R, l) &:= \frac{1 + l^2 - R^2}{2l} \\ f_y(R, l) &:= \sqrt{1 - f_x^2(R, l)} \end{aligned} \tag{4.4}$$

These functions define the coordinates of a solution of the normalized circle-circle intersection problem.

Now that we have explained how to solve the normalized circle-circle intersection problem, we return to finding the intersection of three spheres. We first consider the intersection of spheres $(\boldsymbol{p}_1, d_1)$ and $(\boldsymbol{p}_2, d_2)$. This intersection is a circle whose center lies one line $(\boldsymbol{p}_1\boldsymbol{p}_2)$. Using the Gram-Schmidt procedure, we determine an orthogonal basis $(\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3)$ such that:

$$\boldsymbol{u}_1 = \frac{1}{\|\boldsymbol{p}_2 - \boldsymbol{p}_1\|}(\boldsymbol{p}_2 - \boldsymbol{p}_1) \tag{4.5}$$

We note $\boldsymbol{U}$ the matrix such that:

$$\boldsymbol{U} = \begin{pmatrix} \boldsymbol{u}_1 & \boldsymbol{u}_2 & \boldsymbol{u}_3 \end{pmatrix} \tag{4.6}$$

Applying the results from normalized circle-circle intersection, we find that the intersection of the first two spheres is the circle with center $\boldsymbol{c}_1$, radius r_1 and axes $\boldsymbol{u}_2$ and $\boldsymbol{u}_3$.

$$\begin{aligned} \boldsymbol{c}_1 &:= \boldsymbol{p}_1 + d_1 f_x \left(\frac{d_2}{d_1}, \frac{\|\boldsymbol{p}_2 - \boldsymbol{p}_1\|}{d_1} \right) \boldsymbol{u}_1 \\ r_1 &:= d_1 f_y \left(\frac{d_2}{d_1}, \frac{\|\boldsymbol{p}_2 - \boldsymbol{p}_1\|}{d_1} \right) \end{aligned} \tag{4.7}$$

We must now determine the intersection of this circle with the third sphere. For that, we note that the intersection of the third sphere with the plane that contains this circle is another circle, so that we shall be left with another circle-circle intersection problem. The intersection of the third sphere with plane $(\boldsymbol{c}_1, \boldsymbol{u}_1)$ has center $\boldsymbol{c}_2$ and radius r_2:

$$\begin{aligned} \boldsymbol{c}_2 &:= \boldsymbol{c}_1 + \left(\boldsymbol{I} - \boldsymbol{u}_1 \boldsymbol{u}_1^T \right) (\boldsymbol{p}_3 - \boldsymbol{c}_1) \\ r_2 &:= \sqrt{d_3^2 - \left(\boldsymbol{u}_1^T (\boldsymbol{p}_3 - \boldsymbol{c}_1) \right)^2} \end{aligned} \tag{4.8}$$

Here, $\boldsymbol{I}$ designates the identity matrix. We are now left with the final circle-circle intersection. In the plane whose origin is $\boldsymbol{c}_1$ and whose axes are $\boldsymbol{u}_2$ and $\boldsymbol{u}_3$, the first circle is centered at $(0, 0)$ and has radius r_1 and the second circle is centered at $\begin{pmatrix} \boldsymbol{u}_2^T \\ \boldsymbol{u}_3^T \end{pmatrix} (\boldsymbol{p}_3 - \boldsymbol{c}_1)$ and has radius r_2. The two intersection points can be found by reduction to the normalized circle-circle intersection problem. One point is below the ground and the other is $\boldsymbol{o}$.

Now that we have explained how to determine $\boldsymbol{o}$, we are left with explaining how to determine $\boldsymbol{v}$. This is much more straightforward. The velocity vector $\boldsymbol{v}$ is the solution of this linear system:

$$\frac{2}{\lambda} \begin{pmatrix} \frac{1}{\|\boldsymbol{p}_1 - \boldsymbol{o}\|} (\boldsymbol{p}_1 - \boldsymbol{o})^T \\ \frac{1}{\|\boldsymbol{p}_2 - \boldsymbol{o}\|} (\boldsymbol{p}_2 - \boldsymbol{o})^T \\ \frac{1}{\|\boldsymbol{p}_3 - \boldsymbol{o}\|} (\boldsymbol{p}_3 - \boldsymbol{o})^T \end{pmatrix} \boldsymbol{v} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} \tag{4.9}$$

This completes our presentation of how to determine the kinematic state $(\boldsymbol{o}, \boldsymbol{v})$ from the minimum number of correspondences.

The iterative refinement algorithm was introduced in section 2.2 and involves optimizing the cost function defined by equation 2.49 given a set of correspondences between ground and IDF-ID coordinates.

The inference procedure is depicted in figure 4.4. Note that we have not implemented this procedure, as explained in the following section.

4.3 Experiments

In this section, we describe our experiments consisting in training a scene coordinates regression model and in infering kinematic states. Evaluating the system as presented in figure 4.4 would require as many SAR images as realizations of the kinematic state prediction error. Also, since we do not have two different geospatial products of the same modality and of the same location, and since we have not achieved generalization from one modality to another, we can only test on the training data themselves (but we do not test on the training resampled image patches) and consequently do not prove generalization between different pieces of geospatial data. Our testing procedure is presented in figure 4.5. We sample kinematic states and resample the geospatial data to simulate IDF-IF image patches extracted from the observation. We estimate the kinematic state from these images patches and compare the estimated kinematic state with the sampled kinematic state.

Our experiments are based on a grayscale optical image from Karlsruhe at a resolution of 2 m/pixel. The parameters for sampling kinematic states for training and evaluation are reported in table 4.1. The rest of our settings are reported in table 4.2. A few sample image patches generated from the reference geocoded image are represented in figure 4.6. The discretization step for easting and northing is approximately 9.8 m. We train our model for 14 hours with about 10 million simulated image patches. We use Stochastic Gradient Descent (SGD) with a learning rate of 10^{-2} and a batch size of 32. The evolution of the training loss is represented in figure 4.7.

The windows at which we extract correspondences are represented in figure 4.8. There are 13 windows so we estimate the kinematic state from 13 correspondences between geographic coordinates and IDF-ID coordinates. Finally, we perform 8918 experiments consisting in sampling a kinematic state and estimating it using our method. we resample training and testing image patches from the same image. Estimation performances would likely be lower on different image. Each estimation takes 14 milliseconds on our computer equiped with an *Intel Core i7-12700* CPU and a *GeForce RTX 3080* GPU. The distribution of estimation errors is represented in figure 4.9. The median absolute estimation errors are:

— $P(|\text{position_error_x}| < 12.2) = 0.5$
— $P(|\text{position_error_y}| < 17.6) = 0.5$
— $P(|\text{position_error_z}| < 22.9) = 0.5$
— $P(|\text{velocity_error_x}| < 0.4) = 0.5$
— $P(|\text{velocity_error_y}| < 0.5) = 0.5$
— $P(|\text{velocity_error_z}| < 4.5) = 0.5$

We observe that the neural network correctly predicts the geographic coordinates. The precision limiting factor is the quantization noise, since the errors are distributed approximately uniformly over the quantization interval. This is seen in the first plot of figure 4.9. The vertical component of the velocity is mathematically more sensitive than the horizontal components. A further work should focus on reducing the quantization noise,

Parameter	**Interval center**	**Interval length**
Northing	0 m	4 km
Easting	0 m	4 km
Altitude	1 km	200 m
Elevation	0 deg	20 deg
Orientation	90 deg	20 deg
Speed	250 m/s	10 m

Table 4.1 – Parameters for kinematic states sampling.

Parameter	**Value**
Minimum imaging Doppler frequency	-5 kHz
Maximum imaging Doppler frequency	5 kHz
Minimum imaging distance	1.5 km
Maximum imaging distance	3 km
Window Doppler frequency range	2 kHz
Window distance range	200 m
Patch raster width	244 pixels
Patch raster height	244 pixels
Number of easting bins	1024 bins
Number of northing bins	1024 bins
Output easting range	10 km
Output northing range	10 km

Table 4.2 – Other parameters.

maybe taking inspiration from DSAC [26]. The inference time is small (14 milliseconds) and, in a first approximation, independent of the extent of the initial uncertainty. The precision could be improved by predicting a higher number of correspondences and reducing the quantization noise. There was a negligible amount of outliers in the predicted geographic coordinates so we have not integrated an outlier filtering technique such as RANdom SAmple Consensus (RANSAC) in the state estimation algorithm.

The experiments that we have presented are a first prototype that call for further investigations. Their limit is that we have used the same image for inference and evaluation and that it was an optical image. We have not studied the generalization between two images of the same modality or of different modalities.

4.4 Towards multimodal SCR

In this chapter, up to now, we have not addressed the fact that the reference and observation modalities have to be different, violating a constraint that we set out in chapter 1. A monomodal SCR could still be interesting in some mission settings. For instance, if we can image the region of interest before the mission using the mission sensor, the device may use a monomodal SCR algorithm to localize itself. That being said, how could we hopefully obtain a multimodal SCR algorithm?

4.4.1 Directions to investigate

Multimodal SCR finds itself beyond the current state of the art. As depicted in table 4.3, given colocated SAR and optical images of some sample locations and given an optical image of the mission location, we want to train an algorithm to infer geographic coordinates on a SAR image of the mission location.

Let us first consider approaches based on discriminative modelling. As depicted in figure 4.10, this involves bridging the modality gap. We need a procedure to embed SAR image patches and optical image patches into a common representation space from which to predict geographic coordinates. This representation has to satisfy two properties:

— Representations of the same place computed from either SAR or optical image patches are equivalent from the point of view of the coordinates regression procedure.

— Geographic coordinates should be predictable from a normalized representation.

We may construct the normalization procedures together with the regression procedure so that they optimize a combination of these two objectives or we may construct them so that they optimize a surrogate objective. In the first case, we may resort to adversarial domain adaptation [222]. In an adversarial game, one player strives to predict geographic coordinates from normalized representations and another player strives to predict the modalities from which the representations originate. In the second case, we may resort for instance to image translation [100, 266, 7, 71]. Image translation (not to be confused with a geometric operation in this context) designates a task studied in the field of computer

vision: transforming an image from one modality to another. Ideally, in order to avoid performing image translation at mission time, one would transform the reference modality (e.g. optical) into the observation modality (i.e. SAR) instead of the opposite. However, the quality of the transformation and so the accuracy of the navigation system may not be equal in both directions. Converting from optical to SAR is also rather unusual in the scientific literature since the purpose of image translation is generally to ease the interpretation of SAR images. Finally, we may use local descriptors. As presented in chapter 3, local descriptors are representations that uniquely identify geographic points and are independant of the original modality. In theory, local descriptors satisfy the two desired properties. It remains to be seen experimentally whether this is achieved well enough in practice. We should first train local descriptors, secondly train a coordinates regression model on descriptors extracted from the reference modality and thirdly evaluate the regression model on descriptors extracted from the observation modality.

Let us secondly consider the path of generative modelling. In the last few years, researchers have proposed multimodal variational autoencoders [237] [197] [58]. They aim to learn latent representations composed of shared and modality-specific parts. They can be used to learn to perform operations on images (i.e. image completion, edge detection). In our case, we would hopefully determine the geographic coordinates from the shared latent code. We can obtain a shared latent code from either modality. In practice, on the colocated SAR and optical images of the sample places, we would fit a multimodal variational autoencoder. On the optical image of the mission place, we would learn to predict geographic coordinates from shared latent codes computed from optical image patches. Finally, on the SAR image of the mission place, we would predict geographic coordinates from shared latent codes computed from SAR image patches.

4.4.2 What does not work

Due to lack of time, we have not explored all the approaches that we have mentioned. We have experimented with two approaches that proved unsuccessful.

The first one is an instance of adversarial training. It involves four neural networks.

— We intend the SAR normalization network to normalize SAR image patches into interoperable abstract representations.
— We intend optical normalization network to normalize optical image patches into interoperable abstract representations.
— We intend discrimination network to predict the original modality of an abstract representation (i.e. SAR versus optical).
— We intend regression network to predict geographic coordinates from abstract representations.

At training, we alternatively optimize the discrimination network on the one hand and the other networks on the other hand.

— We train the discrimination network to predict whether abstract representations originate from SAR or optical image patches, using the sample places and not the

mission place.

— We train the normalization networks and the regression network to produce abstract representations that confound the discriminator (on the sample places) and to predict geographic coordinates from optical image patches (on the mission place).

At evaluation, we predict geographic coordinates from SAR image patches on the mission place.

The theoretical problem with this approach, however, is that the optical patch normalization network may very well produce modality agnostic abstract representations only on the sample places. It may learn to behave differently on the mission place and on the sample places. If we trained the discriminator on the optical image patches of the mission place in addition to the sample places, it may learn to recognize the mission place, never seeing SAR image patches of the mission place.

Another unsuccessful approach that we have experimented with is as follows. There are one SAR normalization network and one optical normalization network. They transform respectively SAR and optical image patches into (hopefully) interoperable abstract representations. There is one coordinates regression network for each place (one for each sample place and one for the mission place). They predict geographic coordinates from abstract representations. We first train all the optical normalization networks and all the regression networks, on the optical images of all places, on the task of predicting geographic coordinates. Afterward, we train the SAR normalization network, on the sample places, on the task of predicting geographic coordinates (we freeze the regression networks of the sample places). At evaluation, we use the SAR normalization network trained at the second stage and the regression network trained at the first stage. The problem with this approach is that the regression networks are too dependent on the optical modality.

4.5 Conclusion

In this chapter, we have proposed a kinematic state estimation system based on SCR. We synthesize training observation patches by resampling a reference geocoded image. We have described how to estimate the kinematic state from the minimum number of correspondences. We have reported experimental results showing that the system is promising in terms of estimating the complete kinematic state at a very low computation cost. Further work is still required in order to comply with the hypothesis that the observation and the reference have different modalities.

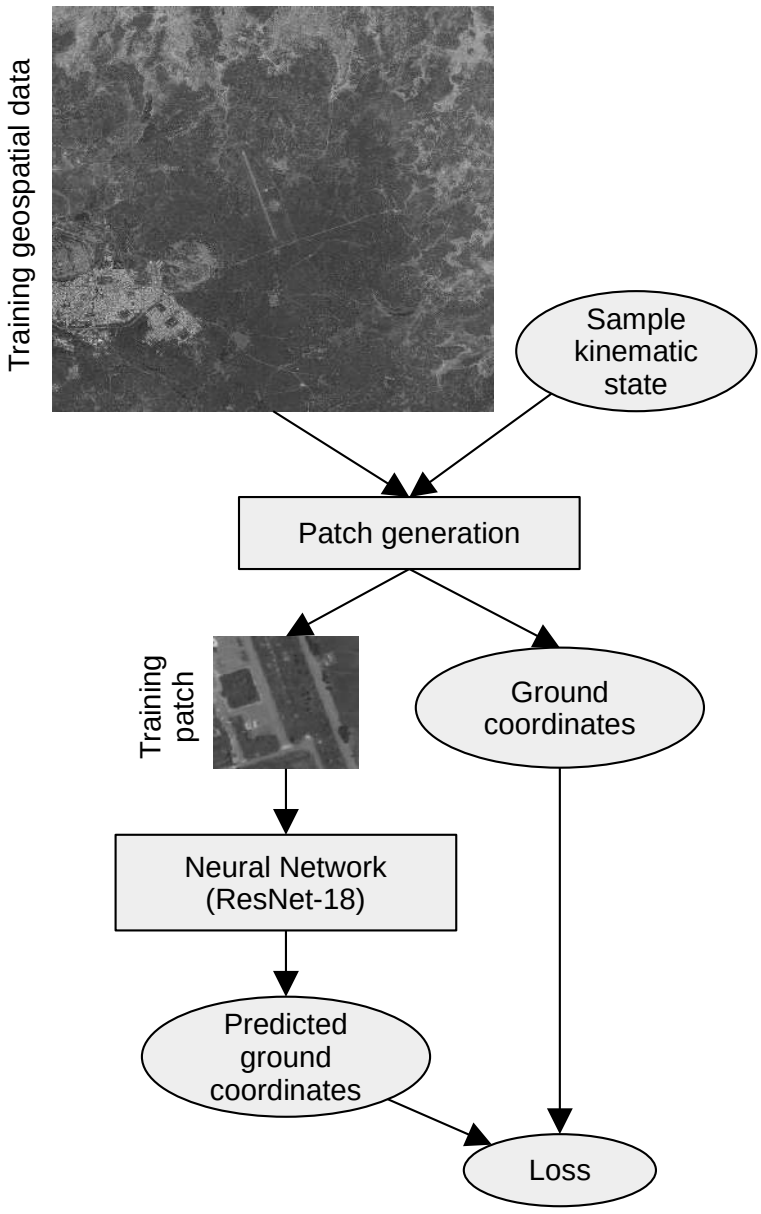


Figure 4.2 – Our SCR training procedure.

	Sample place 1	**...**	**Sample place N**	**Mission place**
Optical	For training	...	For training	For training
SAR	For training	...	For training	For inference

Table 4.3 – Data usage for multimodal SCR.

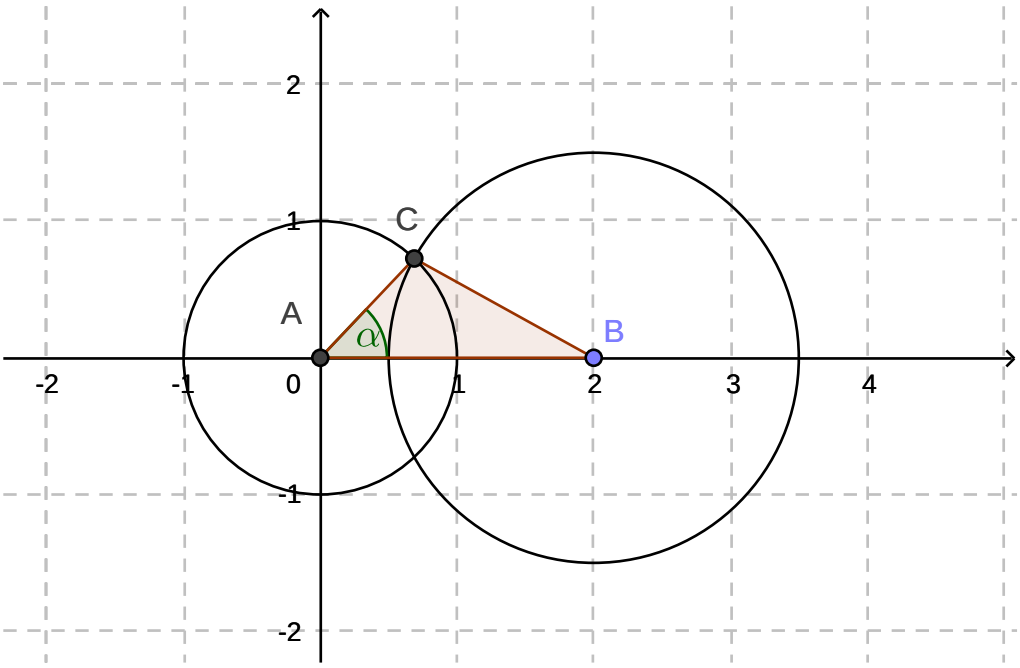


Figure 4.3 – Normalized circle-circle intersection problem.

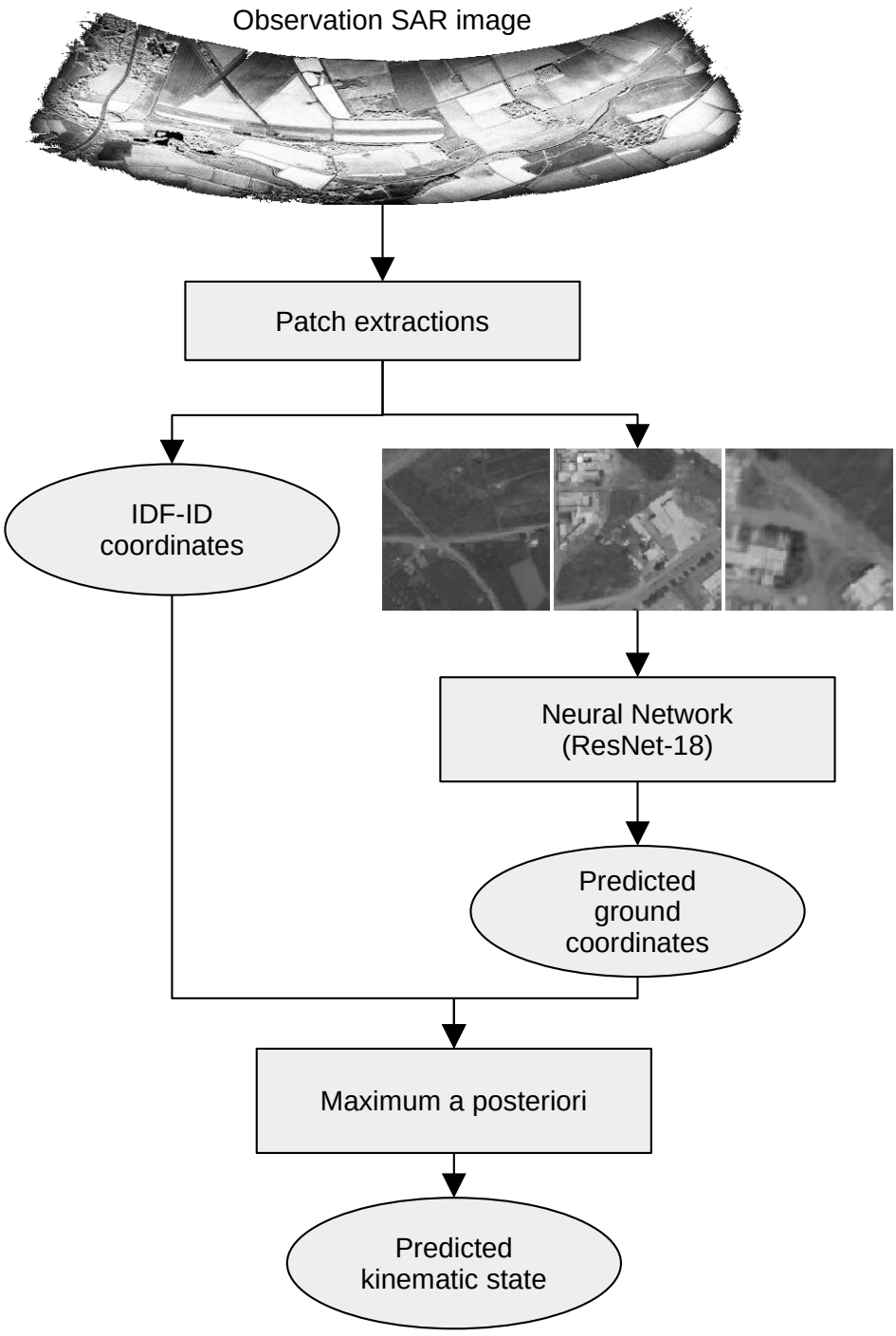


Figure 4.4 – Intended (but unimplemented) inference procedure.

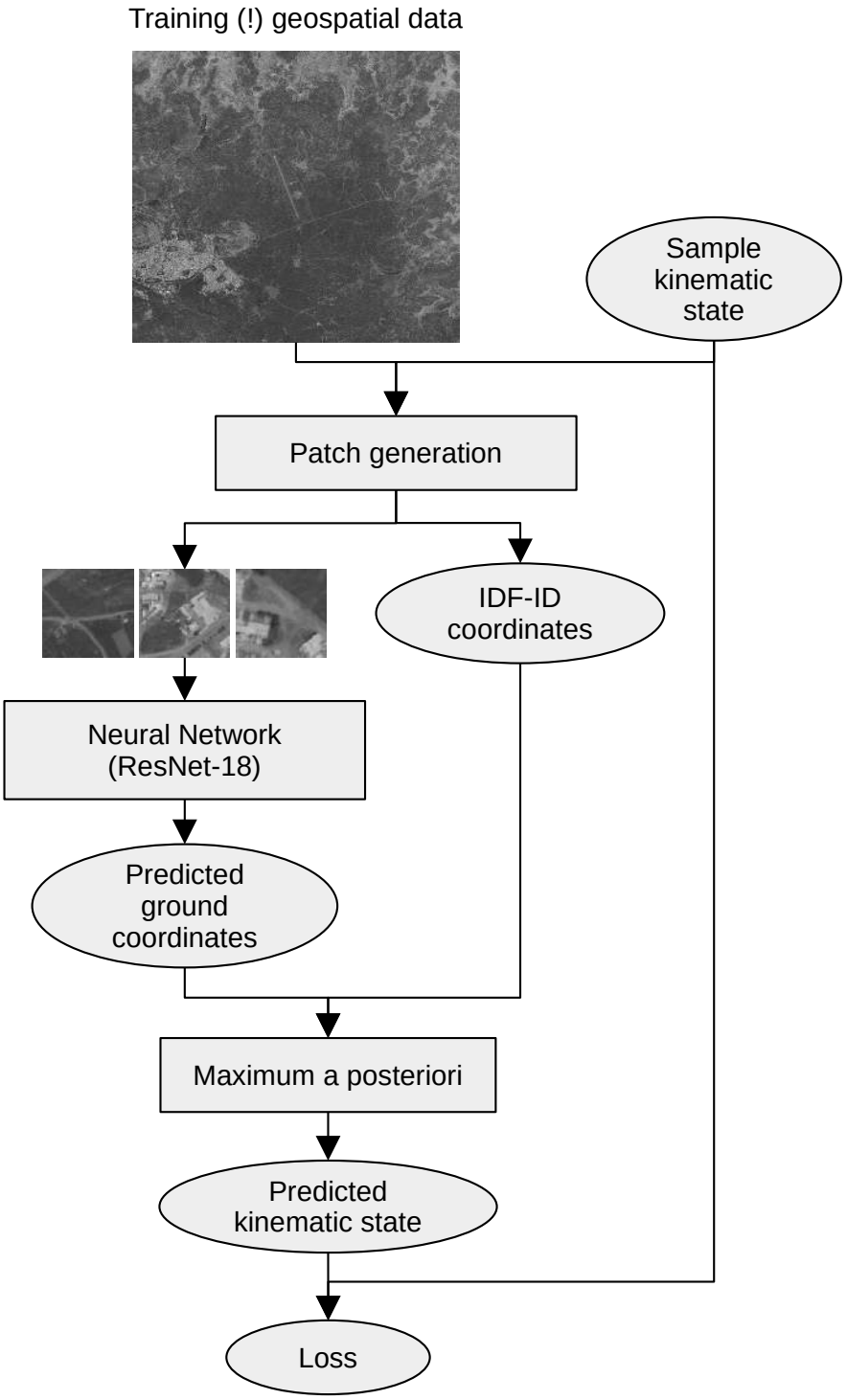


Figure 4.5 – Implemented testing procedure.



Figure 4.6 – A few image patches resampled from an optical image.

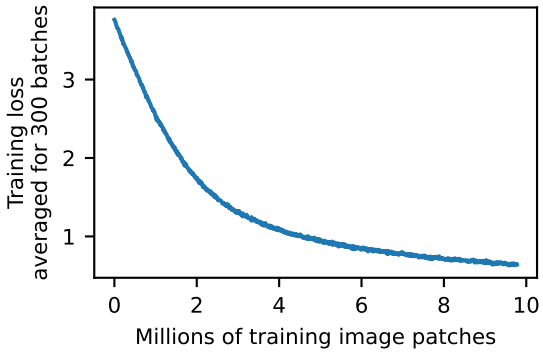


Figure 4.7 – SCR Training loss.

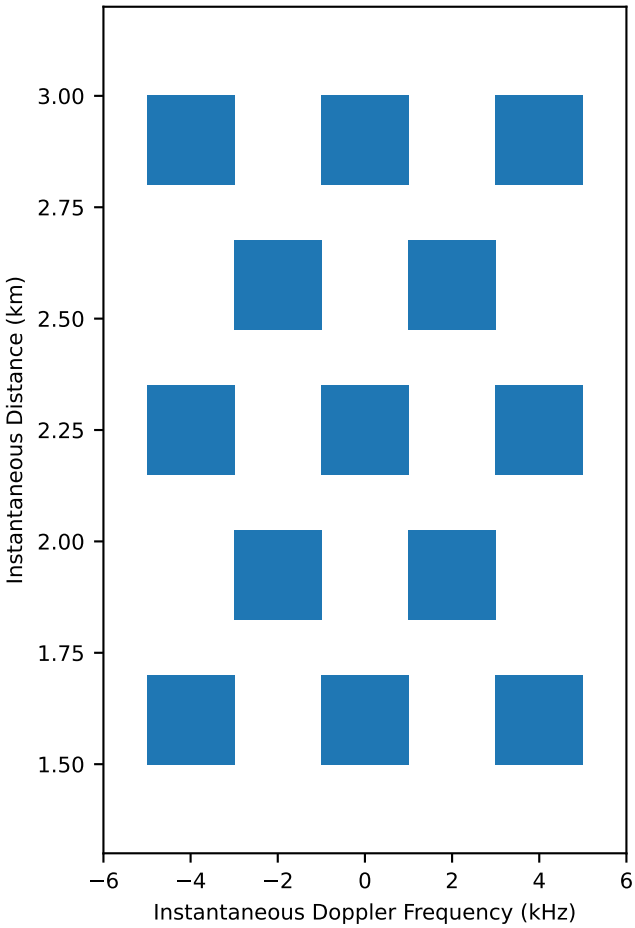


Figure 4.8 – The windows, on an image in IDF-ID geometry, at which to extract correspondences. Each window is colored in blue. The horizontal axis represents Doppler frequency. The vertical axis represents distance.

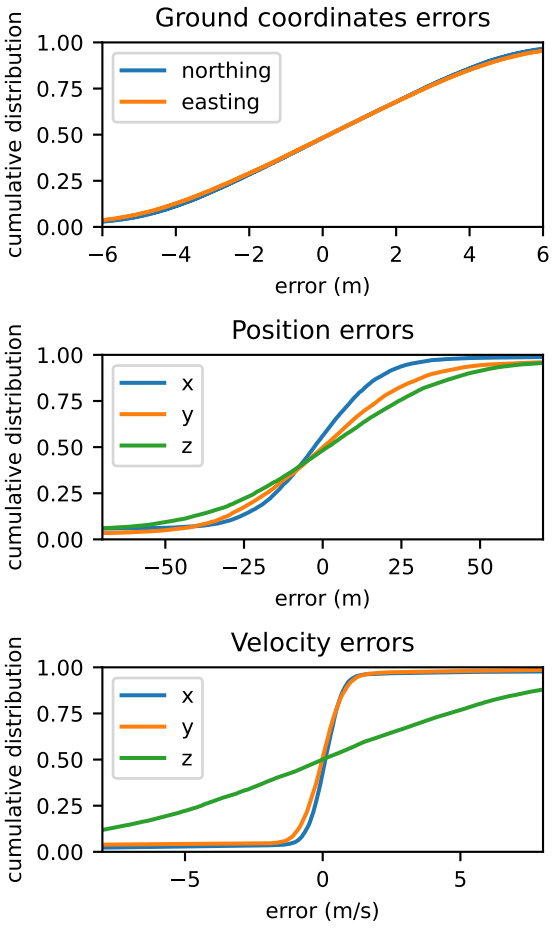


Figure 4.9 – Distribution of kinematic state estimation errors.

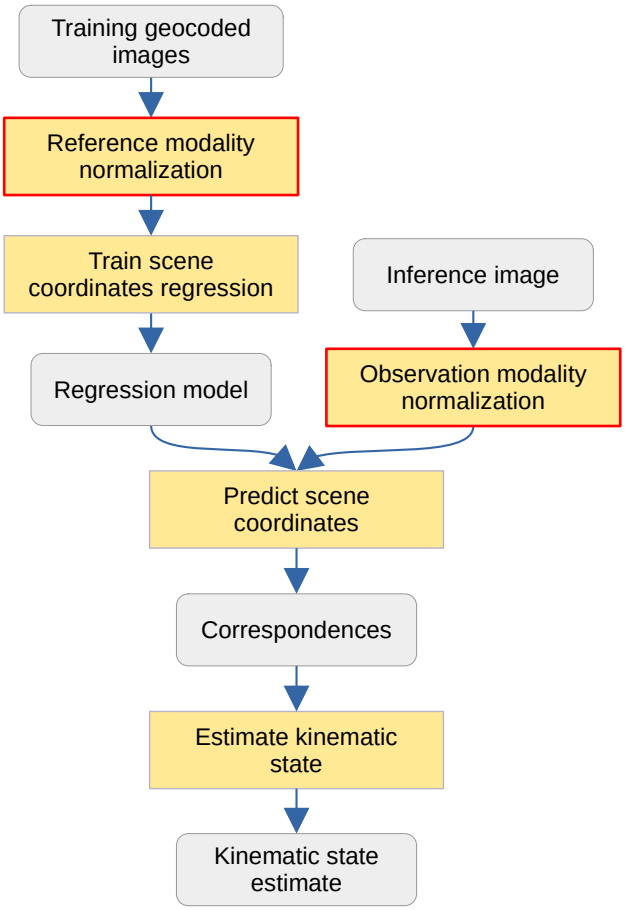


Figure 4.10 – Sketch of a kinematic state estimation system based on multimodal SCR.

Conclusion

We have sought algorithms for determining the position and velocity of an aerial device relative to the ground given a radar observation of the ground, assuming that the radar observation of the ground can be focused into a SAR image. All the prior information about the mission geographic environment is represented as geospatial data. The modality of these data is generally different from the mission sensor. Our experiments have involved geospatial data modalities as diverse as OpenStreetMap, Miranda (SAR), I-Master (SAR), KOMPSAT-5 (SAR), Pléiades (optical), *Digitale Orthophotos* (optical) and Bing Satellite (optical). We also have to comply with mission and system constraints, especially the limited inference time and computational means.

We have viewed the state estimation problem as a data alignment problem. The position and velocity of the aerial device are the parameters of the transform that relates the coordinates associated with the observation SAR image and the coordinates associated with the reference geospatial data. The coordinates associated with the observation SAR image are either Instantaneous Doppler Frequency - Instantaneous Distance (IDF-ID) or Time of Closest Approach - Closest Distance (TCA-CD). The coordinates associated with the reference geospatial data are two-dimensional cartesian coordinates in a local tangent plane.

The difficulty of this data alignment problem lies in the facts that:

— the dimension of the state space is relatively high,
— the initial state uncertainty is large,
— the reference and the observation geospatial data have different modalities,
— simulating sufficiently realistic observations of the mission geographic environment is unavailable.

The methods that we have developed fall into two categories: alignment based on sparse correspondences and direct alignment. Alignment based on sparse correspondences involves finding corresponding features, in the reference geospatial data and in the observation SAR image, and in fitting a coordinates transform to the matched coordinates. We have shown how to fit such a transform to a minimal number of matches (with a direct numerical procedure) or to a larger number of matches (as a non-linear least square optimization problem). One may apply traditional feature detection and matching pipelines to our alignment problem.

We have shown that the Iterative Closest Point algorithm was applicable to our class of coordinate transforms. It determines the matching and the coordinates transform jointly. We have shown that, given one set-to-set correspondance between IDF-ID and ground coordinates, we could robustly estimate the full kinematic state.

Since some geospatial data alignment pipelines are based on contours, we have proposed proposed a multimodal contour detection algorithm. Given an optical image, a

neural network predicts the distance map of the contours that would be detected on a SAR image of the same location. We have implemented and successfully experimented with a complete translation estimation system based on this multimodal contour detector.

We have investigated the direct alignment of the reference and observation geospatial data. This method involves determining a coordinate transform that maximizes that the average similarity of the matched locations. We have proposed the use of learned multimodal local descriptors to measure this similarity. We have introduced algorithmic devices to get a practical computation load and data volume. We have conducted extensive experiments showing that estimating a translation within computation budget using this method is very robust. We have also showed that we could successfully estimate the full kinematic state with this approach. Our system has the advantage of handling the combination of geospatial data modalities generically, which makes it more practical for operational use. It handles a large initial uncertainty, a large variety of modality combinations and is conceptually simple.

Finally, we have investigated a state estimation approach based on Scene Coordinates Regression. We predict reference coordinates directly from patches of the SAR observation. This approach promises an excellent computation efficiency: the computation load and the volume of navigation data are very small and largely independent of the initial state uncertainty. We have evaluated the potential performances of this approach in our context. New developments in the field of transfer learning are required to unlock it.

A future axis of research is to combine the methods that we have proposed, which are downstream of the image formation, with approaches that determine part of the state at the stage of focalization. As well, for better precision, practical navigation systems may involve a sequence of images rather than a single image. Further research may consequently focus on extending our methods to multiple images.

Appendix A

SAR processing with an imprecise motion estimate

We call geometric distortion the difference between the coordinates of a point read on a SAR image and the actual coordinates of the corresponding physical structure. The computer vision algorithms that we have presented in this thesis assume that we can produce a SAR image in TCA-CD or IDF-ID coordinates (as defined in section 2.1.1) and that this SAR image has negligible geometric distortions (i.e. the coordinates predicted from the image are the physical coordinates to a great precision) and sufficient quality of focus for the rest of the processing chain to work. Although this thesis focuses more on the computer vision part than the signal processing part of the kinematic state estimation system, the formation of an image in our setting requires some considerations. If this hypothesis could not be fulfilled, our system would be unimplementable. In this appendix, we discuss whether this assumption can be satisfied.

A.1 Sources of image defects

A typical SAR imaging system aims to produce an image of the ground and relies on a trajectory estimate provided by a GNSS. It is composed of the following parts:

1. **Focalisation.** In this step, the raw radar observation is processed so that the energy received from a scatterer is gathered to a single point of the image plane. Focusing algorithms are approximations to inner products between received echos and theoretical echos. Theoretical echos depend on assumptions regarding the antenna motion and the scene geometry (e.g. a constant velocity trajectory and a flat ground). Typically, at least the norm of the velocity vector has to be known. The approximations made by the focusing algorithm may introduce geometric distortions. For instance, Ribalta showed that the stop-and-go approximation introduces a geometric distortion in the case of a back-projection algorithm for a FMCW signal [180]. As well, the intra-pulse Doppler is often neglected, which may lead to geometric distortions, although these are negligible in most practical settings.
2. **Motion compensation and autofocus.** The actual motion of the antenna may not have been the motion assumed at the stage of focusing. For instance, it may have been assumed that the trajectory was uniform rectilinear but the actual trajectory deviated because of air turbulence, motor vibrations, etc. The speed estimate from an IMU may also have been incorrect. Because of the unfulfilled kinematic

and geometric assumptions, the SAR image obtained at the previous stage suffers from blurs. Motion compensation and autofocus are a collection of techniques for correcting the defects of that SAR image [40]. Motion compensation relies on an external estimate of the trajectory deviation while autofocus relies only on the SAR image. Some autofocus methods determine a trajectory correction that optimizes a criterion of image sharpness such as entropy [205] [243] or energy [74]. Other methods, such as Phase Gradient Autofocus (PGA) [226] estimate and deconvolve by the impulse response. Identification of the impulse response relies on detecting and averaging the echos of dominant scatterers. Autofocus may possibly introduce geometric distortions. For instance, if the peak of the impulse response is not centered, an autofocus based on dominant scatterers may misidentify the impulse response and introduce a shift.

3. **Resampling and geocoding.** The image is resampled from an image plane to another image plane. For instance, we may wish to have an image in a geographic geometry instead of the imaging geometry. This stage is referred to as geocoding. It relies on assumptions regarding the antenna motion and the scene geometry. Fine geocoding requires a Digital Elevation Model (DEM). This last stage of constructing a SAR image may introduce large geometric distortions because of its strong dependance on geometric assumptions.

Geocoding is a source of geometric defects that is easily avoidable in our context. In other words, the SAR image should be kept and worked on in the imaging geometry. In our context, a first estimate of the speed may come from inertial navigation but autofocusing need be employed to correct the blur.

Another line of work is blind SAR focusing, which aims to construct SAR images from raw echoes only, without any assumptions regarding the ground geometry of the trajectory. Guaragnella *et al* proposed to extract the range and azimuth impulse responses from a Singular Value Decomposition (SVD) of the raw SAR data and to use them to produce a SAR image [82, 83]. Overall, the literature on blind SAR focusing is little developed.

A.2 Illustrative experiment

In this section, as a simplistic feasability experiment and for the purpose of acquainting with the image formation process, we illustrate how to form a SAR image with an incomplete and imprecise motion estimate. We simulate radar echos from point targets and then determine the platform speed that optimizes the contrast of a SAR image.

A.2.1 Signal simulation

We simulate the signals from a SAR imaging radar with the same parameters as SAR-Light, a lightweight FMCW radar developed by ONERA [28] which can be embedded on a small consumer UAV. The carrier frequency is 9.8 GHz. The bandwidth is 600 MHz.

The chirp duration is 125 microseconds. The pulse repetition interval is 130 microseconds. We assume a sawtooth frequency. The antenna aperture is 22 degrees both in azimuth and elevation. The sampling rate is 10 MHz. We assume an uniform rectilinear trajectory at the speed of 10 m/s. We assume no squint. The kinematic and geometric configuration is shown in figure A.1.

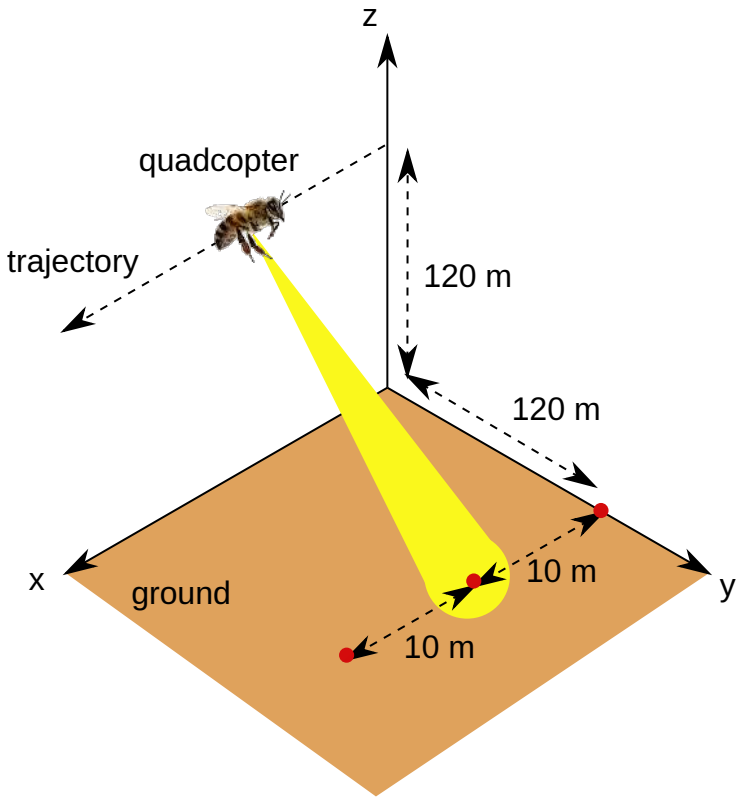


Figure A.1 – Geometric setting.

Our signal simulation program is decomposed into the following parts.

Emission A function computes the phase of the emitted signal at any time. Given the large difference in orders of magnitude between the electromagnetic times and the kinematic times and given that we do not separate them (i.e. we do not make the stop-and-go approximation), our simulator uses a floatting point arithmetic with 50 decimals instead of the standard IEEE 64 bit floating point arithmetic (which has about 14 decimals).

Reception and demodulation A function computes the phase of the received signal minus the phase of the emitted signal at any time. The received phase is computed using the quasi-stationary approximation. In other words, the distance travelled by the wave between emission and reception is approximated as twice the current range.

Quantification and sampling We generate 16 bits integer IQ samples (16 bits for the in-phase component and 16 bits for the quadrature component). For each pulse, we start sampling after emission plus a delay corresponding to the maximum unambiguous range (i.e. we wait for the echo of a target located at the maximum unambiguous range). In this way, we skip the echo from the fast downchirp.

We do not simulate the amplitude fluctuation due to the radar equation. We do not introduce noise other than quantification noise. We do not modelize the athmospheric losses (negligible at this distance anyway). The antenna radiation pattern is one if the target is in the field of view else zero.

We generate a data cube of 69232 pulses with 1229 samples per pulse covering 9 seconds. The theoretical range resolution is:

$$\delta R_r = \frac{F_e c}{2KN} = \frac{c}{2B} \approx 25\text{cm} \tag{A.1}$$

Here, c is the speed of light, K is the chirp rate, B is the bandwidth, N is the number of samples and F_e is the sampling rate. The azimuth resolution is:

$$\delta R_a = \frac{\lambda}{4\sin(\frac{\alpha}{2})} \approx 4\text{cm} \tag{A.2}$$

Here, λ is the radar wavelength and α is the azimuth antenna beamwidth.

A.2.2 Focalization

We produce a SAR image using the back-projection algorithm. It is a flexible algorithm in terms of imaging geometries but it is sensitive to motion deviations and relatively slow. We derive the algorithm in detail using techniques introduced by [180, 213]. Our notations are:

M Number of pulses.

N Number of samples per pulse.

T sampling period.

$s_{m,n}$ Samples of dechirped signal, with m the pulse index and n designates the fast time index.

c speed of light.

t_m time of first sample of pulse m.

Δ time offset from the emission of a pulse to the first sample.

K chirp rate.

$\boldsymbol{o}$ trajectory origin.

$\boldsymbol{v}$ trajectory velocity.

$\boldsymbol{p}$ spatial coordinates of a scatterer.

$\hat{\sigma}$ estimated reflectivity.

R range between the antenna and the scatterer.

v_r radial velocity of the antenna toward the scatterer.

We derive the back-projection algorithm as an approximation to the Time Domain Correlation (TDC) algorithm. With the TDC algorithm, the estimated intensity is simply the dot product between the received signal and a predicted signal. This writes as follows.

$$\hat{\sigma}(\boldsymbol{p}) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} s_{m,n} \exp\left(-j\phi(m, nT, \boldsymbol{p})\right) \tag{A.3}$$

with

$$\begin{aligned} \phi(m, t, \boldsymbol{p}) = & \\ & - \frac{4\pi f_c R(t + t_m, \boldsymbol{p})}{c} \\ & + K\pi \left(t + \Delta - \frac{2R(t + t_m, \boldsymbol{p})}{c} \right)^2 \\ & - K\pi \left(t + \Delta \right)^2 \end{aligned} \tag{A.4}$$

and

$$R(t, \boldsymbol{p}) = \|\boldsymbol{o} + \boldsymbol{v}t - \boldsymbol{p}\| \tag{A.5}$$

We may rewrite ϕ as follows.

$$\begin{aligned} \phi(m, t, \boldsymbol{p}) = & \\ & - \frac{4\pi f_c R(t + t_m, \boldsymbol{p})}{c} \\ & - 2K\pi \left(\Delta + t - \frac{R(t + t_m, \boldsymbol{p})}{c} \right) \frac{2R(t + t_m, \boldsymbol{p})}{c} \end{aligned} \tag{A.6}$$

$$\begin{aligned} \phi(m, t, \boldsymbol{p}) = & \\ & - \frac{4\pi f_c R(t + t_m, \boldsymbol{p})}{c} \\ & - 2K\pi \left(\Delta + t \right) \frac{2R(t + t_m, \boldsymbol{p})}{c} \\ & + K\pi \left(\frac{2R(t + t_m, \boldsymbol{p})}{c} \right)^2 \end{aligned} \tag{A.7}$$

In order to derive the back-projection algorithm, we shall make two approximations. The first one is to linearize the distance to the scatterer about some time u.

$$R(t_m + t, \boldsymbol{p}) \approx R(t_m + u, \boldsymbol{p}) - v_r(t_m + u, \boldsymbol{p})(t - u) \tag{A.8}$$

$$R(t_m + t, \boldsymbol{p}) \approx R(t_m + u, \boldsymbol{p}) + v_r(t_m + u, \boldsymbol{p})u - v_r(t_m + u, \boldsymbol{p})t \tag{A.9}$$

We introduce notations A and B.

$$R(t_m + t, \boldsymbol{p}) \approx A(m, \boldsymbol{p}) + tB(m, \boldsymbol{p}) \tag{A.10}$$

$$\begin{aligned} A(m, \boldsymbol{p}) :=& R(t_m + u, \boldsymbol{p}) + v_r(t_m + u, \boldsymbol{p})u \\ B(m, \boldsymbol{p}) :=& - v_r(t_m + u, \boldsymbol{p}) \end{aligned} \tag{A.11}$$

$$\begin{aligned} \phi(m, t, \boldsymbol{p}) = & \\ & - \frac{4\pi f_c}{c}(A(m, \boldsymbol{p}) + tB(m, \boldsymbol{p})) \\ & - \frac{4K\pi\,(\Delta + t)}{c}(A(m, \boldsymbol{p}) + tB(m, \boldsymbol{p})) \\ & + \frac{4K\pi}{c^2}\left(A(m, \boldsymbol{p}) + tB(m, \boldsymbol{p})\right)^2 \end{aligned} \tag{A.12}$$

$$\phi(m, t, \boldsymbol{p}) = C(m, \boldsymbol{p}) + tD(m, \boldsymbol{p}) + t^2 E(m, \boldsymbol{p}) \tag{A.13}$$

$$\begin{aligned} C(m, \boldsymbol{p}) := & \\ & - \frac{4\pi f_c}{c}A(m, \boldsymbol{p}) \\ & - \frac{4K\pi\Delta}{c}A(m, \boldsymbol{p}) \\ & + \frac{4K\pi}{c^2}A(m, \boldsymbol{p})^2 \\ D(m, \boldsymbol{p}) := & \\ & - \frac{4\pi f_c}{c}B(m, \boldsymbol{p}) \\ & - \frac{4K\pi}{c}A(m, \boldsymbol{p}) \\ & - \frac{4K\pi\Delta}{c}B(m, \boldsymbol{p}) \\ & + \frac{8K\pi}{c^2}A(m, \boldsymbol{p})B(m, \boldsymbol{p}) \\ E(m, \boldsymbol{p}) := & \\ & - \frac{4K\pi}{c}B(m, \boldsymbol{p}) \\ & + \frac{4K\pi}{c^2}B(m, \boldsymbol{p})^2 \end{aligned} \tag{A.14}$$

Now, we make the second approximation so that the phase become an affine function of time. The affine function of t that is closest to t^2 in L_2 norm over the time of the pulse

is the following.

$$t^2 \approx Lt - \frac{L^2}{6} \tag{A.15}$$

$$L := NT \tag{A.16}$$

$$\phi(m,t,\boldsymbol{p}) = C(m,\boldsymbol{p}) + tD(m,\boldsymbol{p}) + E(m,\boldsymbol{p})(Lt - \frac{L^2}{6}) \tag{A.17}$$

$$\phi(m,t,\boldsymbol{p}) = C(m,\boldsymbol{p}) + tD(m,\boldsymbol{p}) + E(m,\boldsymbol{p})Lt - E(m,\boldsymbol{p})\frac{L^2}{6} \tag{A.18}$$

$$\phi(m,t,\boldsymbol{p}) = \left(C(m,\boldsymbol{p}) - E(m,\boldsymbol{p})\frac{L^2}{6}\right) + t\left(D(m,\boldsymbol{p}) + E(m,\boldsymbol{p})L\right) \tag{A.19}$$

$$\phi(m,t,\boldsymbol{p}) = F(m,\boldsymbol{p}) + tG(m,\boldsymbol{p}) \tag{A.20}$$

$$\begin{aligned} F(m,\boldsymbol{p}) :=& C(m,\boldsymbol{p}) - E(m,\boldsymbol{p})\frac{L^2}{6} \\ G(m,\boldsymbol{p}) :=& D(m,\boldsymbol{p}) + E(m,\boldsymbol{p})L \end{aligned} \tag{A.21}$$

$$\hat{\sigma}(\boldsymbol{p}) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} s_{m,n} \exp\left(-jF(m,\boldsymbol{p}) - jnTG(m,\boldsymbol{p})\right) \tag{A.22}$$

$$\hat{\sigma}(\boldsymbol{p}) = \sum_{m=0}^{M-1} \exp\left(-jF(m,\boldsymbol{p})\right) \sum_{n=0}^{N-1} s_{m,n} \exp\left(-jnTG(m,\boldsymbol{p})\right) \tag{A.23}$$

We recognize a discrete Fourier transform (DFT).

$$\hat{\sigma}(\boldsymbol{p}) = \sum_{m=0}^{M-1} \exp\left(-jF(m,\boldsymbol{p})\right) S_m\left(\frac{TG(m,\boldsymbol{p})N}{2\pi}\right) \tag{A.24}$$

A SAR image can then be formed with M discrete Fourier transforms of signals of size N and, for each pixel, one monodimensional summation. In our implementation, we interpolate the Fourier transforms linearly.

Note that we can actually write $F(m,\boldsymbol{p})$ and $G(m,\boldsymbol{p})$ in terms of $R(t_m + u,\boldsymbol{p})$ and $v_r(t_m + u,\boldsymbol{p})$ instead of $\boldsymbol{p}$. In other words, we are not required to know the three-dimensional coordinates of a point to estimate its reflectivity. We only need to know its range and radial velocity at the pulses center times. Besides, if we know the time of closest approach and the closest distance of a point as well as the speed of the antenna (supposing an uniform rectilinear motion), we can determine the range and radial velocity of the point at any time using rectangular triangle geometry. We can consequently build a SAR image in the TCA-CD geometry knowing only the speed of the antenna and supposing an uniform rectilinear trajectory.

Figure A.2 represents a SAR image. The horizontal axis represents the time of closest approach and the vertical axis represents the closest distance. Figures A.3 and figures A.4 are a closer look at one impulse response. The width of the correlation peak is 5 cm

in azimuth and 40 cm in range. These are close to the theoretical values. The resolutions reported in [28] are 25 cm in range and 50 cm in azimuth. Our correlation peak is offset by 29 cm in range. We assume that this is due to the approximations made by our focusing algorithm. There is no offset along the azimuth. The accuracy reported in [28] is between 1 cm and 8 cm in range and between 29 cm and 42 cm in azimuth.

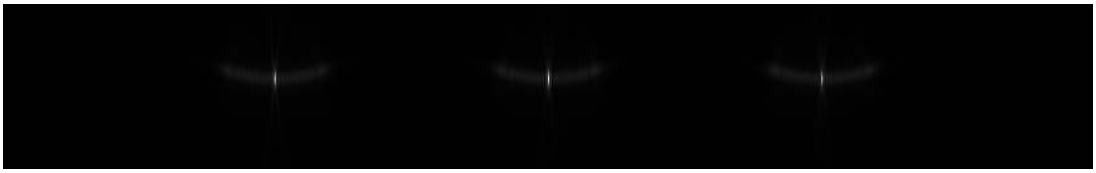


Figure A.2 – SAR image of three scatterers. Horizontal is azimuth and vertical is range.

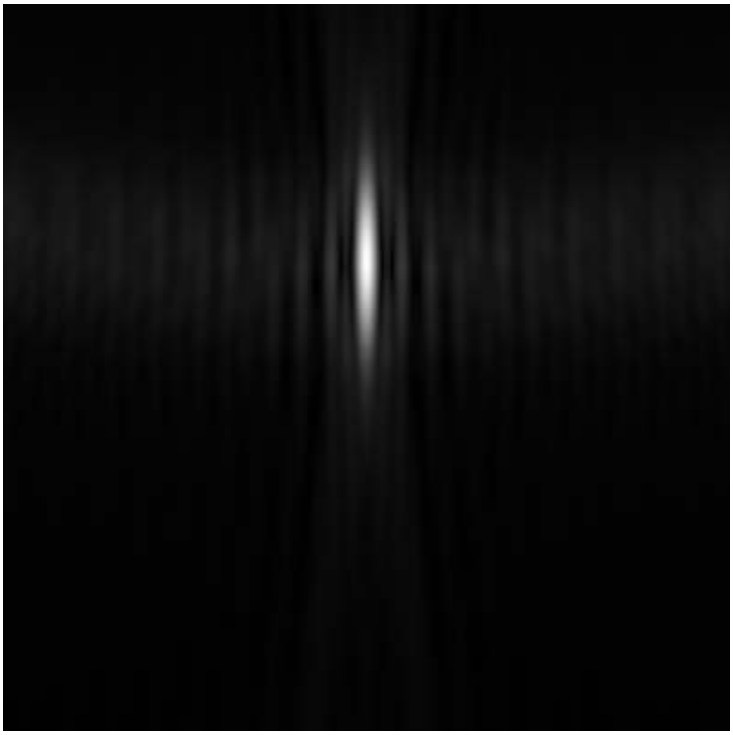


Figure A.3 – Impulse response. Size is two meters by two meters. Horizontal is azimuth. Vectical is range.

A.2.3 Autofocusing

We now suppose that the prior speed estimate is imprecise and we consider a primitive parametric autofocus algorithm based on optimizing an image contrast metric. We plot two image contrast metrics as a function of the speed relative error. We consider these two metrics:

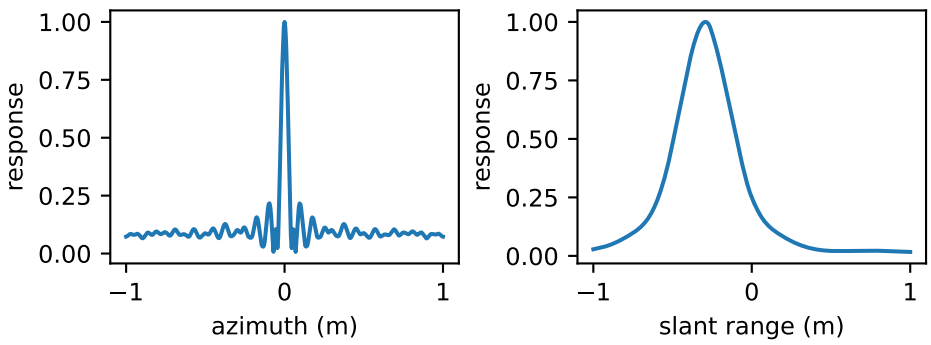


Figure A.4 – Impulse response along azimuth and range.

Maximum amplitude. This is the maximum amplitude of a pixel of the SAR image. The more we manage to gather the energy received from a scatterer to a single point, the higher the energy at that point.

Image entropy. This is the statistical entropy of the amplitudes of the pixels of the SAR image. In other words, the pixel amplitudes are treated as realizations of a random variable.

The plot of the image contrast metrics is represented in figure A.5. We observe that global

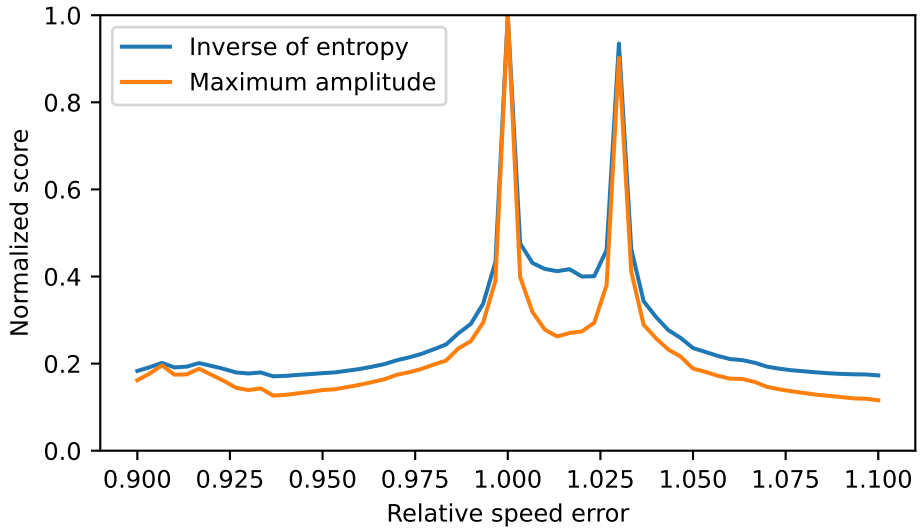


Figure A.5 – Contrast metrics as a function of the relative speed error.

extremum at the right speed. However, there is also a local extremum at an incorrect speed. This illustrate a failure mode of optimization based autofocus algorithms: getting trapped in an incorrect local extremum. Finally, we have observed that the impulse

responses were approximately centered at the correct location despite incorrect speed estimates.

A.3 Conclusion

In this primitive example, we have illustrated how to produce a SAR image despite an incomplete and imprecise estimate of the motion of the antenna. We have justified the assumption that our algorithms could dispose of a SAR image in TCA-CD or IDF-ID geometry. The engineering of an actual imaging system presents challenges, such as obtaining an image formation algorithm whose computation load is admissible, designing a system that handles realistic signals, scene content and mission settings.

Appendix B

Computer vision on SAR images

The algorithms and techniques of computational vision are generally presented in the context of optical images. However, algorithms designed to process optical images may not be suited for processing SAR images. In this chapter, we describe the specifics of SAR images that vision algorithms should account for and we discuss methods that are adapted to SAR images. We refer to paragraph 1.1.3 for an introduction to SAR.

B.1 Intensity statistics

The reflectivity associated with a resolution cell that contains several scatterers depends strongly on the point of view of the antenna relative to the resolution cell, because of the vectorial addition of the contributions of all the scatterers found in that resolution cell [181]. We may model the reflectivity as a random variable. Various families of statistical distributions have been proposed to model the distribution of pixel intensities or amplitude [156, 157, 113, 133, 3, 209, 162]. Some are justified by theoretical models while other are purely empirical. Common families of distributions are reported in table B.1. These models can be fit using maximum likelihood estimators, moments estimators or log-cumulant estimators. As an illustration, several distributions corresponding to the same homogeneous area are plotted in figure B.1.

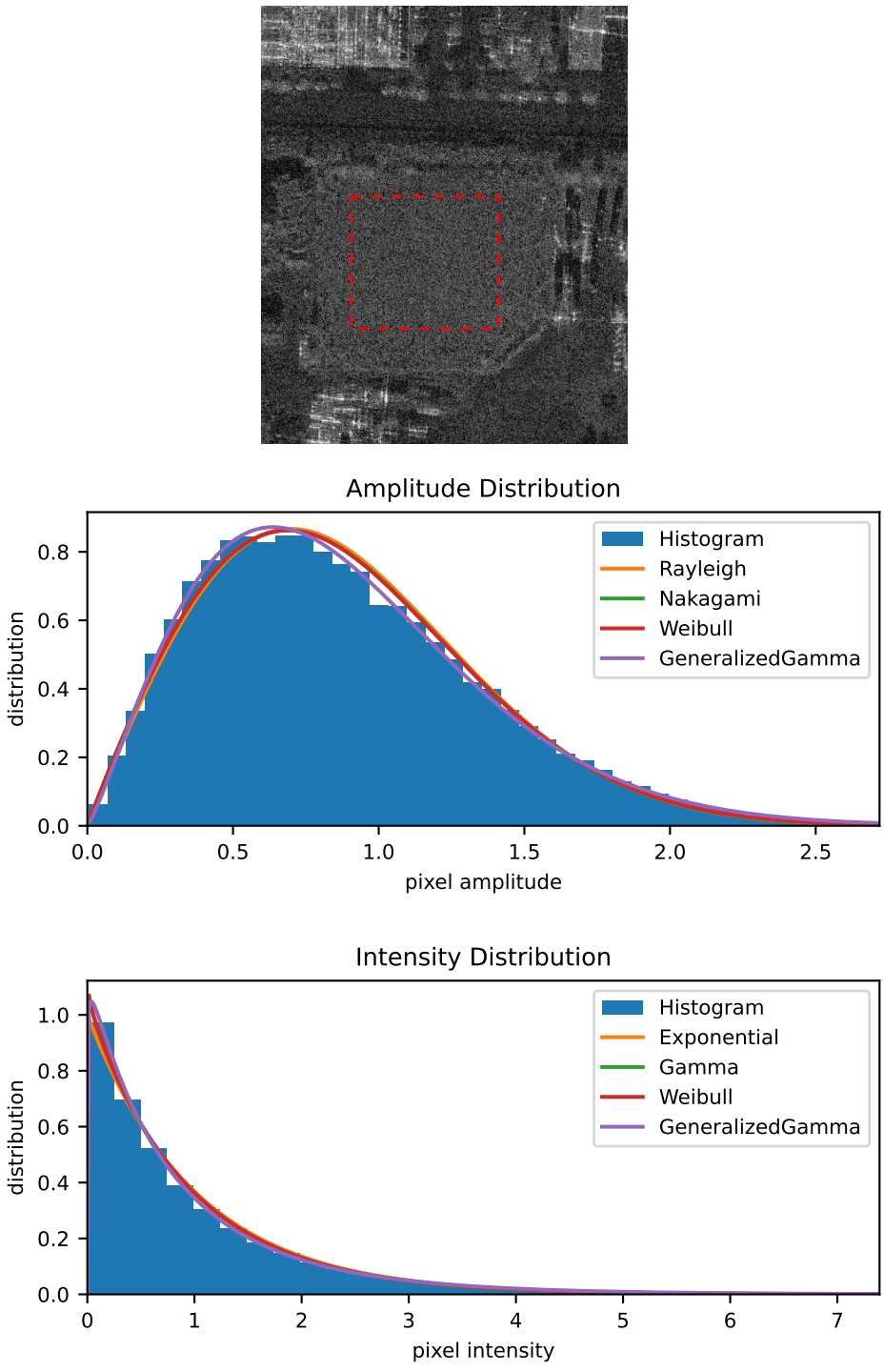


Figure B.1 – An homogeneous area and the distribution of amplitudes and intensities.

Table B.1 – Intensity distribution models

Name	**Density of probability on $\mathbb{R}_+$**	**Quantity**
exponential	$f_\lambda(x) = \lambda \exp(-\lambda x)$	single look intensity
Rayleigh	$f_\sigma(x) = \frac{x}{\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right)$	single look amplitude
gamma	$f_{\mu,L}(x) = \frac{x^{L-1} L^L e^{-\frac{Lx}{\mu}}}{\mu^L \Gamma(L)}$	multi-look intensity
Nakagami	$f_{\lambda,L}(x) = \frac{2(\lambda L)^L x^{2L-1} e^{-\lambda L x^2}}{\Gamma(L)}$	multi-look amplitude
lognormal	$f_{m,\sigma}(x) = \frac{e^{-\frac{(\ln(x)-m)^2}{2\sigma^2}}}{\sigma x \sqrt{2\pi}}$	generic
Weibull	$f_{\eta,\mu}(x) = \frac{\eta x^{\eta-1} e^{-\left(\frac{x}{\mu}\right)^\eta}}{\mu^\eta}$	generic
generalized gamma	$f_{\nu,\kappa,\sigma}(x) = \frac{\lvert\nu\rvert}{\sigma\Gamma(\kappa)} \left(\frac{x}{\sigma}\right)^{\kappa\nu-1} e^{-\left(\frac{x}{\sigma}\right)^\nu}$	generic
Rice	$f_{\mu,\lambda}(x) = \frac{2x}{\mu^2} \exp\left(-\lambda^2 - \frac{x^2}{\mu^2}\right) I_0\left(\frac{2\lambda x}{\mu}\right)$	amplitude of a strong reflector amidst clutter

B.2 Despeckling

B.2.1 Methods

Because of the resolution cells intensity fluctuation, seemingly homogenous areas appear to be corrupted by a multiplicative noise and exhibit a texture called speckle. Speckle can be detrimental to contour and feature extraction. Many algorithms have been proposed to smooth homogeneous areas of SAR images. Some focalization algorithms directly integrate a smoothness prior and produce images with attenuated speckle [267]. Focusing and despeckling can otherwise be decoupled. Algorithms that operate on focused images are numerous and involve various techniques from signal processing.

Adaptative filters

The earliest methods for SAR image despeckling were proposed in the 1980s by Lee, Frost, Kuan and Lopes. Frost [70, 69], Kuan [116, 115] and Lee [127, 124, 123] developped minimum mean square error (MMSE) filters adapted to local image statistics. In order to improve the preservation of edges, Lee proposed two variants of his filter: the sigma filter [125, 128], consisting in averaging similar intensities, and the refined filter [126], consisting in directional selection of the samples used for gathering local statistics. Finally, Kuan and Lopes proposed maximum a posteriori (MAP) filters [115, 144]. These methods are simple but offer poor performances in terms of speckle attenuation of edge preservation. Some introduce artificial discontinuities.

Sparse representation

Some despeckling algorithms consist in filtering a sparse image representation. The rules used to update the sparse representation differ from those in use in the context of gaussian noise and are specific to speckle and rely on MMSE [8] or MAP [207, 206] estimators. Transform-based methods have competitive runtime but may introduce transform-specific artifacts.

Self-similarity

The methods based on self-similarity rely on identifying pixels whose surroundings are similar. Such pixels are expected to have the same ideal intensity. The noise-free intensity of each pixel is estimated from the intensities of the pixels whose surroundings are identified as similar to that of the pixel under consideration. The main challenge is to define a similarity measure between two patches despite their different noise realizations. The use of self-similarity for image denoising was introduced by Buades in [30]. It was extended to SAR images with algorithms such as SPSR [246], PPB [59] and NL-SAR [61].

Block matching 3d

Block matching and 3d filtering (BM3D) was introduced in [52, 51]. It combines ideas from self-similarity and transform-based denoising. The image is divided into overlapping tiles. For each tile, similar patches are identified and are stacked into a three dimensional image. Each three dimensional image is denoised using a transform method and its slices serve as estimates of the corresponding parts of the noise-free image. An adaptation of BM3D to SAR, called SARBM3D, was introduced in [168, 169]. It relies on a patch similarity metric and a wavelet shrinkage adapted to SAR.

Inverse problems

The denoised image is retrieved as the solution of an optimization problem modelling the degradation process and assumptions about noise-free images, generally the sparsity of edges. Among the many algorithms that are reported in the literature, an instance of a variational despeckling algorithm is MIDAL [22]. It combines a log-gamma observation likelihood and a total variation prior. The numerical optimization is carried out using the alternating direction method of multipliers (ADMM) algorithm.

In the framework of plug-and-play ADMM [38], the proximal operator of the regularization may be replaced by any gaussian denoiser (provided that it verifies some mild conditions). This allows for the use of a gaussian denoiser to remove other kind of noises. An instance of plug-and-play ADMM applied to SAR image despeckling is MuLoG [60].

Variational methods may provide despeckled images of high visual quality but are biased toward the prior and induce heavy computational loads.

Supervised learning

Learning to denoise optical images was introduced in [102, 31] and [262] and was extended to SAR images in [43] (SAR-CNN), [229] (ID-CNN), [263] (SAR-DRN), [119] and [158] (deSpeckNet).

The main challenge with a supervised learning approach is the unavailability of speckle-free SAR images. Several coping strategies are documented in the literature. First, optical images can be speckled synthetically to serve as training data. However, this approach suffers from a modality mismatch as the synthetic speckle may not prove sufficiently realistic and as the statistics of optical images differ from those of real SAR images. A second strategy relies on using a stack of SAR images of the same scene at different times. Their temporal average can be used as the target despeckled image. However, acquiring many images of the same scene with the same radar trajectory can be very challenging or unavailable.

In order to reduce the amount of data required for training, the following strategies were investigated in the literature :

— Injecting prior knowledge into the loss function. Such knowledge may be the total variation or the Kullback-Leiber divergence with respect to some known noise distribution [225].
— Injecting prior knowledge into the network architecture. Such knowledge may be a decomposition such as contourlets [139].
— Samples generation with generative adversarial learning [228].
— Learning by transfer. The network can be pretrained on synthetic data and then refined using real but less abundant temporally-multilooked SAR images.

Various network architectures were reported in the literature (CNN, U-Net). In a residual learning fashion, most models predict the noise value instead of directly predicting the despeckled image. With the exception of ID-CNN [229], most models operate in log-intensity format.

Unsupervised learning

It was observed in [131] that a network could be trained to denoise even in the absence of example noise-free images. Instead, it could be trained using only pairs of images of the same scenes with different noise realizations, in a strategy called noise2noise. This idea was extended by noise2void [110, 117]. Here, a neural network is trained to predict the probability distribution of the intensity of a pixel from the noise corrupted intensities of its surrounding. Noise2noise and noise2void were adapted to the specifics of SAR images in [56, 154, 153] respectively.

Deep prior

It was discovered in [223, 141] that some neural network architectures had regularizing properties which could be leveraged to perform computer vision tasks without any kind

of training. If the weights of some neural networks are optimized so that the network outputs a given image out of some fixed noise realization, it is observed that, in the course of optimization, a denoised image will be produced first before the given image is faithfully reproduced. Deep prior denoising is no practical algorithm. It suffers from a radically uneconomical use of computational ressources, a poor explicability, the presence of unpredictible artifacts and the lack of a theoretical framework.

B.2.2 Comparison

We have implemented and compared the following algorithms:

Frost The adaptive MMSE filter from Frost [69].

MMSE The adaptive MMSE filter from Kuan [115].

MAP The adaptive MAP filter from Lopes [144].

PPB The self-similarity method from Deladalle [59]. Our implementation makes use of integral images as defined in [57].

WaveletMMSE The transform-based method from Argenti [8]. We have used Haar wavelets. Despite their simplicity, they achieved the most satisfying visual quality.

SARBM3D The method defined in [169] which combines self-similarity and sparsity.

MIDAL The variational method defined in [22]. We have used the total variation defined in [216] instead of the one from the original article.

BlindspotNet The method based on unsupervised learning and defined in [153].

The output of these algorithms applied to different Miranda SAR images can be seen in figures B.2, B.3 and B.4. The computation times are reported in table B.2. The computer used for these measurements was composed of an *Intel Core i7-10700K* CPU and an *NVIDIA GeForce RTX 3090* GPU. We also report in table B.2 the equivalent number of looks (ENL) corresponding to an homogeneous area (water).

$$\mathrm{ENL} = \frac{\mathbb{E}[X]}{\mathrm{Var}(X)} \tag{B.1}$$

where X is a pixel's denoised intensity. Larger ENL correspond to smoother images. ENL should be computed over homogeneous areas such as water.

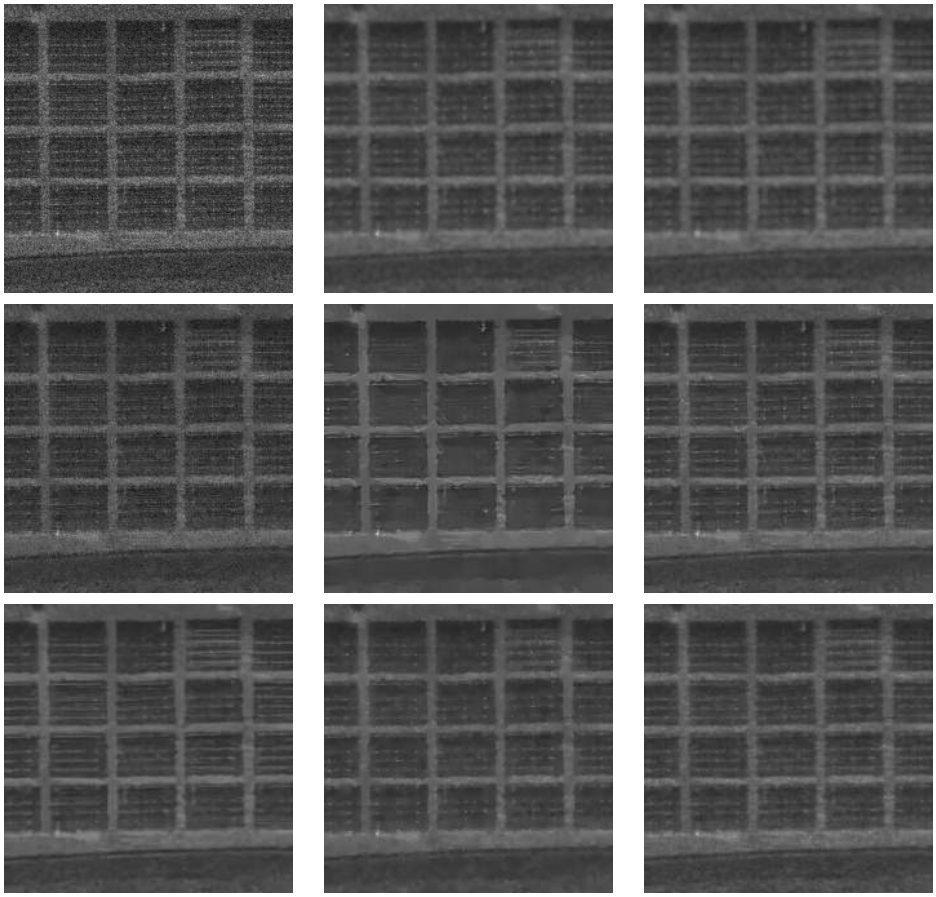


Figure B.2 – Denoised images of a solar panels. Algorithms: Original, Frost, MMSE, MAP, PPB, WaveletMMSE, SARBM3D, MIDAL, BlindspotNet.

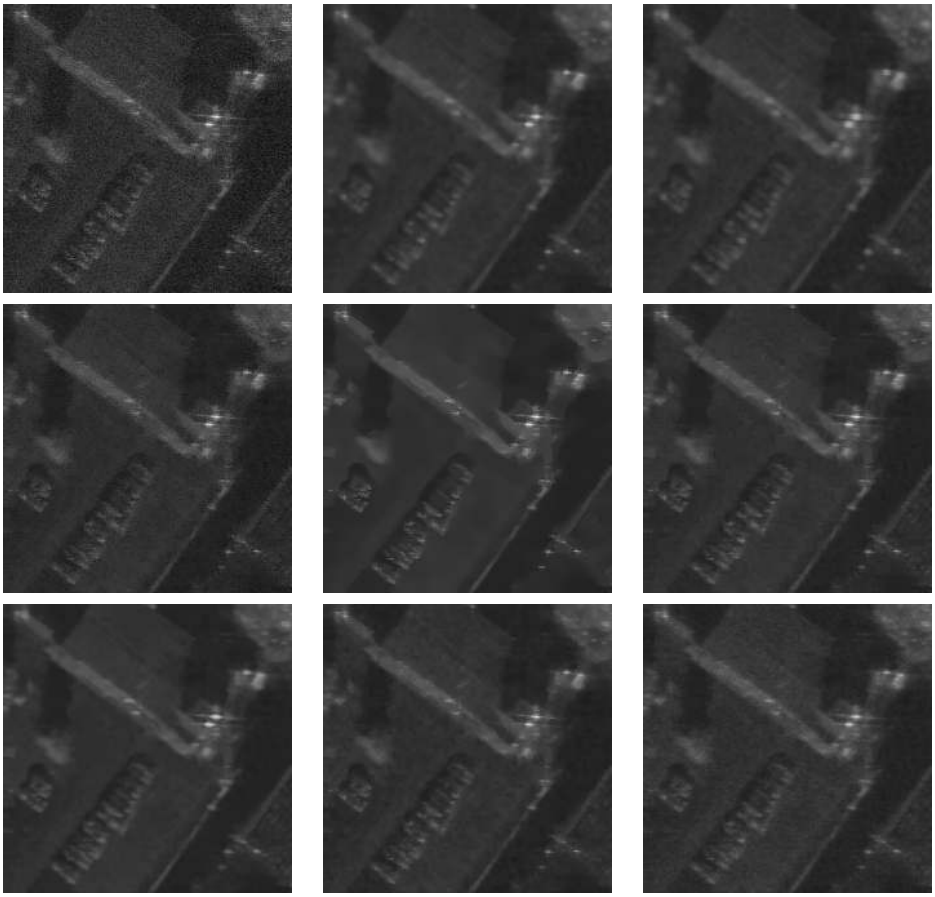


Figure B.3 – Denoised images of an urban area. Algorithms: Original, Frost, MMSE, MAP, PPB, WaveletMMSE, SARBM3D, MIDAL, BlindspotNet.

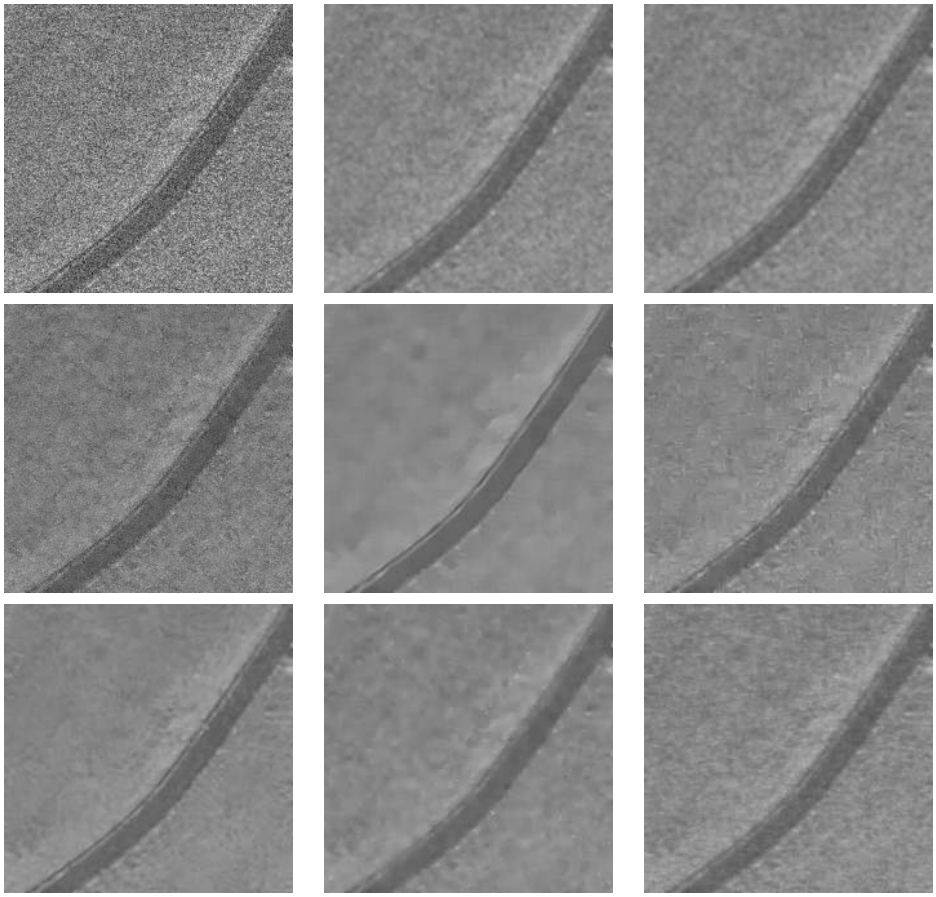


Figure B.4 – Denoised images of a road. Algorithms: Original, Frost, MMSE, MAP, PPB, WaveletMMSE, SARBM3D, MIDAL, BlindspotNet.

Table B.2 – Runtime for denoising

Algorithm	**Runtime (ms)**	**ENL (water)**
Original	N/A	6.1
Frost	54	139.5
MMSE	1	106.3
MAP	2	29.2
PPB	823	585.3
WaveletMMSE	437	152.4
SARBM3D	21458	408.5
MIDAL	875	190.1
BlindspotNet	19	100.2

B.3 Contour detection

Some image features and other vision algorithms are based on contours, which are the lines of discontinuity of an image. Extracting contours from an optical images is typically done with finite-differences, such as Sobel filtering, possibly composed with an edge thinning algorithm such as non-maxima suppression [34]. Using finite differences for contour detection is computationally efficient but insufficiently robust to speckle so other techniques more appropriate for SAR images were proposed.

B.3.1 Ratio of averages

One such contour detector that is adapted to SAR is based on the ratio of averages [24, 261, 73, 104, 10, 11, 140, 257, 199]. The complete pipeline is depicted in figure B.6. For each candidate edge orientation, the image is fed to two convolutional filters such as those represented in figure B.5 and defined in [199]. Each filter estimates the mean intensity on either side of a location at some orientation. For each pair of filtered images, for each pixel, we compute the maximum ratio of the left and right mean intensities:

$$u_{ij}^{\theta} = 1 - \frac{\min\left\{l_{ij}^{\theta}, r_{ij}^{\theta}\right\}}{\max\left\{l_{ij}^{\theta}, r_{ij}^{\theta}\right\}} \tag{B.2}$$

where l_{ij}^{θ} and r_{ij}^{θ} are respectively the left and the right filtered images and u_{ij}^{θ} is the edge map for orientation θ. Finally, the edge maps of all orientations are aggregated to retrieve the edge response and the edge orientation maps:

$$v_{ij} = \max_{\theta} u_{ij}^{\theta} \tag{B.3}$$

$$\alpha_{ij} = \arg\max_{\theta} u_{ij}^{\theta} \tag{B.4}$$

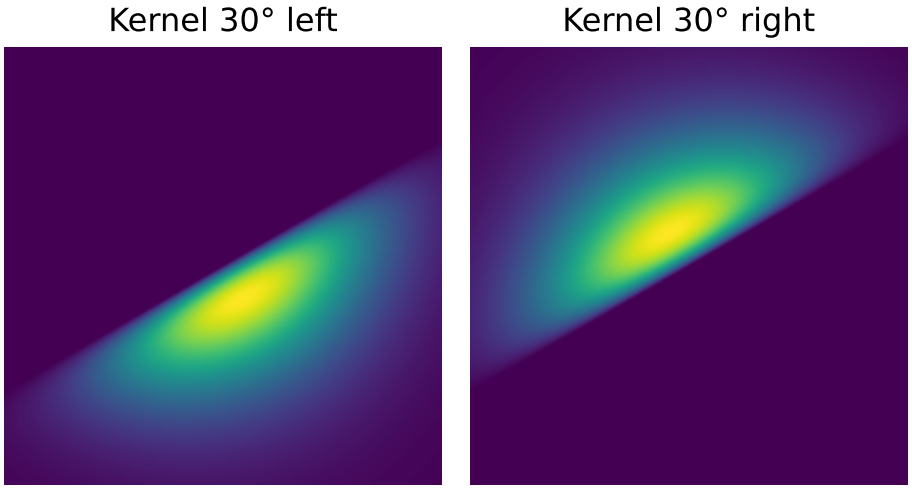


Figure B.5 – A pair of convolution kernels for contour detection at 30°.

The edges can further be thinned with non-maxima suppression, as introduced by Canny [34].

The computational load associated with this contour detection algorithm is much higher that of finite differences, as the number of convolutions required is twice the number of possible orientations instead of the constant two. This can be alleviated by considering only two orthogonal orientations and retrieving the edge orientation as with finite differences, as proposed by Dellinger in [62].

Results of contour detection based on the ratio of averages are illustrated in figure B.7. The results were obtained with a pipeline combining SARBM3D denoising, ratio-of-averages contour detection and non-maxima suppression edge thinning.

B.3.2 Phase congruency

Kovesi introduced a contrast invariant contour detection algorithm, for images with gaussian noise, based on phase congruency [107, 108, 109]. It relies on the observation that all the Fourier components of a step signal are in phase at a point of discontinuity. In order to recognize such points, we perform a local Fourier transform, using Gabor or log-Gabor filters, and we compute the ratio between the magnitude of the sum of the frequency responses and the sum of the magnitudes of the frequency responses. If the frequency responses are in phase, then the ratio between the magnitude of the sum and the sum of the magnitude approaches one. Actually, Kovesi proposed modified definitions of the phase congruency in order to improve the robustness to noise and to reject constant signals (whose frequency components are also in-phase). Let us denote M_n^{even} and M_n^{odd} the real and imaginary components of the log-Gabor convolution kernel for some frequency

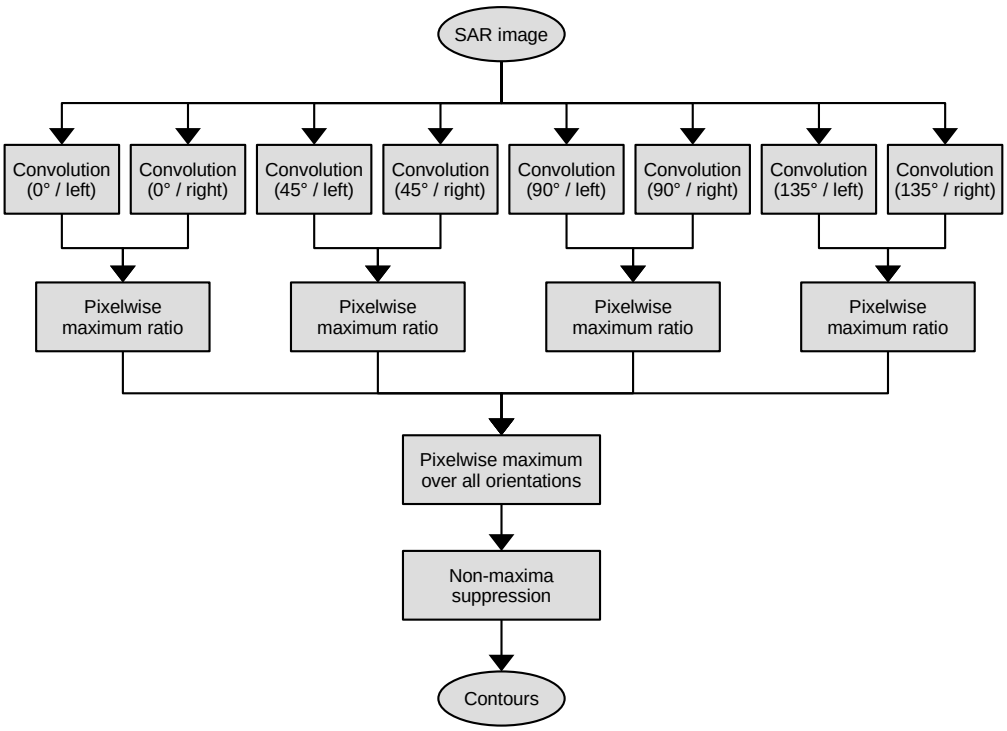


Figure B.6 – Contour detection pipeline (for 4 orientations).

identified by n. Let us denote I an input image. The responses to real and imaginary log-Gabor filters are:

$$\begin{cases} e_n(x) = I * M_n^{even}(x) \\ o_n(x) = I * M_n^{odd}(x) \end{cases} \tag{B.5}$$

The magnitude of the response to frequency n at x is:

$$A_n(x) = \sum_n \sqrt{e_n(x)^2 + o_n(x)^2} \tag{B.6}$$

The energy at x is:

$$E(x) = \sqrt{\left(\sum_n e_n(x)\right)^2 + \left(\sum_n o_n(x)\right)^2} \tag{B.7}$$

Using these notations, one of the monodimensional phase congruency definitions proposed by Kovesi is:

$$PC(x) = \frac{\lfloor E(x) - T \rfloor}{\sum_n A_n(x) + \epsilon} \tag{B.8}$$

Here, T and ϵ are parameters that depend on the noise standard deviation, index n identifies frequency components. Instances of monodimensional log-Gabor filters are rep-

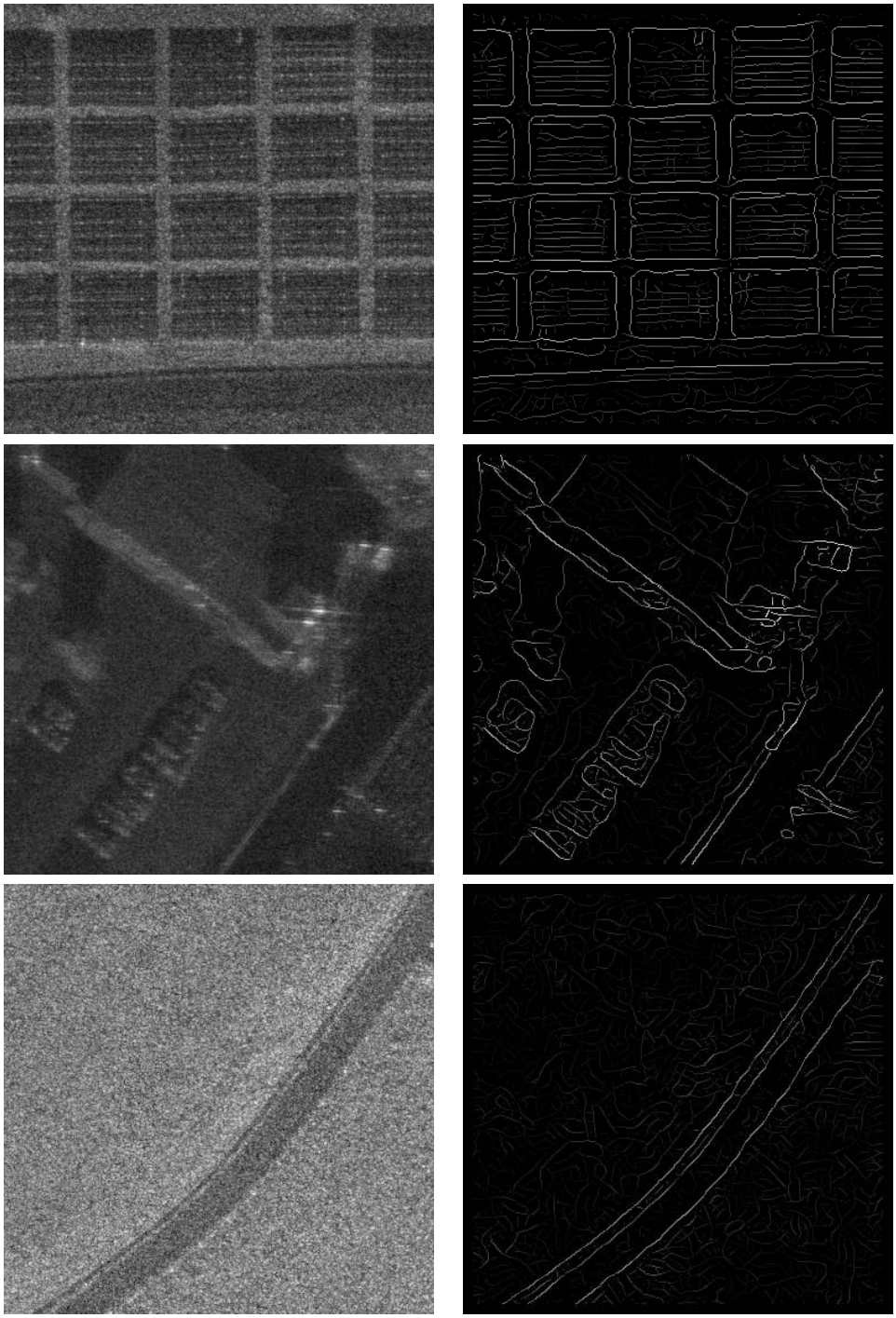


Figure B.7 – Contour detection by ratio of averages and suppression of non maxima.

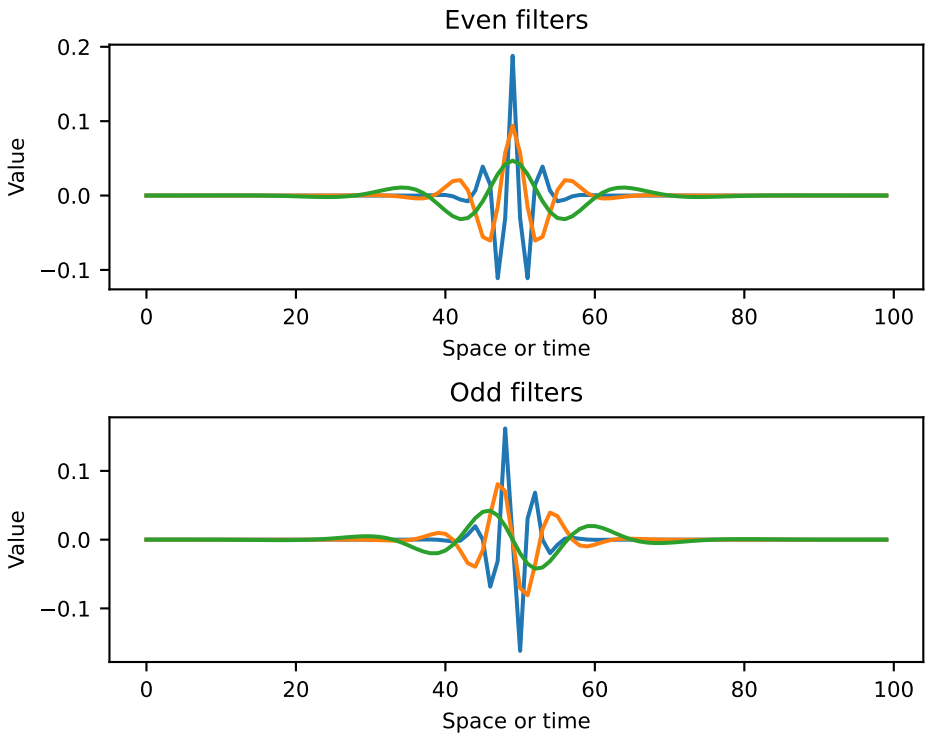


Figure B.8 – Monodimensional log-Gabor filters. Top: real part. Bottom: imaginary part.

resented in figure B.8. An example of phase congruency edge response is represented in figure B.9.

One way of extending phase congruency from monodimensional to bidimensional signals consists in computing monodimensional phase congruency with bidimensional log-Gabor filters of various orientations and to take the maximum phase congruency response over the different orientations. An instance of a bidimensional log-Gabor filter is represented in figure B.10. Instances of contour detections based on phase congruency are depicted in figure B.11. The images were denoised with a local linear minimum mean square error filter. The contours were not thinned.

An adaptation of phase congruency to SAR was proposed in [241]. It relies with estimating the frequency response using a technique inspired from the ratio of averages.

B.3.3 Other methods

The detection of linear structures in particular has attracted much research. The methods reported in the literature rely on region growing and statistical testing such as LSD-SAR [76, 138], Hough transform and conditional random fields [112] and generalized likelihood ratio test [75].

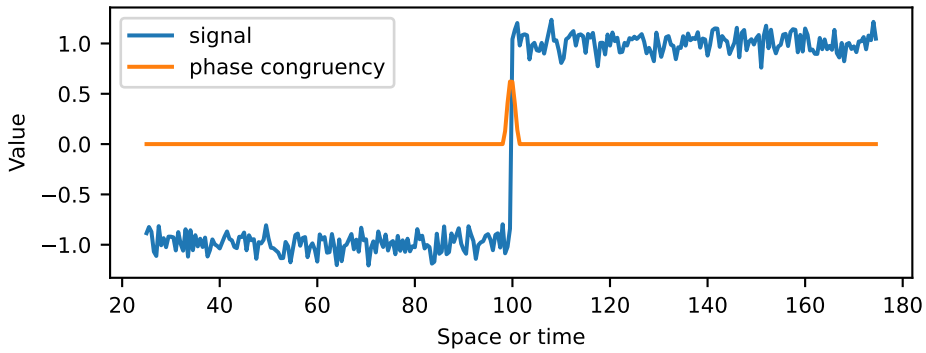


Figure B.9 – Monodimensional log-Gabor edge response.

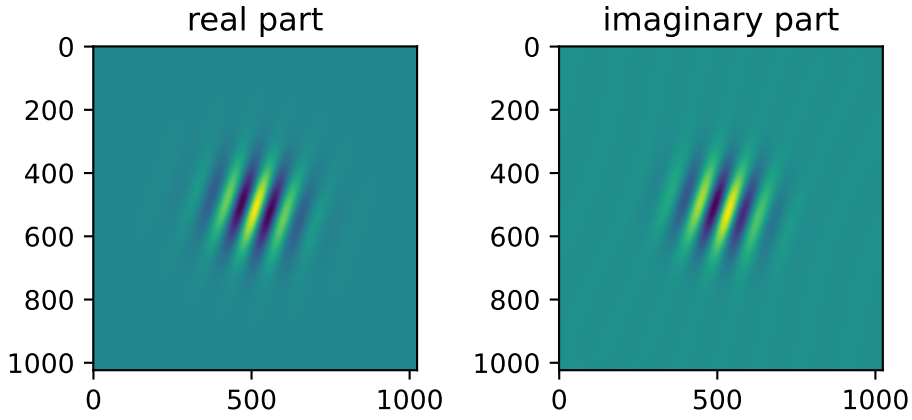


Figure B.10 – Instance of a two-dimensional log-Gabor filter.

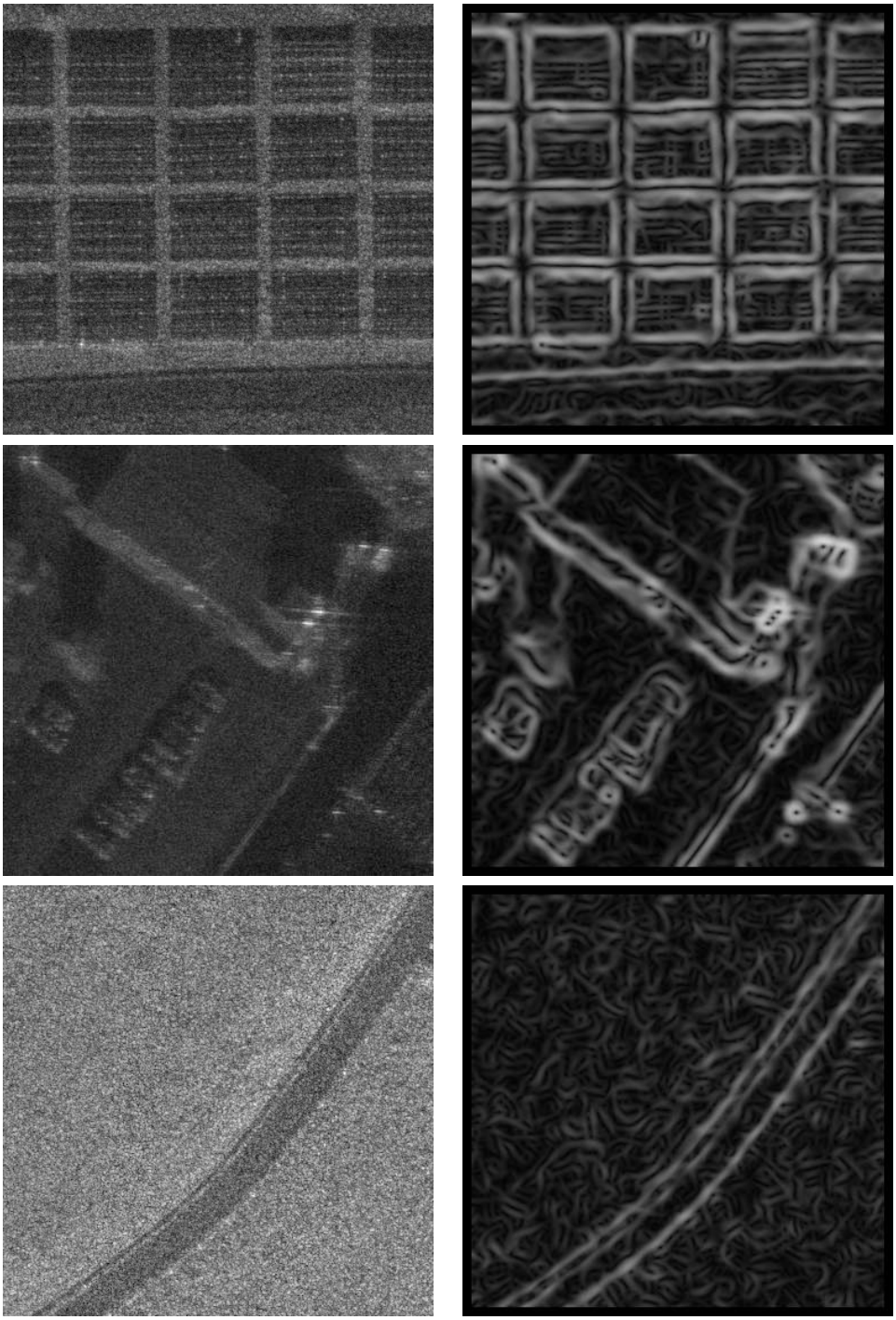


Figure B.11 – Contour detection by phase congruency.

B.4 Segmentation

Segmentation is the task of subdividing an image into distinct regions based on some homogeneity criterion related to intensity, texture or semantics. Its result is the assignment of a class to every pixel.

B.4.1 Model-based segmentation

The various handcrafted segmentation algorithms developed for mainstream computer vision have been adapted to SAR images. They include Simple Linear Iterative Clustering (SLIC) [1, 269, 268] Mumford-Shah, active contours and level sets [256, 91, 177, 89], markov random fields [238, 171, 118], watersheds [230, 182], graph cuts [97, 25, 259], cuts in hierarchical representations such as the binary partition tree [188, 5, 4, 187], and region merging [72].

An example of model-based segmentation is given in figure B.12. The image is first denoised with self-similarity (PPB), a binary partition tree representation of the denoised image is constructed and the tree is cut so as to keep only the last few regions (three regions for the image that shows a road and 25 regions for the image that shows solar panels).

B.4.2 Segmentation based on handcrafted features

Semantic or texture segmentation of SAR images can be accomplished by extracting and classifying handcrafted dense features. Some well-known features include:

Features from intensity cooccurence One of the oldest features are gray-level cooccurence matrices (GLCM) [84, 45, 48, 46, 47]. They are joint histograms, over local neighborhoods, of the pairs of intensity values of the pairs of pixels that are separated by a given step in space.

Features based on intensity comparisons Some features describe a local neighborhood by comparing the intensities of specified pairs of pixels in this neighborhood. Among these are random ferns [165], BRIEF [33], ORB [186] and Local Binary Patterns (LBP) [164]. LBP performs comparisons between a central pixel and neighboring pixels to produce a binary description which is made invariant to rotation by a bit-rotation that minimizes the number represented by the sequence of binary digits representing the set of comparisons. Then, local histograms of such normalized binary patterns are produced and constitute local features.

Features from contour orientation Some features are based on local histograms of gradient orientation, such as HOG [53], SIFT [145] and its adaptation to SAR [62] and SURF [16], which approximates the gradient using integral images and box filters. In adapting these features to SAR images, the contours must be estimated in a manner which is robust to speckle, for instance using the method described in section B.3.

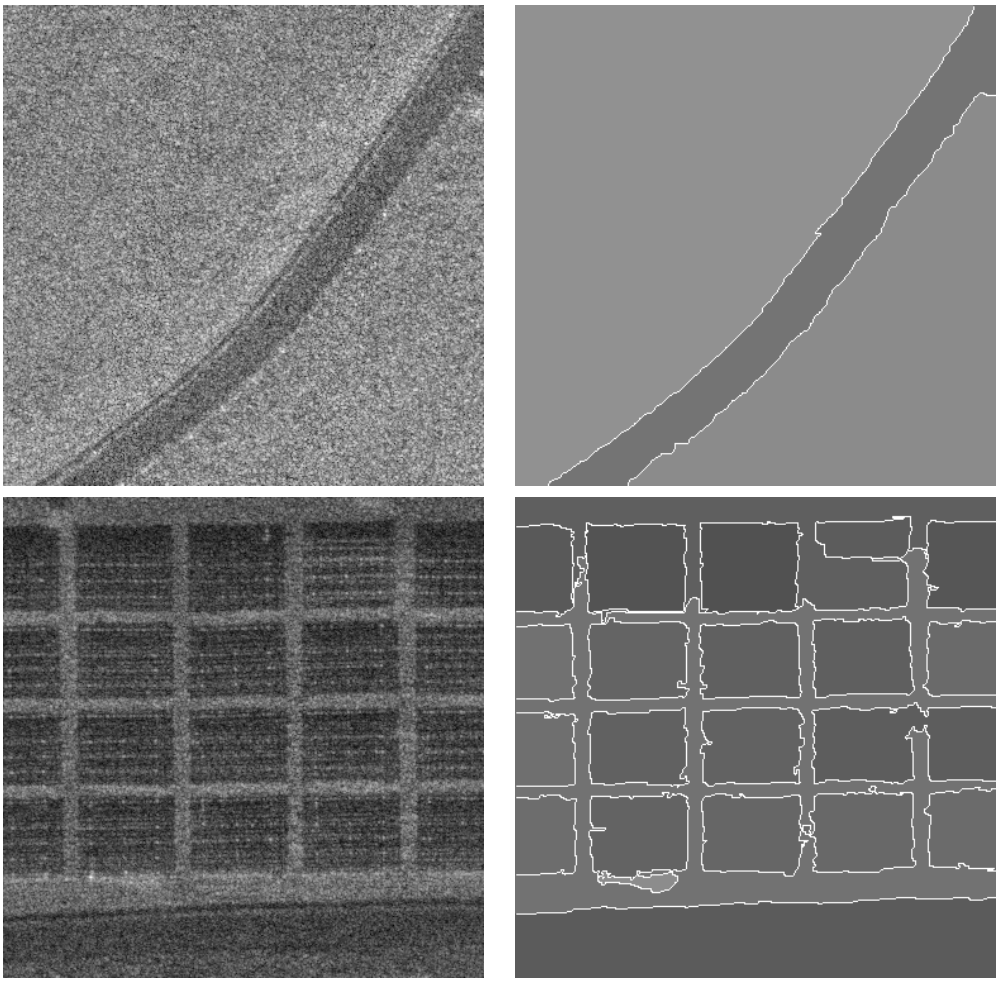


Figure B.12 – Intensity-based segmentation into homogeneous regions with self-similarity denoising and the binary partition tree.

Features from linear filtering The responses to Gabor filters at various scales and orientations can be used as features for instance. As well, wavelet-like decompositions such as curvelets and contourlets [143] can be used as features. [194]

Features from morphological filtering Various kinds of morphological filters were applied to the extraction of features from remote sensing images. Early morphological feature extraction was based on openings and closings with structural elements of various shapes, orientations and sizes. Researchers also proposed extracting features using filters by reconstruction, among which are attribute filters [27, 190]. Their use for feature extraction was introduced in [173, 159, 55, 18] and extended in various ways in [170, 63, 175]. The efficient calculation of attribute filters relies on a hierarchical image representation call the component tree [103, 208]. This has spurred research on extracting features from other hierarchical image representations [174] such as the tree of shapes [37, 155, 12, 54] and partitioning trees [23, 19]. The use of attribute profiles for segmenting SAR images is reported in [219].

An example of semantic segmentation based on handcrafted features is given in figure B.13. The image is denoised with PPB, contours are extracted with the ratio of averages and local histogram of contours are computed and serve as features to discriminate the presence or absence of forest.

B.4.3 Segmentation based on deep learning

Deep learning was successfully applied to the semantic segmentation of SAR images. Although many network architectures are described in the general technical literature (e.g. FCN, U-Net, DeepLab, SegNet, ParseNet, PSPNet, DeconvNet), the literature specialized on SAR mainly reports the use of the U-Net architecture [183] or its derivatives. Detection or rivers, roads, buildings and water was notably reported in the literature [88, 231, 264, 212].

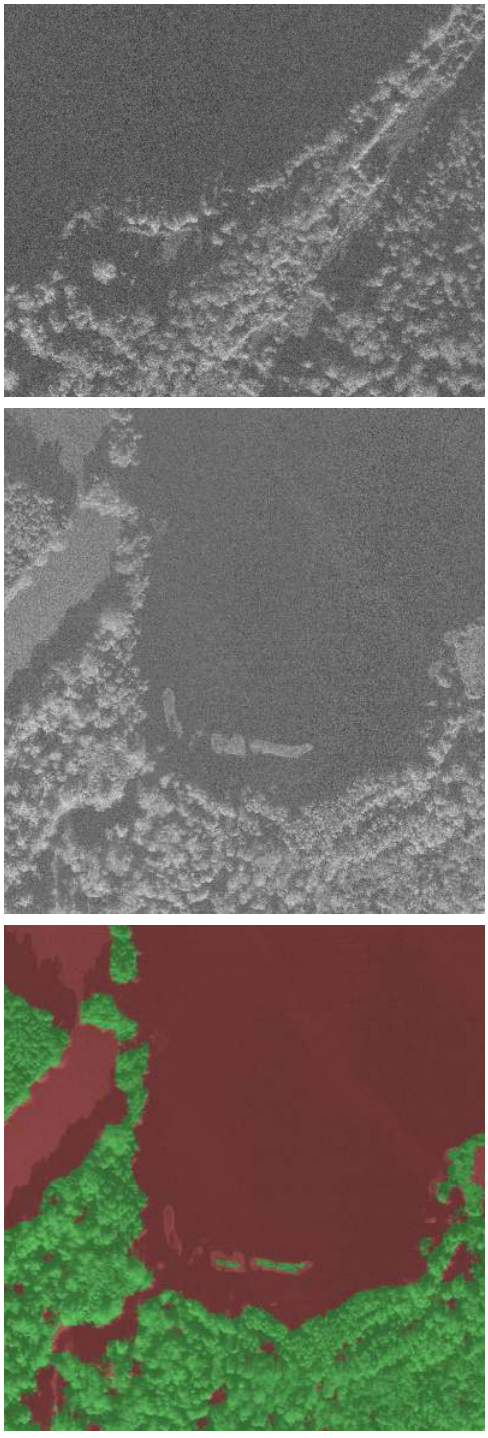


Figure B.13 – Semantic segmentation into forest / no forest using handcrafted features. Top down: training image, test image, segmentation result.

B.5 Object detection

We refer to object detection as the task of predicting, from a SAR image, bounding boxes and the classes of objects that they contain.

Constant False Alarm Rate (CFAR) detection algorithms are a popular way of detecting targets in radar signals and SAR images. Precisely, they detect pixel blobs whose intensity is statistically anomalously high. They consist in estimating the background intensity distribution and identifying pixel blobs whose intensity distribution is not predicted by the estimated background radiometric model. This is typically performed on a windows, as illustrated in figure B.14. The center of the window is the test region, the intermediate ring is an unused buffer region and the outer ring serves for background intensity estimation. The classes of the predicted windows can for example be infered using deep neural networks or handcrafted features. Contour features, such as moment features, or texture features, such as those obtained from dimensionality reduction, are popular for this purpose.

Deep objection detection algorithms such as YOLO [178, 39], SSD [142] and R-CNN [66, 179] were applied to the detection tasks where the background is relatively simple, such as detecting ships in the ocean, aircrafts on a runway or military vehicle in open field.

Finally, due to the relative rarety of SAR images, the poor generalization from other imaging modalities, the computational expense of simulating complex scenes, handcrafted algorithms remain prevalent for some tasks [247, 146, 147, 189].

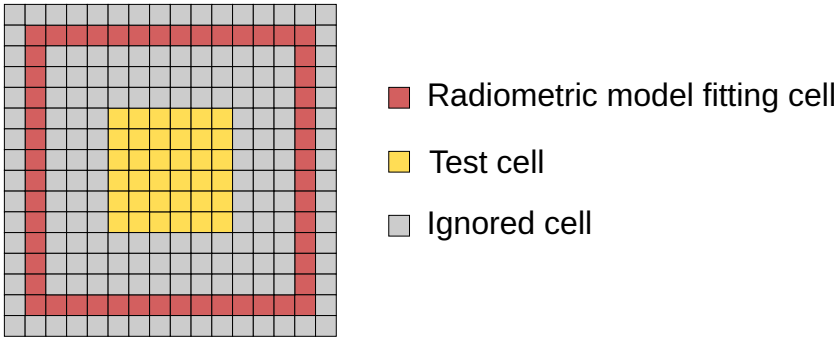


Figure B.14 – CFAR detection cells.

B.6 Popular datasets

Some popular datasets of SAR images include:

MSTAR This is a set of X-band SAR images of 13 different models of military vehicles seen under various angles [184]. It was produced in the mid 1990s and can be used as a benchmark for Automatic Target Recognition (ATR).

SARptical This is a set of 10,000 pairs of TerraSAR-X high resolution spotlight SAR images and UltraCAM aerial optical images [232].

SEN1-2 This is a set of 282,384 pairs of Sentinel-1 (SAR) and Sentinel-2 (optical) image patches [192, 191].

OpenSARShip This is a set of 41 Sentinel-1 (SAR) images annotated with 11,346 ships. It is intended as a benchmark for Automatic Target Detection (ATD) [90, 132].

UAVSAR The National Aeronautics and Space Administration (NASA) Jet Propulsion Laboratory (JPL) provides a set of nearly 20 000 airborne Ka, L and P band SAR images at various processing levels [1].

***Umbra* and *Capella Space* Open Data** These two private companies have released to the public large datasets of SAR images [2] [3].

1. UAVSAR `https://uavsar.jpl.nasa.gov/`
2. Capella Space open data `https://registry.opendata.aws/capella_opendata/`
3. Umbra open data `https://registry.opendata.aws/umbra-open-data/`

Appendix C

Object detection using a cascade of regressions

SAR images exhibit artefacts such as speckle, electromagnetic shades, layover, blurring and strong reflections. SAR images are often rare because their acquisition is expensive, requires technical expertise and relies on sophisticated devices. The generalization between different sensors or imaging modes is poor. These factors complicate the design of efficient object detection algorithms for SAR images. There exists a large body of work concerning the detection of ships on the surface of the sea from SAR images. But this detection problem is comparatively easy since there is little clutter on the sea.

In this chapter, we propose a new Automatic Target Detection (ATD) algorithm for SAR images. We describe this algorithm from a generic point of view, without referring to a particular kind of object to be detected. Afterward, we describe how to instanciate this algorithm for the detection of buildings in particular. Finally, we report our experimental results.

C.1 Generic detection algorithm

Our algorithm is inspired from Cascaded Pose Regression (CPR) [65]. CPR fits geometric shape models to images. The parameters of the shape are required to form a mathematical group. CPR produces a sequence of shapes and a sequence of local features. Given a current shape, the current local features are extracted from the image, relative to the current shape, and the next shape is inferred from the current local features, using some regression model. There are as many regression models as there are iterations and each one must be trained. We refer to the original article for more details about CPR. CPR was for instance successfully applied to the shape or pose estimation of the face [236, 172], the hand [214] and human organs [9].

Our detection algorithm consists in (1) spilling many marker points uniformly over the image, (2) displacing each marker according a learned dynamic that draws each marker towards the target present in its perception field, (3) classifying each marker as having reached a target or not and (4) coalesce clusters of markers. The algorithm is depicted in figure C.1 and is detailed hereafter.

1. **Markers initialization.** We distribute N markers uniformly over the image according to stratified sampling. In other words, the image is divided into cells and a

marker is set in each cell at a random location within the cell. The initial location of marker i is noted as $x_{i,0}$ with $1 \leq i \leq N$.

2. **Markers displacement.** We perform K displacements for all markers. The location of marker i at step j is denoted by $x_{i,j}$ with $1 \leq j \leq K$. The displacements depend on local image features collected at locations that are relative to the marker being moved. The dynamic is $x_{i,j} = x_{i,j-1} + \Phi_j\left(h_j(I, x_{i,j-1})\right)$ with I the image, h_j the feature extraction function and Φ_j the displacement function. The features are covariant with the image, which is to say that $\forall \delta \in \mathbb{R}^2$ we have $h_j(I_\delta, x + \delta) = h_j(I, x)$ with I_δ the image I translated by δ. The displacement function would typically be regression model (i.e. regression forest) learned from training data.

3. **Markers classification.** Some markers may not have reached a target object because none was ever present in its perception field for instance. So we need to classify each marker as having reached a target or not. The classification of each marker is denoted as $c_i = \Psi(h_{K+1}(x_{i,K}))$ with Ψ a classifier that would typically be learned from training data.

4. **Markers clustering.** At this point of the algorithm, several positive markers may have reached the same target. In order to return a single detection for each target object, we aggregate the responses with unsupervised clustering. Any clustering algorithm would suit. The one that we uses consists in defining a graph whose vertices correspond to the positive markers and whose edges indicate that the distance between the markers is closer than some threshold. The barycenter of each connected component of this graph is considered as a detection.

The output of this algorithm is a set of points thought to be located on object instances. The algorithm could be slightly modified to produce bounding boxes instead of points. The algorithm relies on K regression models and one classification model. These models can be trained from SAR image annotated with a set of points located on target instances. The regression model of displacement j, with $1 \leq i \leq K$, is trained by initializing markers, performing the displacements from 1 to $j - 1$ and matching each marker with its nearest annotation point. Markers whose nearest annotation point lies beyond the marker's perception field are ignored for training. Likewise, the classification model is trained by initializing markers, performing all the displacements and matching each marker with its nearest annotation point. Markers that are close enough to their respective annotation points are classified as positive. Others are classified as negative.

The parameters of the proposed detection algorithm are the number of stages, the cell size for markers initialization, the regression models, the classification model, the feature extractors, the radius for regression matching, the radius for classification matching and the radius for clustering.

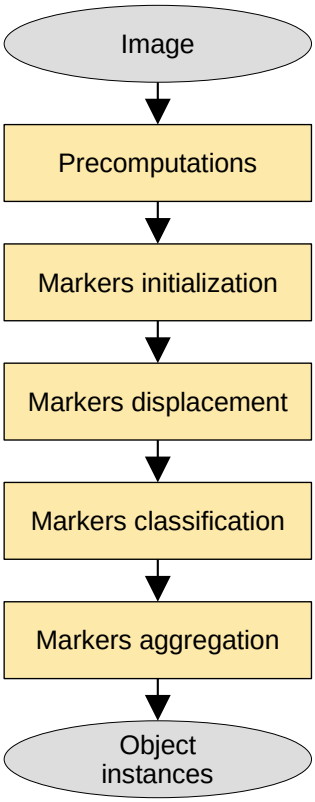


Figure C.1 – Generic detection algorithm.

C.2 Detecting buildings

We have presented a generic object detection algorithm. We now describe how to use it for detecting buildings in SAR images. We have to specify the feature extractors, the regression models and the classification model. We assume that the geometry of the input SAR image is ground or pseudo-ground.

Features In a SAR image, a building visually appears as a strong reflector followed by an electromagnetic shade of a characteristic angular shape. The local features that we employ are accordingly composed of local histograms of contour orientation and of a low-resolution image patch in the vicinity of the marker.

The local features are defined as follows. First, in a preprocessing step, the SAR image is denoised using Kuan's adaptive Minimum Mean Square Error (MMSE) filter [116, 115] and is resampled to a given resolution and orientation of the electromagnetic shades. This normalization improves the generalization between images of different resolutions and between images with different orientations of the electromagnetic shades. The normalized image is fed separately to a contour detector and to a low-pass filter. The contour detection is based on the ratio-of-averages as presented in appendix B.

The local features are extracted from the low-passed image and from the contour map. Given a point P at which to extract local features, we cut a 30 m $\times$ 30 m patch arround P, we subdivide the patch in a 7×7 grid. In each cell of the grid, we compute an histogram of contour orientation and we compute the mean intensity (the image was low-passed so that we just have to read the pixel at the center of the cell). The dimension of the local features accordingly is $7 \times 7 \times (1 + 12) = 637$.

Regression models The regression models map visual features to displacements. They are non-linear perceptrons made of an affine layer, a batch normalization, a sigmoid non-linearity and an affine layer. The dimension of the intermediate layer is 16. They are trained using the L2 loss.

Classification model The classification model maps visual features to detection scores. It is a non-linear perceptron with the same architecture as the regression models. It is trained using the cross entropy loss.

Note that we use the same feature extractor for all steps. We set the number of steps to five and the initialization cell size to about 8 meters. The complete building detection algorithm is depicted in figure C.2.

C.3 Experiments

We conduct our experiments using three crops of different Miranda SAR images. We have manually annotated the images using a custom tool depicted in figure C.3. The annotation serves for training the algorithm and for computing error metrics. The area

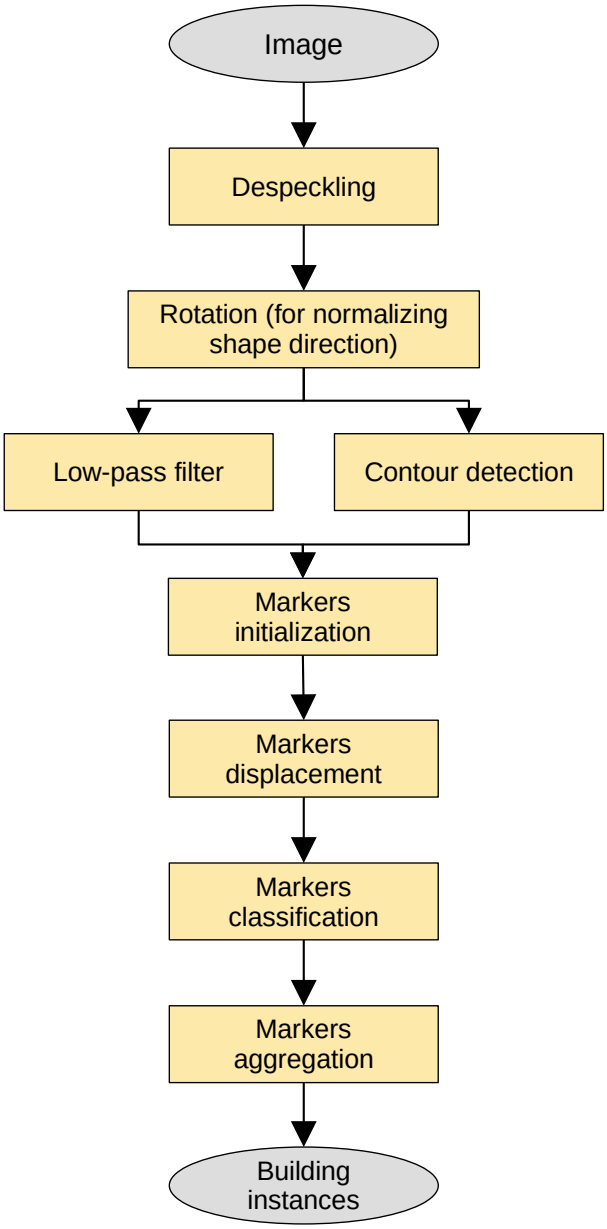


Figure C.2 – Building detection algorithm.

and the number of buildings of each case is reported is table C.1. We perform a three-fold cross validation study and report the detection recall and precision with a matching radius of 8 meters. Recall is the proportion of the true instances that are detected. Precision is the proportion of the detection that correspond to true instances. Our program was written in C++, without multithreading, and executed on an *Intel Core i7-12700* CPU.

The results are reported in table C.2 and in figures C.4, C.5 and C.6. The results for case I are poor. Trees are often taken for buildings because of their shades. The results for cases II and III are more satisfying. Long buildings are often missed however, certainly because of the shape of their shades which is not seen in the training images. The execution times are in the order of one second, which is quite competitive. The amount of training data was modest as well.

C.4 Conclusion

We have presented an algorithm for detecting object instances in SAR images and we have applied it for detecting buildings. Its execution time is relatively low and it can be trained from little data. Following this preliminary works, we should investigate the use of better features to better discriminate between trees and buildings. As well, we could try to detect other kinds of objects. Detecting objects that have known geographic coordinates may be part of a kinematic state estimation system based on correspondences between reference (e.g. ground) and observation (e.g. IDF-ID) coordinates.

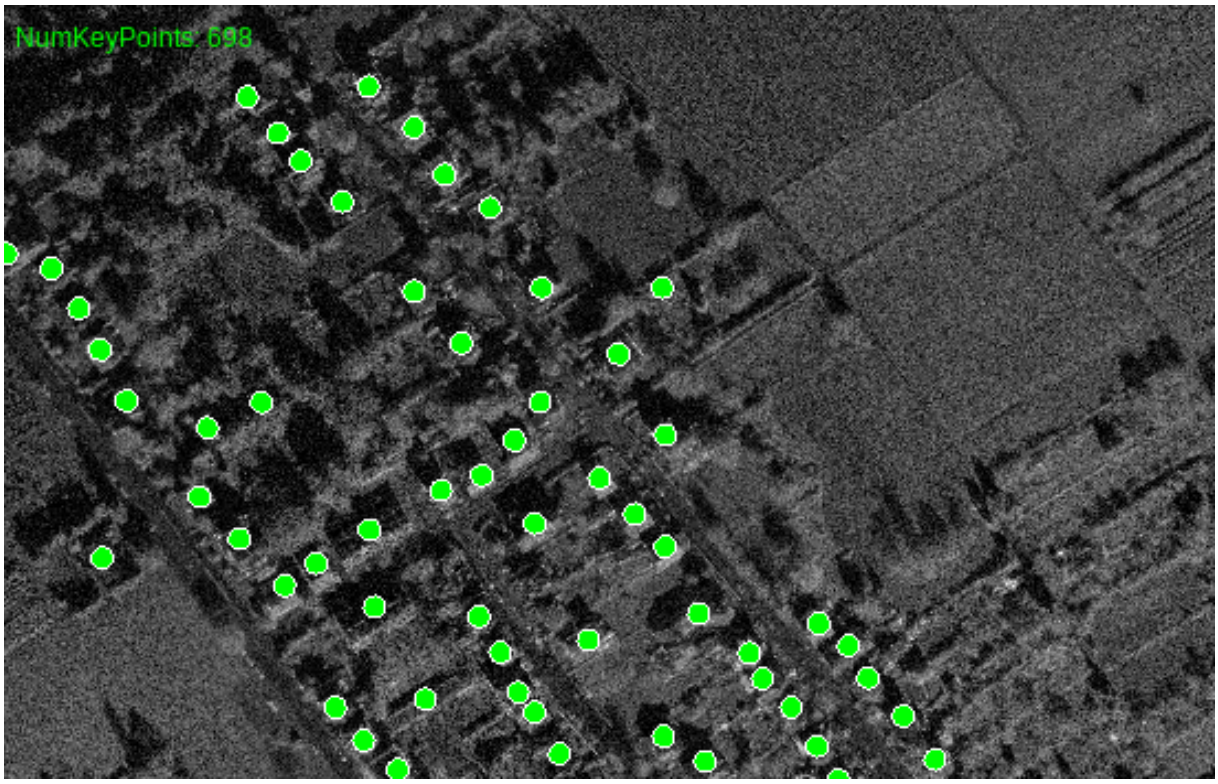


Figure C.3 – Screenshot of the annotation tool.

Case	Approx. ground area (km²)	Number of buildings
I	0.2	45
II	0.45	698
III	0.5	89

Table C.1 – Detection test cases.

Case	Recall	Precision	Inference time (s)
I	0.36	0.25	0.821
II	0.60	0.74	1.76
III	0.48	0.53	2.20

Table C.2 – Detection results.

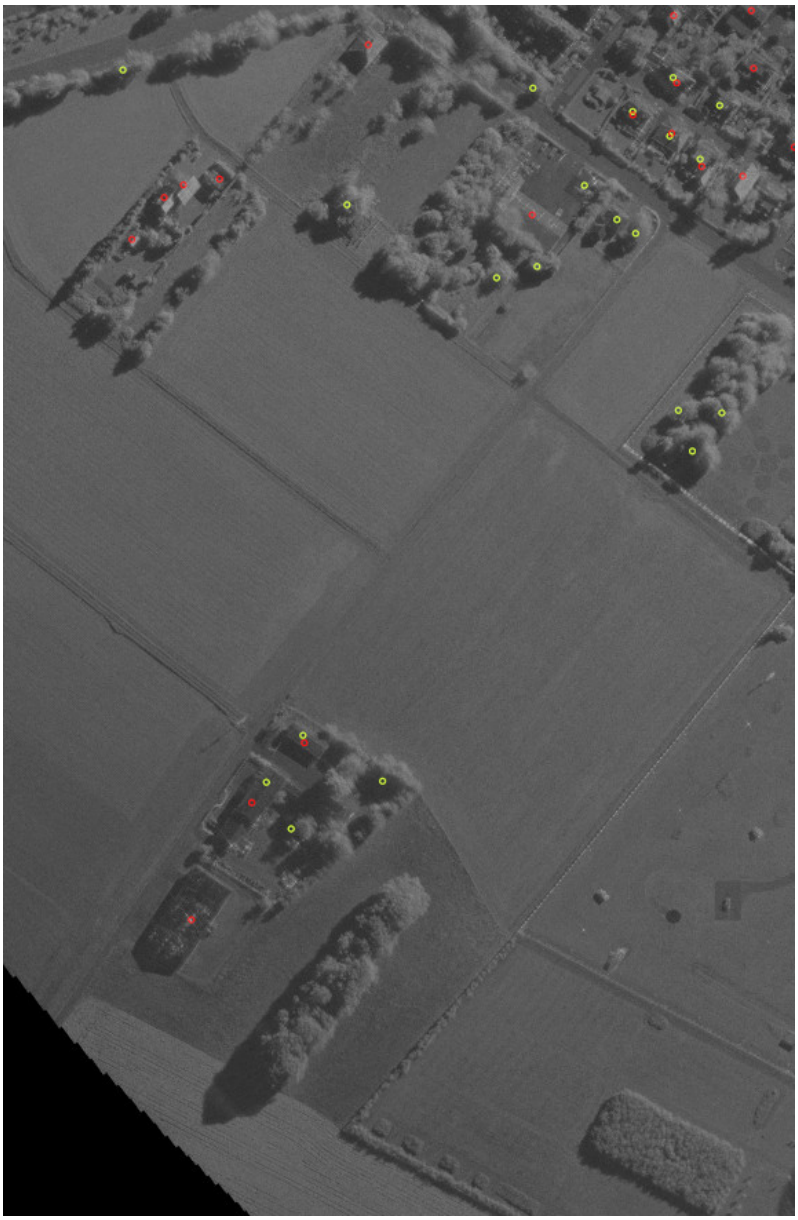


Figure C.4 – Results for case I. Red: ground-truth. Yellow: prediction.

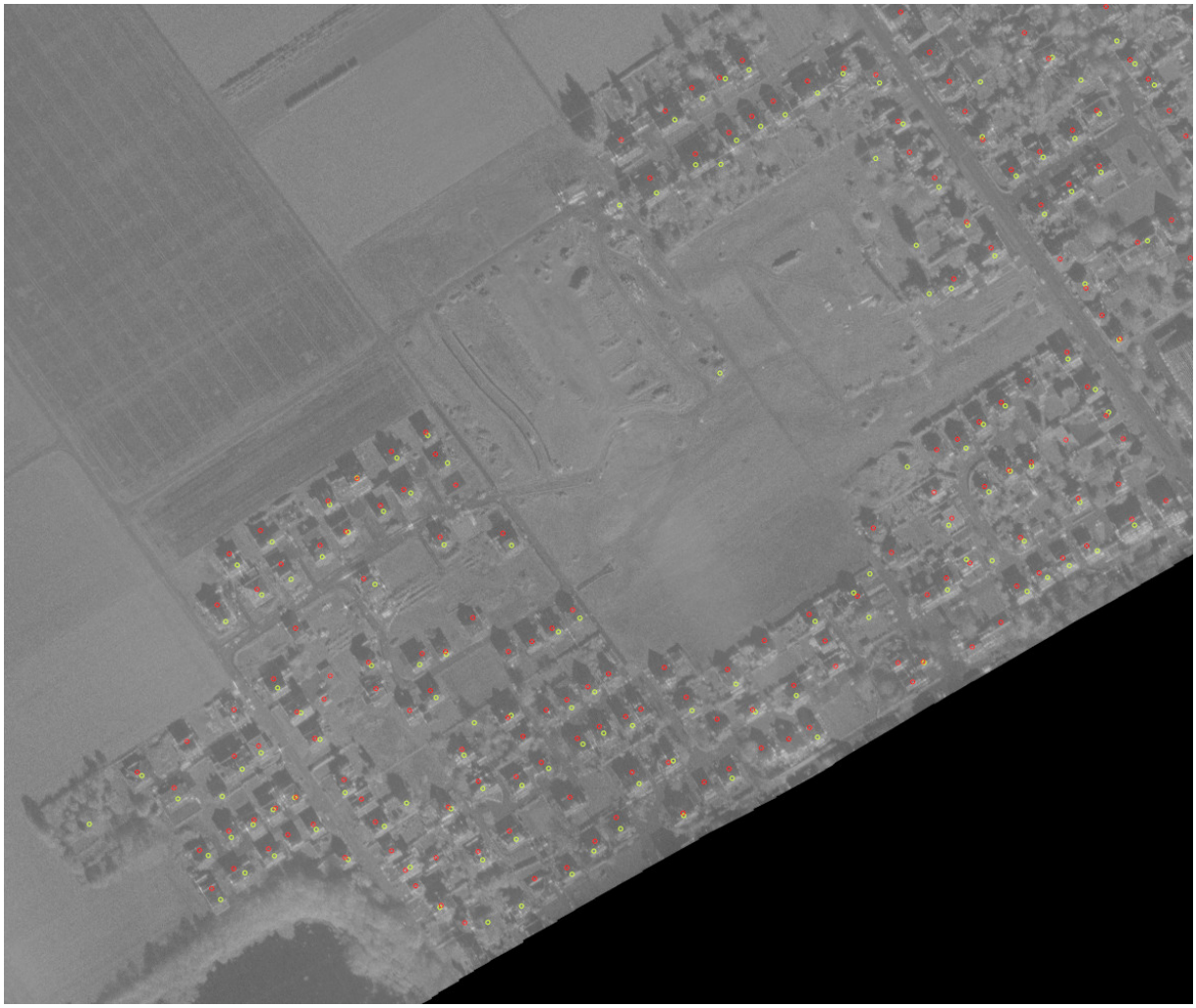


Figure C.5 – Results for case II. Red: ground-truth. Yellow: prediction.



Figure C.6 – Results for case III. Red: ground-truth. Yellow: prediction.

Bibliography

[1] Radhakrishna Achanta et al., « SLIC Superpixels Compared to State-of-the-Art Superpixel Methods », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.*11* (2012), pp. 2274–2282, DOI: 10.1109/TPAMI.2012.120.

[2] Cristhian A. Aguilera, Angel D. Sappa, and Ricardo Toledo, « LGHD: A feature descriptor for matching across non-linear intensity variations », *in*: *2015 IEEE International Conference on Image Processing (ICIP)*, 2015, pp. 178–181, DOI: 10.1109/ICIP.2015.7350783.

[3] Alireza Ahmadabadi, « Some properties of generalized gamma distribution », *in*: *Mathematical Sciences Quarterly Journal* 4 (Mar. 2010).

[4] Alberto Alonso-Gonzalez et al., « Processing Multidimensional SAR and Hyperspectral Images With Binary Partition Tree », *in*: *Proceedings of the IEEE* 101.*3* (Mar. 2013), Conference Name: Proceedings of the IEEE, pp. 723–747, ISSN: 1558-2256, DOI: 10.1109/JPROC.2012.2205209.

[5] Alberto Alonso-González, Carlos López-Martínez, and Philippe Salembier, « Filtering and segmentation of polarimetric SAR images with Binary Partition Trees », *in*: *2010 IEEE International Geoscience and Remote Sensing Symposium*, ISSN: 2153-7003, July 2010, pp. 4043–4046, DOI: 10.1109/IGARSS.2010.5653466.

[6] Hani Altwaijry et al., « Learning to Match Aerial Images with Deep Attentive Architectures », *in*: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, ISSN: 1063-6919, June 2016, pp. 3539–3547, DOI: 10.1109/CVPR.2016.385.

[7] Dongyang Ao et al., « Dialectical GAN for SAR Image Translation: From Sentinel-1 to TerraSAR-X », *in*: *Remote Sensing* 10.*10* (2018), ISSN: 2072-4292, DOI: 10.3390/rs10101597, URL: https://www.mdpi.com/2072-4292/10/10/1597.

[8] F. Argenti and L. Alparone, « Speckle removal from SAR images in the undecimated wavelet domain », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 40.*11* (Nov. 2002), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 2363–2374, ISSN: 1558-0644, DOI: 10.1109/TGRS.2002.805083.

[9] Jang Pyo Bae et al., « Cascaded Regression-Based Segmentation of Cardiac CT under Probabilistic Correspondences », *in*: *Applied Sciences* 10.*14* (2020), ISSN: 2076-3417, DOI: 10.3390/app10144947, URL: https://www.mdpi.com/2076-3417/10/14/4947.

[10] Zhengyao Bai and Peikun He, « An improved ratio edge detector for target detection in SAR images », *in*: *International Conference on Neural Networks and Signal Processing, 2003. Proceedings of the 2003*, vol. 2, Dec. 2003, 982–985 Vol.2, DOI: 10.1109/ICNNSP.2003.1280765.

[11] Zhengyao Bai et al., « An optimal edge detector for bridge target detection in SAR images », *in*: *Proceedings. 2005 International Conference on Communications, Circuits and Systems, 2005.* Vol. 2, May 2005, p. 851, DOI: 10.1109/ICCCAS.2005.1495242.

[12] Coloma Ballester, Vicent Caselles, and P. Monasse, « The tree of shapes of an image », en, *in*: *ESAIM: Control, Optimisation and Calculus of Variations* 9 (Jan. 2003), pp. 1–18, ISSN: 1292-8119, 1262-3377, DOI: 10.1051/cocv:2002069, URL: http://www.esaim-cocv.org/10.1051/cocv:2002069 (visited on 05/10/2021).

[13] Vassileios Balntas et al., « Learning local feature descriptors with triplets and shallow convolutional neural networks », en, *in*: *Procedings of the British Machine Vision Conference 2016*, York, UK: British Machine Vision Association, 2016, pp. 119.1–119.11, ISBN: 978-1-901725-59-9, DOI: 10.5244/C.30.119, URL: http://www.bmva.org/bmvc/2016/papers/paper119/index.html (visited on 07/04/2022).

[14] Vassileios Balntas et al., « PN-Net: Conjoined Triple Deep Network for Learning Local Image Descriptors », en, *in*: *arXiv:1601.05030 [cs]* (Jan. 2016), arXiv: 1601.05030, URL: http://arxiv.org/abs/1601.05030 (visited on 11/02/2021).

[15] Timo Balz, Horst Hammer, and Stefan Auer, « Potentials and limitations of SAR image simulators – A comparative study of three simulation approaches », en, *in*: *ISPRS Journal of Photogrammetry and Remote Sensing* 101 (Mar. 2015), pp. 102–109, ISSN: 0924-2716, DOI: 10.1016/j.isprsjprs.2014.12.008, URL: https://www.sciencedirect.com/science/article/pii/S0924271614002834 (visited on 06/28/2021).

[16] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, « SURF: Speeded Up Robust Features », en, *in*: *Computer Vision – ECCV 2006*, ed. by Aleš Leonardis, Horst Bischof, and Axel Pinz, vol. 3951, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 404–417, ISBN: 978-3-540-33832-1 978-3-540-33833-8, DOI: 10.1007/11744023_32, URL: http://link.springer.com/10.1007/11744023_32 (visited on 02/09/2022).

[17] Peter N Belhumeur, Joao P Hespanha, and David J Kriegman, « Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection », en, *in*: *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* 19.7 (1997), p. 10.

[18] J.A. Benediktsson, M. Pesaresi, and K. Amason, « Classification and feature extraction for remote sensing images from urban areas based on morphological transformations », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 41.*9* (Sept. 2003), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 1940–1949, ISSN: 1558-0644, DOI: 10.1109/TGRS.2003.814625.

[19] Jon Atli Benediktsson et al., « Hierarchical Analysis of Remote Sensing Data: Morphological Attribute Profiles and Binary Partition Trees », en, *in*: *Mathematical Morphology and Its Applications to Image and Signal Processing*, ed. by Pierre Soille, Martino Pesaresi, and Georgios K. Ouzounis, vol. 6671, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 306–319, ISBN: 978-3-642-21568-1 978-3-642-21569-8, DOI: 10.1007/978-3-642-21569-8_27, URL: http://link.springer.com/10.1007/978-3-642-21569-8_27 (visited on 05/10/2021).

[20] Jon Louis Bentley, « Multidimensional binary search trees used for associative searching », en, *in*: *Communications of the ACM* 18.*9* (Sept. 1975), pp. 509–517, ISSN: 0001-0782, 1557-7317, DOI: 10.1145/361002.361007, URL: https://dl.acm.org/doi/10.1145/361002.361007 (visited on 11/02/2022).

[21] P.J. Besl and Neil D. McKay, « A method for registration of 3-D shapes », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14.*2* (1992), pp. 239–256, DOI: 10.1109/34.121791.

[22] José M. Bioucas-Dias and Mário A. T. Figueiredo, « Multiplicative Noise Removal Using Variable Splitting and Constrained Optimization », *in*: *IEEE Transactions on Image Processing* 19.*7* (July 2010), Conference Name: IEEE Transactions on Image Processing, pp. 1720–1730, ISSN: 1941-0042, DOI: 10.1109/TIP.2010.2045029.

[23] Petra Bosilj et al., « Attribute Profiles from Partitioning Trees », en, *in*: *Mathematical Morphology and Its Applications to Signal and Image Processing*, ed. by Jesús Angulo, Santiago Velasco-Forero, and Fernand Meyer, vol. 10225, Series Title: Lecture Notes in Computer Science, Cham: Springer International Publishing, 2017, pp. 381–392, ISBN: 978-3-319-57239-0 978-3-319-57240-6, DOI: 10.1007/978-3-319-57240-6_31, URL: http://link.springer.com/10.1007/978-3-319-57240-6_31 (visited on 05/17/2021).

[24] A.C. Bovik, « On detecting edges in speckle imagery », *in*: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 36.*10* (Oct. 1988), Conference Name: IEEE Transactions on Acoustics, Speech, and Signal Processing, pp. 1618–1627, ISSN: 0096-3518, DOI: 10.1109/29.7550.

[25] Y. Boykov, O. Veksler, and R. Zabih, « Fast approximate energy minimization via graph cuts », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23.*11* (2001), pp. 1222–1239, DOI: 10.1109/34.969114.

[26] Eric Brachmann et al., « DSAC — Differentiable RANSAC for Camera Localization », *in*: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2492–2500, DOI: 10.1109/CVPR.2017.267.

[27] Edmond J. Breen and Ronald Jones, « Attribute Openings, Thinnings, and Granulometries », en, *in*: *Computer Vision and Image Understanding* 64.*3* (Nov. 1996), pp. 377–389, ISSN: 10773142, DOI: 10.1006/cviu.1996.0066, URL: https://linkinghub.elsevier.com/retrieve/pii/S1077314296900661 (visited on 05/25/2021).

[28] Frédéric Brigui et al., « SAR-Light - First SAR Images from the New Onera SAR Sensor on UAV Platform », *in*: *IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*, 2022, pp. 7721–7724, DOI: 10.1109/IGARSS46834.2022.9884340.

[29] M Brown, Gang Hua, and S Winder, « Discriminative Learning of Local Image Descriptors », en, *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.*1* (Jan. 2011), pp. 43–57, ISSN: 0162-8828, DOI: 10.1109/TPAMI.2010.54, URL: http://ieeexplore.ieee.org/document/5432199/ (visited on 04/06/2022).

[30] A. Buades, B. Coll, and J. M. Morel, « Image Denoising Methods. A New Nonlocal Principle », en, *in*: *SIAM Review* 52.*1* (Jan. 2010), pp. 113–147, ISSN: 0036-1445, 1095-7200, DOI: 10.1137/090773908, URL: http://epubs.siam.org/doi/10.1137/090773908 (visited on 05/10/2021).

[31] Harold C. Burger, Christian J. Schuler, and Stefan Harmeling, « Image denoising: Can plain neural networks compete with BM3D? », *in*: *2012 IEEE Conference on Computer Vision and Pattern Recognition*, ISSN: 1063-6919, June 2012, pp. 2392–2399, DOI: 10.1109/CVPR.2012.6247952.

[32] Tatjana Bürgmann, Wolfgang Koppe, and Michael Schmitt, « Matching of TerraSAR-X derived ground control points to optical image patches using deep learning | Elsevier Enhanced Reader », en, *in*: *ISPRS Journal of Photogrammetry and Remote Sensing* 158 (Sept. 2019), DOI: 10.1016/j.isprsjprs.2019.09.010, (visited on 10/26/2021).

[33] Michael Calonder et al., « BRIEF: Computing a Local Binary Descriptor Very Fast », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.*7* (2012).

[34] John Canny, « A Computational Approach to Edge Detection », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8.*6* (Nov. 1986), Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 679–698, ISSN: 1939-3539, DOI: 10.1109/TPAMI.1986.4767851.

[35] M. Caris et al., « Multi-Channel Radar at 35 GHz for Airborne SAR-Applications », *in*: *2018 19th International Radar Symposium (IRS)*, ISSN: 2155-5753, June 2018, pp. 1–8, DOI: 10.23919/IRS.2018.8448036.

[36] Jon R. Carr and James S. Sobek, « Digital Scene Matching Area Correlator (DSMAC) », *in*: *Image Processing for Missile Guidance*, ed. by Thomas F. Wiener, vol. 0238, Backup Publisher: International Society for Optics and Photonics, SPIE, 1980, pp. 36 –41, DOI: 10.1117/12.959130, URL: https://doi.org/10.1117/12.959130.

[37] Gabriele Cavallaro et al., « Region-based classification of remote sensing images with the morphological tree of shapes », en, *in*: *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Beijing, China: IEEE, July 2016, pp. 5087–5090, ISBN: 978-1-5090-3332-4, DOI: 10.1109/IGARSS.2016.7730326, URL: http://ieeexplore.ieee.org/document/7730326/ (visited on 05/10/2021).

[38] Stanley H. Chan, Xiran Wang, and Omar A. Elgendy, « Plug-and-Play ADMM for Image Restoration: Fixed-Point Convergence and Applications », *in*: *IEEE Transactions on Computational Imaging* 3.*1* (2017), pp. 84–98, DOI: 10.1109/TCI.2016.2629286.

[39] Yang-Lang Chang et al., « Ship Detection Based on YOLOv2 for SAR Imagery », en, *in*: *Remote Sensing* 11.*7* (Jan. 2019), Number: 7 Publisher: Multidisciplinary Digital Publishing Institute, p. 786, ISSN: 2072-4292, DOI: 10.3390/rs11070786, URL: https://www.mdpi.com/2072-4292/11/7/786 (visited on 03/28/2022).

[40] Jianlai Chen et al., « Motion Compensation/Autofocus in Airborne Synthetic Aperture Radar: A Review », *in*: *IEEE Geoscience and Remote Sensing Magazine* 10.*1* (2022), pp. 185–206, DOI: 10.1109/MGRS.2021.3113982.

[41] K. S. Chen, *Principles of Synthetic Aperture Radar Imaging: A System Simulation Approach*, en, Taylor & Francis Group, Sept. 2020, ISBN: 978-0-367-86837-6.

[42] Yang Chen and Gérard Medioni, « Object modelling by registration of multiple range images », *in*: *Image and Vision Computing* 10.*3* (1992), pp. 145–155, ISSN: 0262-8856, DOI: https://doi.org/10.1016/0262-8856(92)90066-C, URL: https://www.sciencedirect.com/science/article/pii/026288569290066C.

[43] G. Chierchia et al., « SAR image despeckling through convolutional neural networks », *in*: *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2017, pp. 5438–5441, DOI: 10.1109/IGARSS.2017.8128234.

[44] S. Chopra, R. Hadsell, and Y. LeCun, « Learning a similarity metric discriminatively, with application to face verification », *in*: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, ISSN: 1063-6919, June 2005, 539–546 vol. 1, DOI: 10.1109/CVPR.2005.202.

[45] D.A. Clausi and M.E. Jernigan, « A fast method to determine co-occurrence texture features », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 36.*1* (Jan. 1998), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 298–300, ISSN: 1558-0644, DOI: 10.1109/36.655338.

[46] David Clausi, « An analysis of co-occurrence texture statistics as a function of grey level quantization », *in*: *Canadian Journal of Remote Sensing* 28 (Feb. 2002), DOI: 10.5589/m02-004.

[47] David A. Clausi and Yongping Zhao, « Grey level co-occurrence integrated algorithm (GLCIA): a superior computational method to rapidly determine co-occurrence probability texture features », en, *in*: *Computers & Geosciences* 29.7 (Aug. 2003), pp. 837–850, ISSN: 00983004, DOI: 10.1016/S0098-3004(03)00089-X, URL: https://linkinghub.elsevier.com/retrieve/pii/S009830040300089X (visited on 01/24/2022).

[48] David A Clausi and Yongping Zhao, « Rapid extraction of image texture by co-occurrence using a hybrid data structure », en, *in*: *Computers & Geosciences* 28.6 (July 2002), pp. 763–774, ISSN: 00983004, DOI: 10.1016/S0098-3004(01)00108-X, URL: https://linkinghub.elsevier.com/retrieve/pii/S009830040100108X (visited on 01/24/2022).

[49] Alberto Crivellaro, Pascal Fua, and Vincent Lepetit, *Dense Methods for Image Alignment with an Application to 3D Tracking*, tech. rep., 2014, p. 25, URL: http://infoscience.epfl.ch/record/197866.

[50] Ian G. Cumming and Frank Hay-chee Wong, *Digital Processing of Synthetic Aperture Radar Data: Algorithms and Implementation*, en, Artech House, Jan. 2005, ISBN: 978-1-58053-058-3.

[51] Kostadin Dabov et al., « Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering », en, *in*: *IEEE Transactions on Image Processing* 16.8 (Aug. 2007), pp. 2080–2095, ISSN: 1057-7149, 1941-0042, DOI: 10.1109/TIP.2007.901238, URL: https://ieeexplore.ieee.org/document/4271520/ (visited on 11/05/2021).

[52] Kostadin Dabov et al., « Image denoising with block-matching and 3D filtering », *in*: *Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning*, vol. 6064, Backup Publisher: International Society for Optics and Photonics, SPIE, 2006, p. 606414, DOI: 10.1117/12.643267, URL: https://doi.org/10.1117/12.643267.

[53] N. Dalal and B. Triggs, « Histograms of Oriented Gradients for Human Detection », en, *in*: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, San Diego, CA, USA: IEEE, 2005, pp. 886–893, ISBN: 978-0-7695-2372-9, DOI: 10.1109/CVPR.2005.177, URL: http://ieeexplore.ieee.org/document/1467360/ (visited on 05/10/2021).

[54] Mauro Dalla Mura, Jon Atli Benediktsson, and Lorenzo Bruzzone, « Self-dual Attribute Profiles for the Analysis of Remote Sensing Images », en, *in*: *Mathematical Morphology and Its Applications to Image and Signal Processing*, ed. by Pierre Soille, Martino Pesaresi, and Georgios K. Ouzounis, Lecture Notes in Computer

Science, Berlin, Heidelberg: Springer, 2011, pp. 320–330, ISBN: 978-3-642-21569-8, DOI: 10.1007/978-3-642-21569-8_28.

[55] Mauro Dalla Mura et al., « Morphological Attribute Profiles for the Analysis of Very High Resolution Images », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 48.*10* (Oct. 2010), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 3747–3762, ISSN: 1558-0644, DOI: 10.1109/TGRS.2010.2048116.

[56] Emanuele Dalsasso, Loïc Denis, and Florence Tupin, « SAR2SAR: a semi-supervised despeckling algorithm for SAR images », *in*: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14 (2021), pp. 4321–4329, DOI: 10.1109/JSTARS.2021.3071864.

[57] Jerome Darbon et al., *Fast nonlocal filtering applied to electron cryomicroscopy*, Journal Abbreviation: 2008 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Proceedings, ISBI Pages: 1334 Publication Title: 2008 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Proceedings, ISBI, May 2008, DOI: 10.1109/ISBI.2008.4541250.

[58] Imant Daunhawer et al., « Self-supervised Disentanglement of Modality-Specific and Shared Factors Improves Multimodal Generative Models », *in*: *Pattern Recognition*, ed. by Zeynep Akata, Andreas Geiger, and Torsten Sattler, Cham: Springer International Publishing, 2021, pp. 459–473, ISBN: 978-3-030-71278-5.

[59] Charles Deledalle, Loic Denis, and Florence Tupin, « Iterative Weighted Maximum Likelihood Denoising With Probabilistic Patch-Based Weights », *in*: *Image Processing, IEEE Transactions on* 18 (Jan. 2010), pp. 2661–2672, DOI: 10.1109/TIP.2009.2029593.

[60] Charles-Alban Deledalle et al., « MuLoG, or How to apply Gaussian denoisers to multi-channel SAR speckle reduction? », *in*: *IEEE Transactions on Image Processing* 26.*9* (Sept. 2017), Publisher: Institute of Electrical and Electronics Engineers, pp. 4389–4403, DOI: 10.1109/TIP.2017.2713946, URL: https://hal.archives-ouvertes.fr/hal-01388858 (visited on 08/25/2021).

[61] Charles-Alban Deledalle et al., « NL-SAR: A Unified Nonlocal Framework for Resolution-Preserving (Pol)(In)SAR Denoising », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 53.*4* (Apr. 2015), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 2021–2038, ISSN: 1558-0644, DOI: 10.1109/TGRS.2014.2352555.

[62] Flora Dellinger et al., « SAR-SIFT: A SIFT-Like Algorithm for SAR Images », en, *in*: *IEEE Transactions on Geoscience and Remote Sensing* 53.*1* (Jan. 2015), pp. 453–466, ISSN: 0196-2892, 1558-0644, DOI: 10.1109/TGRS.2014.2323552, URL: http://ieeexplore.ieee.org/document/6824220/ (visited on 01/26/2022).

[63] Begüm Demir and Lorenzo Bruzzone, « Histogram based attribute profiles for classification of very high resolution remote sensing images », *in*: *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, ISSN: 2153-7003, July 2015, pp. 2393–2396, DOI: 10.1109/IGARSS.2015.7326291.

[64] Ozan Dogan and M. Kartal, « Time Domain SAR Raw Data Simulation of Distributed Targets », *in*: *EURASIP Journal on Advances in Signal Processing* 2010 (Jan. 2010), DOI: 10.1155/2010/784815.

[65] Piotr Dollár, Peter Welinder, and Pietro Perona, « Cascaded pose regression », *in*: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, ISSN: 1063-6919, June 2010, pp. 1078–1085, DOI: 10.1109/CVPR.2010.5540094.

[66] Dumitru Erhan et al., « Scalable Object Detection Using Deep Neural Networks », *in*: *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2155–2162, DOI: 10.1109/CVPR.2014.276.

[67] Martin A. Fischler and Robert C. Bolles, « Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography », en, *in*: *Communications of the ACM* 24.*6* (June 1981), pp. 381–395, ISSN: 0001-0782, 1557-7317, DOI: 10.1145/358669.358692, URL: https://dl.acm.org/doi/10.1145/358669.358692 (visited on 10/27/2022).

[68] Max Frioud et al., « Processing of MIRANDA35 FMCW-SAR Data using a Time-Domain Algorithm », *in*: *EUSAR 2014; 10th European Conference on Synthetic Aperture Radar*, June 2014, pp. 1–4.

[69] Victor S. Frost et al., « A Model for Radar Images and Its Application to Adaptive Digital Filtering of Multiplicative Noise », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-4.*2* (Mar. 1982), Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 157–166, ISSN: 1939-3539, DOI: 10.1109/TPAMI.1982.4767223.

[70] Victor S Frost et al., « Radar Image Preprocessing », en, *in*: *LARS Symposia*, 1980, p. 8, URL: http://docs.lib.purdue.edu/lars_symp/350/.

[71] Mario Fuentes Reyes et al., « SAR-to-Optical Image Translation Based on Conditional Generative Adversarial Networks—Optimization, Opportunities and Limits », *in*: *Remote Sensing* 11.*17* (2019), ISSN: 2072-4292, DOI: 10.3390/rs11172067, URL: https://www.mdpi.com/2072-4292/11/17/2067.

[72] Frédéric Galland et al., « Unsupervised Synthetic Aperture Radar Image Segmentation Using Fisher Distributions », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 47.*8* (Aug. 2009), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 2966–2972, ISSN: 1558-0644, DOI: 10.1109/TGRS.2009.2014364.

[73] S.S. Ganugapati and C.R. Moloney, « A ratio edge detector for speckled images based on maximum strength edge pruning », *in*: *Proceedings., International Conference on Image Processing*, vol. 2, Oct. 1995, 165–168 vol.2, DOI: 10.1109/ICIP.1995.537440.

[74] Yang Gao et al., « Sharpness-Based Autofocusing for Stripmap SAR Using an Adaptive-Order Polynomial Model », *in*: *IEEE Geoscience and Remote Sensing Letters* 11.*6* (2014), pp. 1086–1090, DOI: 10.1109/LGRS.2013.2286410.

[75] Nicolas Gasnier, Loic Denis, and Florence Tupin, « Generalized Likelihood Ratio Tests for Linear Structure Detection in SAR Images », *in*: *EUSAR 2021; 13th European Conference on Synthetic Aperture Radar*, Mar. 2021, pp. 1–6.

[76] Rafael Grompone von Gioi et al., « LSD: a Line Segment Detector », en, *in*: *Image Processing On Line* 2 (Mar. 2012), pp. 35–55, ISSN: 2105-1232, DOI: 10.5201/ipol.2012.gjmr-lsd, URL: https://www.ipol.im/pub/art/2012/gjmr-lsd/?utm_source=doi (visited on 01/05/2023).

[77] M. A. Gleyzes, L. Perret, and P. Kubik, « PLEIADES SYSTEM ARCHITECTURE AND MAIN PERFORMANCES », *in*: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XXXIX-B1 (2012), pp. 537–542, DOI: 10.5194/isprsarchives-XXXIX-B1-537-2012, URL: https://isprs-archives.copernicus.org/articles/XXXIX-B1/537/2012/.

[78] Joe P. Golden, « Terrain Contour Matching (TERCOM): A Cruise Missile Guidance Aid », *in*: *Image Processing for Missile Guidance*, ed. by Thomas F. Wiener, vol. 0238, Backup Publisher: International Society for Optics and Photonics, SPIE, 1980, pp. 10 –18, DOI: 10.1117/12.959127, URL: https://doi.org/10.1117/12.959127.

[79] M. Greco et al., « SAR-based augmented integrity navigation architecture », *in*: *2012 13th International Radar Symposium*, 2012, pp. 225–229, DOI: 10.1109/IRS.2012.6233320.

[80] M. Greco et al., « The study on SAR images exploitation for air platform navigation purposes », *in*: *2011 12th International Radar Symposium (IRS)*, 2011, pp. 347–352.

[81] Damian Gromek et al., « System Architecture for a Miniaturized X-band SAR-based Navigation Aid System », *in*: *2019 Signal Processing Symposium (SPSympo)*, 2019, pp. 271–276, DOI: 10.1109/SPS.2019.8882066.

[82] Cataldo Guaragnella and Tiziana D'Orazio, « B.SAR: blind SAR data focusing », *in*: *Image and Signal Processing for Remote Sensing XXIV*, ed. by Lorenzo Bruzzone and Francesca Bovolo, vol. 10789, Backup Publisher: International Society for Optics and Photonics, SPIE, 2018, p. 1078912, DOI: 10.1117/12.2324888, URL: https://doi.org/10.1117/12.2324888.

[83] Cataldo Guaragnella and Tiziana D'Orazio, « A Data-Driven Approach to SAR Data-Focusing », *in*: *Sensors* 19.*7* (2019), ISSN: 1424-8220, DOI: 10.3390/s19071649, URL: https://www.mdpi.com/1424-8220/19/7/1649.

[84] Robert Haralick, K. Shanmugam, and Ih Dinstein, « Textural Features for Image Classification », *in*: *IEEE Trans Syst Man Cybern* SMC-3 (Jan. 1973), pp. 610–621.

[85] C. Harris and M. Stephens, « A Combined Corner and Edge Detector », *in*: *Proceedings of the 4th Alvey Vision Conference*, 1988, pp. 147–151.

[86] Kaiming He et al., « Deep Residual Learning for Image Recognition », *in*: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778, DOI: 10.1109/CVPR.2016.90.

[87] Daniel Henke et al., « Miranda35 Experiments in Preparation for Small Uav-Based Sar », *in*: *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, ISSN: 2153-7003, July 2019, pp. 8542–8545, DOI: 10.1109/IGARSS.2019.8897902.

[88] Corentin Henry, Seyed Majid Azimi, and Nina Merkle, « Road Segmentation in SAR Satellite Images with Deep Fully-Convolutional Neural Networks », en, *in*: *IEEE Geoscience and Remote Sensing Letters* 15.*12* (Dec. 2018), pp. 1867–1871, DOI: 10.1109/LGRS.2018.2864342, (visited on 09/14/2021).

[89] MatthewS. Horritt, « A statistical active contour model for SAR image segmentation », en, *in*: *Image and Vision Computing* 17.*3* (Mar. 1999), pp. 213–224, ISSN: 0262-8856, DOI: 10.1016/S0262-8856(98)00101-2, URL: https://www.sciencedirect.com/science/article/pii/S0262885698001012 (visited on 05/10/2021).

[90] Lanqing Huang et al., « OpenSARShip: A Dataset Dedicated to Sentinel-1 Ship Interpretation », *in*: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 11.*1* (Jan. 2018).

[91] Xiaoxia Huang, Bo Huang, and Hongga Li, « A Fast Level Set Method for Synthetic Aperture Radar Ocean Image Segmentation », en, *in*: *Sensors* 9.*2* (Feb. 2009), Number: 2 Publisher: Molecular Diversity Preservation International, pp. 814–829, DOI: 10.3390/s90200814, URL: https://www.mdpi.com/1424-8220/9/2/814 (visited on 05/10/2021).

[92] Lloyd Hughes, Michael Schmitt, and Xiao Zhu, « Mining Hard Negative Samples for SAR-Optical Image Matching Using Generative Adversarial Networks », *in*: *Remote Sensing* 10 (Sept. 2018), DOI: 10.3390/rs10101552.

[93] Lloyd H. Hughes et al., « Identifying Corresponding Patches in SAR and Optical Images With a Pseudo-Siamese CNN », *in*: *IEEE Geoscience and Remote Sensing Letters* 15.*5* (May 2018), Conference Name: IEEE Geoscience and Remote Sensing Letters, pp. 784–788, ISSN: 1558-0571, DOI: 10.1109/LGRS.2018.2799232.

[94] Lloyd Haydn Hughes et al., « A deep learning framework for matching of SAR and optical imagery », en, *in*: *ISPRS Journal of Photogrammetry and Remote Sensing* 169 (Nov. 2020), pp. 166–179, ISSN: 0924-2716, DOI: 10.1016/j.isprsjprs.2020.09.012, URL: https://www.sciencedirect.com/science/article/pii/S0924271620302598 (visited on 06/28/2021).

[95] Lloyd Haydn Hughes et al., « Deep Learning for SAR-Optical Image Matching », *in*: *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, ISSN: 2153-7003, July 2019, pp. 4877–4880, DOI: 10.1109/IGARSS.2019.8898635.

[96] Ronny Hänsch, Olaf Hellwich, and Xiaohong Tu, « Machine-learning based detection of corresponding interest points in optical and SAR images », *in*: *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2016, pp. 1492–1495, DOI: 10.1109/IGARSS.2016.7729381.

[97] Ronny Hänsch, Olaf Hellwich, and Xi Wang, « Graph-cut segmentation of polarimetric SAR images », *in*: *2014 IEEE Geoscience and Remote Sensing Symposium*, ISSN: 2153-7003, July 2014, pp. 1733–1736, DOI: 10.1109/IGARSS.2014.6946786.

[98] Raffaele Imbriaco et al., « Aggregated Deep Local Features for Remote Sensing Image Retrieval », *in*: *Remote Sensing* 11.5 (2019), ISSN: 2072-4292, DOI: 10.3390/rs11050493, URL: https://www.mdpi.com/2072-4292/11/5/493.

[99] Geoffrey Irani and Hames Christ, « Image Processing for Tomahawk Scene Matching », *in*: *Johns Hopkins APL Technical Digest* 15.3 (1994).

[100] Phillip Isola et al., « Image-to-Image Translation with Conditional Adversarial Networks », *in*: *CVPR* (2017).

[101] M. Jahrer, Michael Grabner, and Horst Bischof, « Learned local descriptors for recognition and matching », *in*: *Proceedings of the Computer Vision Winter Workshop 2008*, Slovenian Pattern Recognition Society, 2008, pp. 39–46.

[102] Viren Jain and Hyunjune Seung, *Natural Image Denoising with Convolutional Networks*. Journal Abbreviation: Advances in Neural Information Processing Systems 21 - Proceedings of the 2008 Conference Pages: 776 Publication Title: Advances in Neural Information Processing Systems 21 - Proceedings of the 2008 Conference, Jan. 2008.

[103] Ronald Jones, « Connected Filtering and Segmentation Using Component Trees », en, *in*: *Computer Vision and Image Understanding* 75.3 (Sept. 1999), pp. 215–228, ISSN: 1077-3142, DOI: 10.1006/cviu.1999.0777, URL: https://www.sciencedirect.com/science/article/pii/S1077314299907774 (visited on 05/25/2021).

[104] C. Ju and C.R. Moloney, « An edge-enhanced segmentation method for SAR images », *in*: *CCECE '97. Canadian Conference on Electrical and Computer Engineering. Engineering Innovation: Voyage of Discovery. Conference Proceedings*, vol. 2, ISSN: 0840-7789, May 1997, 599–602 vol.2, DOI: 10.1109/CCECE.1997.608306.

[105] Hervé Jégou et al., « Aggregating local descriptors into a compact image representation », *in*: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 3304–3311, DOI: 10.1109/CVPR.2010.5540039.

[106] Alex Kendall, Matthew Grimes, and Roberto Cipolla, « PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization », *in*: *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2938–2946, DOI: 10.1109/ICCV.2015.336.

[107] Peter Kovesi, *Image Features From Phase Congruency*, en, tech. rep., The University of Western Australia - Robotics and Vision Research Group, 1995, p. 31.

[108] Peter Kovesi, « Phase congruency: A low-level image invariant », *in*: *Psychological research* 64 (Feb. 2000), pp. 136–48, DOI: 10.1007/s004260000024.

[109] Peter Kovesi, « Phase Congruency Detects Corners and Edges », *in*: *DICTA*, vol. 2003, Jan. 2003.

[110] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug, « Noise2Void - Learning Denoising From Single Noisy Images », *in*: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2124–2132, DOI: 10.1109/CVPR.2019.00223.

[111] Michael Krump and Peter Stütz, « Deep Learning Based Vehicle Detection on Real and Synthetic Aerial Images: Training Data Composition and Statistical Influence Analysis », *in*: *Sensors* 23.*7* (2023), ISSN: 1424-8220, DOI: 10.3390/s23073769, URL: https://www.mdpi.com/1424-8220/23/7/3769.

[112] Vladimir A. Krylov and James D. B. Nelson, « Stochastic Extraction of Elongated Curvilinear Structures With Applications », *in*: *IEEE Transactions on Image Processing* 23.*12* (Dec. 2014), Conference Name: IEEE Transactions on Image Processing, pp. 5360–5373, ISSN: 1941-0042, DOI: 10.1109/TIP.2014.2363612.

[113] Vladimir A. Krylov et al., « On the Method of Logarithmic Cumulants for Parametric Probability Density Function Estimation », en, *in*: *IEEE Transactions on Image Processing* 22.*10* (Oct. 2013), pp. 3791–3806, ISSN: 1057-7149, 1941-0042, DOI: 10.1109/TIP.2013.2262285, URL: http://ieeexplore.ieee.org/document/6544648/ (visited on 07/22/2021).

[114] Chiung-Shen Ku et al., « Imaging Simulation for Synthetic Aperture Radar: A Full-Wave Approach », en, *in*: *Remote Sensing* 10.*9* (Sept. 2018), Number: 9 Publisher: Multidisciplinary Digital Publishing Institute, p. 1404, DOI: 10.3390/rs10091404, URL: https://www.mdpi.com/2072-4292/10/9/1404 (visited on 06/28/2021).

[115] D. Kuan et al., « Adaptive restoration of images with speckle », *in*: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 35.*3* (Mar. 1987), Conference Name: IEEE Transactions on Acoustics, Speech, and Signal Processing, pp. 373–383, ISSN: 0096-3518, DOI: 10.1109/TASSP.1987.1165131.

[116] Darwin T. Kuan et al., « Adaptive Noise Smoothing Filter for Images with Signal-Dependent Noise », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-7.*2* (Mar. 1985), Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 165–177, ISSN: 1939-3539, DOI: 10.1109/TPAMI.1985.4767641.

[117] Samuli Laine et al., « High-Quality Self-Supervised Deep Image Denoising », *in*: *Advances in Neural Information Processing Systems*, ed. by H. Wallach et al., vol. 32, Curran Associates, Inc., 2019, URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/2119b8d43eafcf353e07d7cb5554170b-Paper.pdf.

[118] O. Lankoande, M.M. Hayat, and Balu Santhanam, « Segmentation of SAR images based on Markov random field model », *in*: *2005 IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, ISSN: 1062-922X, Oct. 2005, 2956–2961 Vol. 3, DOI: 10.1109/ICSMC.2005.1571599.

[119] Francesco Lattari et al., « Deep Learning for SAR Image Despeckling », en, *in*: *Remote Sensing* 11.*13* (Jan. 2019), Number: 13 Publisher: Multidisciplinary Digital Publishing Institute, p. 1532, DOI: 10.3390/rs11131532, URL: https://www.mdpi.com/2072-4292/11/13/1532 (visited on 08/25/2021).

[120] S. Lazebnik, C. Schmid, and J. Ponce, « Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories », en, *in*: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR'06)*, vol. 2, New York, NY, USA: IEEE, 2006, pp. 2169–2178, ISBN: 978-0-7695-2597-6, DOI: 10.1109/CVPR.2006.68, URL: http://ieeexplore.ieee.org/document/1641019/ (visited on 10/26/2021).

[121] Corentin Lebarbu et al., « Complete radar simulation chain: Application to maritime patrol surveillance using SAR/ISAR modes », *in*: *EUSAR 2021; 13th European Conference on Synthetic Aperture Radar*, 2021, pp. 1–6.

[122] Jérémy Lebreton et al., « Image simulation for space applications with the SurRender software », *in*: *ESA GNC 2021*, 2021.

[123] Jong-Sen Lee, « A simple speckle smoothing algorithm for synthetic aperture radar images », *in*: *IEEE Transactions on Systems, Man, and Cybernetics* SMC-13.*1* (Jan. 1983), Conference Name: IEEE Transactions on Systems, Man, and Cybernetics, pp. 85–89, ISSN: 2168-2909, DOI: 10.1109/TSMC.1983.6313036.

[124] Jong-Sen Lee, « Digital Image Enhancement and Noise Filtering by Use of Local Statistics », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-2.*2* (Mar. 1980), Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 165–168, ISSN: 1939-3539, DOI: 10.1109/TPAMI.1980.4766994.

[125] Jong-Sen Lee, « Digital image smoothing and the sigma filter », en, *in*: *Computer Vision, Graphics, and Image Processing* 24.*2* (Nov. 1983), pp. 255–269, ISSN: 0734-189X, DOI: 10.1016/0734-189X(83)90047-6, URL: https://www.sciencedirect.com/science/article/pii/0734189X83900476 (visited on 08/12/2021).

[126] Jong-Sen Lee, « Refined filtering of image noise using local statistics », en, *in*: *Computer Graphics and Image Processing* 15.*4* (Apr. 1981), pp. 380–389, ISSN: 0146-664X, DOI: 10.1016/S0146-664X(81)80018-4, URL: https://www.sciencedirect.com/science/article/pii/S0146664X81800184 (visited on 08/12/2021).

[127] Jong-Sen Lee, « Speckle analysis and smoothing of synthetic aperture radar images », en, *in*: *Computer Graphics and Image Processing* 17.*1* (Sept. 1981), pp. 24–32, ISSN: 0146-664X, DOI: 10.1016/S0146-664X(81)80005-6, URL: https://www.sciencedirect.com/science/article/pii/S0146664X81800056 (visited on 07/26/2021).

[128] Jong-Sen Lee et al., « Improved Sigma Filter for Speckle Filtering of SAR Imagery », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 47.*1* (Jan. 2009), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 202–213, ISSN: 1558-0644, DOI: 10.1109/TGRS.2008.2002881.

[129] Sang-Ryool Lee, « Overview of KOMPSAT-5 program, mission, and system », *in*: *2010 IEEE International Geoscience and Remote Sensing Symposium*, 2010, pp. 797–800, DOI: 10.1109/IGARSS.2010.5652759.

[130] Sun-Gu Lee et al., « Status of the Kompsat-5 SAR Mission, Utilization and Future Plans », *in*: *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, 2020, pp. 3552–3555, DOI: 10.1109/IGARSS39084.2020.9323172.

[131] Jaakko Lehtinen et al., « Noise2Noise: Learning Image Restoration without Clean Data », *in*: *Proceedings of the 35th International Conference on Machine Learning*, ed. by Jennifer Dy and Andreas Krause, vol. 80, Proceedings of Machine Learning Research, PMLR, July 2018, pp. 2965–2974, URL: https://proceedings.mlr.press/v80/lehtinen18a.html.

[132] Boying Li et al., « OpenSARShip 2.0: A large-volume dataset for deeper interpretation of ship targets in Sentinel-1 imagery », *in*: *2017 SAR in Big Data Era: Models, Methods and Applications (BIGSARDATA)*, 2017, pp. 1–5, DOI: 10.1109/BIGSARDATA.2017.8124929.

[133] Heng-Chao Li, Wen Hong, and Yi-Rong Wu, « Generalized gamma distribution with MoLC estimation for statistical modeling of SAR images », *in*: *2007 1st Asian and Pacific Conference on Synthetic Aperture Radar*, Nov. 2007, pp. 525–528, DOI: 10.1109/APSAR.2007.4418665.

[134] Xiaotian Li, Juha Ylioinas, and Juho Kannala, « Full-Frame Scene Coordinate Regression for Image-Based Localization », *in*: *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018, DOI: 10.15607/RSS.2018.XIV.015.

[135] Yun Liao et al., « Feature Matching and Position Matching Between Optical and SAR With Local Deep Feature Descriptor », *in*: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 15 (2022), Conference Name: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, pp. 448–462, ISSN: 2151-1535, DOI: 10.1109/JSTARS.2021.3134676.

[136] Colton Lindstrom et al., « GPS-Denied Navigation Aided by Synthetic Aperture Radar Using the Range-Doppler Algorithm », *in*: *NAVIGATION: Journal of the Institute of Navigation* 69.*3* (2022), Publisher: Institute of Navigation _eprint: https://navi.ion.org/content/69/3/navi.533.full.pdf, ISSN: 0028-1522, DOI: 10.33012/navi.533, URL: https://navi.ion.org/content/69/3/navi.533.

[137] Colton Lindstrom et al., « Sensitivity of back-projection algorithm (BPA) synthetic aperture radar (SAR) image formation to initial position, velocity, and attitude navigation errors », *in*: *IET Radar, Sonar & Navigation* 16.*2* (2022), _eprint: https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/rsn2.12189, pp. 364–378, DOI: https://doi.org/10.1049/rsn2.12189, URL: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/rsn2.12189.

[138] Chenguang Liu et al., « LSDSAR, a Markovian a contrario framework for line segment detection in SAR images », en, *in*: *Pattern Recognition* 98 (Feb. 2020), p. 107034, ISSN: 00313203, DOI: 10.1016/j.patcog.2019.107034, URL: https://linkinghub.elsevier.com/retrieve/pii/S0031320319303371 (visited on 05/10/2021).

[139] Gang Liu et al., « Contourlet-CNN for SAR Image Despeckling », en, *in*: *Remote Sensing* 13.*4* (Jan. 2021), Number: 4 Publisher: Multidisciplinary Digital Publishing Institute, p. 764, DOI: 10.3390/rs13040764, URL: https://www.mdpi.com/2072-4292/13/4/764 (visited on 08/30/2021).

[140] J. Liu and X. Mei, « Research on Edge Detection for SAR Images », *in*: *2006 IEEE International Symposium on Geoscience and Remote Sensing*, ISSN: 2153-7003, July 2006, pp. 3123–3126, DOI: 10.1109/IGARSS.2006.801.

[141] Jiaming Liu et al., « Image Restoration Using Total Variation Regularized Deep Image Prior », *in*: *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 7715–7719, DOI: 10.1109/ICASSP.2019.8682856.

[142] Wei Liu et al., « SSD: Single Shot MultiBox Detector », *in*: *Computer Vision – ECCV 2016*, ed. by Bastian Leibe et al., Cham: Springer International Publishing, 2016, pp. 21–37, ISBN: 978-3-319-46448-0.

[143] Zhe Liu, Xiaowei Fan, and Fangyuan Lv, « SAR Image Segmentation Using Contourlet and Support Vector Machine », *in*: *2009 Fifth International Conference on Natural Computation*, vol. 2, ISSN: 2157-9563, Aug. 2009, pp. 250–254, DOI: 10.1109/ICNC.2009.447.

[144] A. Lopes et al., « Maximum A Posteriori Speckle Filtering And First Order Texture Models In Sar Images », *in*: *10th Annual International Symposium on Geoscience and Remote Sensing*, May 1990, pp. 2409–2412, DOI: 10.1109/IGARSS.1990.689026.

[145] David G. Lowe, « Distinctive Image Features from Scale-Invariant Keypoints », en, *in*: *International Journal of Computer Vision* 60.*2* (Nov. 2004), pp. 91–110, ISSN: 0920-5691, DOI: 10.1023/B:VISI.0000029664.99615.94, URL: http://link.springer.com/10.1023/B:VISI.0000029664.99615.94 (visited on 07/01/2022).

[146] Stilla U Maksymiuk O Brenner A, « Detection of stationary vehicles in airborne decimeter resolution SAR intensity images using morphological attribute filters », *in*: *33. Wissenschaftlich-Technische Jahrestagung der DGPF*, Issue: 22, 2013.

[147] Stilla U Maksymiuk O Schmitt M, « Single tree detection in millimeterwave SAR data by morphological attribute filters », *in*: *34. Wissenschaftlich-Technische Jahrestagung der DGPF*, 2014.

[148] Jakub Markiewicz et al., « Geometrical Matching of SAR and Optical Images Utilizing ASIFT Features for SAR-based Navigation Aided Systems », en, *in*: *Sensors* 19.*24* (Jan. 2019), Number: 24 Publisher: Multidisciplinary Digital Publishing Institute, p. 5500, DOI: 10.3390/s19245500, URL: https://www.mdpi.com/1424-8220/19/24/5500 (visited on 06/28/2021).

[149] Victor Martin-Lac, Jacques Petit-Frere, and Jean-Marc Le Caillec, « A Generic, Multimodal Geospatial Data Alignment System for Aerial Navigation », *in*: *Remote Sensing* 15.*18* (2023), ISSN: 2072-4292, DOI: 10.3390/rs15184510, URL: https://www.mdpi.com/2072-4292/15/18/4510.

[150] Nina Merkle et al., « Exploiting Deep Matching and SAR Data for the Geo-Localization Accuracy Improvement of Optical Satellite Images », en, *in*: *Remote Sensing* 9.*6* (June 2017), Number: 6 Publisher: Multidisciplinary Digital Publishing Institute, p. 586, DOI: 10.3390/rs9060586, URL: https://www.mdpi.com/2072-4292/9/6/586 (visited on 06/28/2021).

[151] Krystian Mikolajczyk and Cordelia Schmid, « A Performance Evaluation of Local Descriptors », en, *in*: *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* 27.*10* (2005), p. 16.

[152] Anastasiia Mishchuk et al., « Working hard to know your neighbor's margins: Local descriptor learning loss », *in*: *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc., 2017, URL: https://proceedings.neurips.cc/paper/2017/hash/831caa1b600f852b7844499430ecac17-Abstract.html (visited on 08/23/2022).

[153] Andrea Bordone Molini et al., « Speckle2Void: Deep Self-Supervised SAR Despeckling With Blind-Spot Convolutional Neural Networks », *in*: *IEEE Transactions on Geoscience and Remote Sensing* (2021), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 1–17, ISSN: 1558-0644, DOI: 10.1109/TGRS.2021.3065461.

[154] Andrea Bordone Molini et al., « Towards Deep Unsupervised Sar Despeckling with Blind-Spot Convolutional Neural Networks », *in*: *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, 2020, pp. 2507–2510, DOI: 10.1109/IGARSS39084.2020.9324183.

[155] P. Monasse and F. Guichard, « Fast computation of a contrast-invariant image representation », *in*: *IEEE Transactions on Image Processing* 9.5 (May 2000), Conference Name: IEEE Transactions on Image Processing, pp. 860–872, ISSN: 1941-0042, DOI: 10.1109/83.841532.

[156] Gabriele Moser, Josiane Zerubia, and Sebastiano B Serpico, « Dictionary-Based Stochastic Expectation–Maximization for SAR Amplitude Probability Density Function Estimation », en, *in*: *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING* 44.1 (2006), p. 14.

[157] Gabriele Moser et al., « High resolution SAR-image classification by Markov random fields and finite mixtures », *in*: *Computational Imaging VIII*, ed. by Charles A. Bouman, Ilya Pollak, and Patrick J. Wolfe, vol. 7533, Backup Publisher: International Society for Optics and Photonics, SPIE, 2010, p. 753308, DOI: 10.1117/12.838594, URL: https://doi.org/10.1117/12.838594.

[158] Adugna G. Mullissa et al., « deSpeckNet: Generalizing Deep Learning Based SAR Image Despeckling », *in*: *IEEE Transactions on Geoscience and Remote Sensing* (2020), pp. 1–15, ISSN: 0196-2892, 1558-0644, DOI: 10.1109/TGRS.2020.3042694, (visited on 08/25/2021).

[159] Mauro Dalla Mura et al., « Morphological attribute filters for the analysis of very high resolution remote sensing images », *in*: *2009 IEEE International Geoscience and Remote Sensing Symposium*, vol. 3, ISSN: 2153-7003, July 2009, pp. III–97–III–100, DOI: 10.1109/IGARSS.2009.5418096.

[160] Cameron Musgrove, Richard Naething, and John Schilling, « Arbitrary scene simulation for synthetic aperture radar », *in*: *Radar Sensor Technology XVIII*, ed. by Kenneth I. Ranney and Armin Doerry, vol. 9077, Backup Publisher: International Society for Optics and Photonics, SPIE, 2014, p. 907709, DOI: 10.1117/12.2049875, URL: https://doi.org/10.1117/12.2049875.

[161] M. S. Müller, S. Urban, and B. Jutzi, « SQUEEZEPOSENET: IMAGE BASED POSE REGRESSION WITH SMALL CONVOLUTIONAL NEURAL NETWORKS FOR REAL TIME UAS NAVIGATION », *in*: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-2/W3 (2017), pp. 49–57, DOI: 10.5194/isprs-annals-IV-2-W3-49-2017, URL: https://isprs-annals.copernicus.org/articles/IV-2-W3/49/2017/.

[162] Jean-Marie Nicolas and Florence Tupin, « A New Parameterization for the Rician Distribution », en, *in*: *IEEE Geoscience and Remote Sensing Letters* 17.*11* (Nov. 2020), pp. 2011–2015, ISSN: 1545-598X, 1558-0571, DOI: 10.1109/LGRS.2019.2957240, URL: https://ieeexplore.ieee.org/document/8933060/ (visited on 07/23/2021).

[163] Davide O. Nitti et al., « Feasibility of Using Synthetic Aperture Radar to Aid UAV Navigation », en, *in*: *Sensors* 15.*8* (Aug. 2015), Number: 8 Publisher: Multidisciplinary Digital Publishing Institute, pp. 18334–18359, ISSN: 1424-8220, DOI: 10.3390/s150818334, URL: https://www.mdpi.com/1424-8220/15/8/18334 (visited on 12/09/2022).

[164] T. Ojala, M. Pietikainen, and T. Maenpaa, « Multiresolution gray-scale and rotation invariant texture classification with local binary patterns », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.*7* (July 2002), Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 971–987, ISSN: 1939-3539, DOI: 10.1109/TPAMI.2002.1017623.

[165] Mustafa Ozuysal et al., « Fast Keypoint Recognition Using Random Ferns », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.*3* (Mar. 2010), Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 448–461, ISSN: 1939-3539, DOI: 10.1109/TPAMI.2009.23.

[166] Yingxue Pang et al., « Image-to-Image Translation: Methods and Applications », *in*: *IEEE Transactions on Multimedia* 24 (2022), pp. 3859–3881, DOI: 10.1109/TMM.2021.3109419.

[167] Seonho Park et al., « Synthetic-Aperture Radar image based positioning in GPS-denied environments using Deep Cosine Similarity Neural Networks », *in*: *Inverse Problems & Imaging* 15 (Jan. 2020), pp. 1–23, DOI: 10.3934/ipi.2021013.

[168] S. Parrilli et al., « A nonlocal approach for SAR image denoising », en, *in*: *2010 IEEE International Geoscience and Remote Sensing Symposium*, Honolulu, HI, USA: IEEE, July 2010, pp. 726–729, ISBN: 978-1-4244-9565-8, DOI: 10.1109/IGARSS.2010.5651432, URL: http://ieeexplore.ieee.org/document/5651432/ (visited on 11/05/2021).

[169] Sara Parrilli et al., « A Nonlocal SAR Image Denoising Algorithm Based on LLMMSE Wavelet Shrinkage », en, *in*: *IEEE Transactions on Geoscience and Remote Sensing* 50.*2* (Feb. 2012), pp. 606–616, ISSN: 0196-2892, 1558-0644, DOI: 10.1109/TGRS.2011.2161586, URL: http://ieeexplore.ieee.org/document/5989862/ (visited on 11/05/2021).

[170] Stijn Peeters et al., « Classification using Extended Morphological Attribute Profiles based on different feature extraction techniques », *in*: *2011 IEEE International Geoscience and Remote Sensing Symposium*, ISSN: 2153-7003, July 2011, pp. 4453–4456, DOI: 10.1109/IGARSS.2011.6050221.

[171] Sonia Pelizzari and Jose Bioucas-Dias, « Oil spill segmentation of SAR images via graph cuts », *in*: *2007 IEEE International Geoscience and Remote Sensing Symposium*, ISSN: 2153-7003, July 2007, pp. 1318–1321, DOI: 10.1109/IGARSS.2007.4423048.

[172] Romuald Perrot, Pascal Bourdon, and David Helbert, « Implementing cascaded regression tree-based face landmarking: An in-depth overview », *in*: *Image and Vision Computing* 102 (2020), p. 103976, ISSN: 0262-8856, DOI: https://doi.org/10.1016/j.imavis.2020.103976, URL: https://www.sciencedirect.com/science/article/pii/S0262885620301086.

[173] M. Pesaresi and J.A. Benediktsson, « A new approach for the morphological segmentation of high-resolution satellite imagery », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 39.*2* (2001), pp. 309–320, DOI: 10.1109/36.905239.

[174] Minh-Tan Pham, Erchan Aptoula, and Sebastien Lefevre, « Classification of Remote Sensing Images Using Attribute Profiles and Feature Profiles from Different Trees: A Comparative Study », en, *in*: *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, Valencia: IEEE, July 2018, pp. 4511–4514, ISBN: 978-1-5386-7150-4, DOI: 10.1109/IGARSS.2018.8517735, URL: https://ieeexplore.ieee.org/document/8517735/ (visited on 05/17/2021).

[175] Minh-Tan Pham, Erchan Aptoula, and Sébastien Lefèvre, « Feature Profiles from Attribute Filtering for Classification of Remote Sensing Images », *in*: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 11.*1* (Jan. 2018), Conference Name: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, pp. 249–256, ISSN: 2151-1535, DOI: 10.1109/JSTARS.2017.2773367.

[176] Matt Pharr, Wenzel Jakob, and Greg Humphreys, *Physically Based Rendering: From Theory to Implementation*, 3rd, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2016, ISBN: 0-12-800645-5.

[177] Nicholas Redding et al., « An efficient algorithm for Mumford-Shah segmentation and its application to SAR imagery », *in*: *Proceedings of the 1999 Conference on Digital Image Computing: Techniques and Applications (DICTA-99)*, Dec. 1999, pp. 35–41.

[178] Joseph Redmon et al., « You Only Look Once: Unified, Real-Time Object Detection », *in*: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788, DOI: 10.1109/CVPR.2016.91.

[179] Shaoqing Ren et al., « Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks », *in*: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.*6* (2017), pp. 1137–1149, DOI: 10.1109/TPAMI.2016.2577031.

[180] Angel Ribalta, « Time-Domain Reconstruction Algorithms for FMCW-SAR », *in*: *IEEE Geoscience and Remote Sensing Letters* 8.*3* (2011), pp. 396–400, DOI: 10.1109/LGRS.2010.2078486.

[181] Mark A. Richards, *Fundamentals of Radar Signal Processing*, en, McGraw Hill Professional, June 2005, ISBN: 978-0-07-144474-3.

[182] Jos Roerdink and A. Meijster, « The Watershed Transform: Definitions, Algorithms and Parallelization Strategies », *in*: *Fundam Inf* 41 (Oct. 2003), DOI: 10.3233/FI-2000-411207.

[183] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, « U-Net: Convolutional Networks for Biomedical Image Segmentation », *in*: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, ed. by Nassir Navab et al., Cham: Springer International Publishing, 2015, pp. 234–241, ISBN: 978-3-319-24574-4.

[184] Timothy Ross et al., « Standard SAR ATR evaluation experiments using the MSTAR public release data set », *in*: *PROCEEDINGS OF SPIE*, Orlando, FL, United States: SPIE, 1998.

[185] Tamar Rott Shaham et al., « Spatially-Adaptive Pixelwise Networks for Fast Image Translation », *in*: *Computer Vision and Pattern Recognition (CVPR)*, 2021.

[186] Ethan Rublee et al., « ORB: An efficient alternative to SIFT or SURF », *in*: *2011 International Conference on Computer Vision*, ISSN: 2380-7504, Nov. 2011, pp. 2564–2571, DOI: 10.1109/ICCV.2011.6126544.

[187] P. Salembier and L. Garrido, « Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval », *in*: *IEEE Transactions on Image Processing* 9.*4* (Apr. 2000), Conference Name: IEEE Transactions on Image Processing, pp. 561–576, ISSN: 1941-0042, DOI: 10.1109/83.841934.

[188] Philippe Salembier and Samuel Foucher, « Optimum Graph Cuts for Pruning Binary Partition Trees of Polarimetric SAR Images », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 54.*9* (Sept. 2016), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 5493–5502, ISSN: 1558-0644, DOI: 10.1109/TGRS.2016.2566581.

[189] Philippe Salembier, Sergi Liesegang, and Carlos López-Martínez, « Ship Detection in SAR Images Based on Maxtree Representation and Graph Signal Processing », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 57.*5* (May 2019), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 2709–2724, ISSN: 1558-0644, DOI: 10.1109/TGRS.2018.2876603.

[190] Philippe Salembier and Michael H.F. Wilkinson, « Connected operators », *in*: *IEEE Signal Processing Magazine* 26.*6* (Nov. 2009), Conference Name: IEEE Signal Processing Magazine, pp. 136–157, ISSN: 1558-0792, DOI: 10.1109/MSP.2009.934154.

[191] M. Schmitt et al., « SEN12MS – A CURATED DATASET OF GEOREFERENCED MULTI-SPECTRAL SENTINEL-1/2 IMAGERY FOR DEEP LEARNING AND DATA FUSION », *in*: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-2/W7 (2019), pp. 153–160, DOI: 10.5194/isprs-annals-IV-2-W7-153-2019, URL: https://isprs-annals.copernicus.org/articles/IV-2-W7/153/2019/.

[192] Michael Schmitt, Lloyd Haydn Hughes, and Xiao Xiang Zhu, « The SEN1-2 Dataset for Deep Learning in SAR-Optical Data Fusion », *in*: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-1 (July 2018), pp. 141–146, DOI: 10.5194/isprs-annals-IV-1-141-2018.

[193] Florian Schroff, Dmitry Kalenichenko, and James Philbin, « FaceNet: A Unified Embedding for Face Recognition and Clustering », en, *in*: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 815–823, DOI: 10.1109/CVPR.2015.7298682, (visited on 07/13/2022).

[194] Arivazhagan Selvaraj, Subash Kumar, and Lakshmanan Ganesan, « Texture classification using curvelet transform », *in*: *International Journal of Wavelets Multiresolution and Information Processing* 5 (May 2007), DOI: 10.1142/S0219691307001847.

[195] Alexander Semenov, Maciej Rysz, and Garrett Demeyer, « Synthetic Aperture Radar Image Based Navigation Using Siamese Neural Networks », *in*: *Synthetic Aperture Radar (SAR) Data Applications*, ed. by Maciej Rysz et al., Cham: Springer International Publishing, 2022, pp. 79–89, ISBN: 978-3-031-21225-3, DOI: 10.1007/978-3-031-21225-3_4, URL: https://doi.org/10.1007/978-3-031-21225-3_4.

[196] Eli Shechtman and Michal Irani, « Matching Local Self-Similarities across Images and Videos », *in*: *2007 IEEE Conference on Computer Vision and Pattern Recognition*, ISSN: 1063-6919, June 2007, pp. 1–8, DOI: 10.1109/CVPR.2007.383198.

[197] Yuge Shi et al., « Variational Mixture-of-Experts Autoencoders for Multi-Modal Deep Generative Models », *in*: *Advances in Neural Information Processing Systems*, ed. by H. Wallach et al., vol. 32, Curran Associates, Inc., 2019.

[198] Jamie Shotton et al., « Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images », *in*: *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2930–2937, DOI: 10.1109/CVPR.2013.377.

[199] Peng-Lang Shui and Dong Cheng, « Edge Detector of SAR Images Using Gaussian-Gamma-Shaped Bi-Windows », *in*: *IEEE Geoscience and Remote Sensing Letters* 9.*5* (Sept. 2012), Conference Name: IEEE Geoscience and Remote Sensing Letters, pp. 846–850, ISSN: 1558-0571, DOI: 10.1109/LGRS.2012.2184521.

[200] K Simonyan and A Zisserman, « Very deep convolutional networks for large-scale image recognition », *in*: *3rd International Conference on Learning Representations (ICLR 2015)*, Computational and Biological Learning Society, 2015, pp. 1–14.

[201] G.M. Siouris, *Missile Guidance and Control Systems*, Springer New York, 2006, ISBN: 978-0-387-21816-8.

[202] L. Sirovich and M. Kirby, « Low-dimensional procedure for the characterization of human faces », en, *in*: *Journal of the Optical Society of America A* 4.*3* (Mar. 1987), p. 519, ISSN: 1084-7529, 1520-8532, DOI: 10.1364/JOSAA.4.000519, URL: https://opg.optica.org/abstract.cfm?URI=josaa-4-3-519 (visited on 07/13/2022).

[203] Sivic and Zisserman, « Video Google: a text retrieval approach to object matching in videos », *in*: *Proceedings Ninth IEEE International Conference on Computer Vision*, 2003, 1470–1477 vol.2, DOI: 10.1109/ICCV.2003.1238663.

[204] Zoran Sjanic and Fredrik Gustafsson, « Navigation and SAR focusing with map aiding », *in*: *IEEE Transactions on Aerospace and Electronic Systems* 51.*3* (2015), pp. 1652–1663, DOI: 10.1109/TAES.2015.130397.

[205] Zoran Sjanic and Fredrik Gustafsson, « Simultaneous navigation and SAR autofocusing », *in*: *2010 13th International Conference on Information Fusion*, 2010, pp. 1–7, DOI: 10.1109/ICIF.2010.5711931.

[206] S. Solbo and T. Eltoft, « Homomorphic wavelet-based statistical despeckling of SAR images », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 42.*4* (Apr. 2004), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 711–721, ISSN: 1558-0644, DOI: 10.1109/TGRS.2003.821885.

[207] Stian Solbø and Torbjorn Eltoft, « T-WMAP: A statistical speckle filter operating in the wavelet domain », *in*: *International Journal of Remote Sensing* 25 (Mar. 2004), pp. 1019–1036, DOI: 10.1080/0143116031000150040.

[208] Roberto Souza et al., « An Overview of Max-Tree Principles, Algorithms and Applications », *in*: *2016 29th SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)*, 2016, pp. 15–23, DOI: 10.1109/SIBGRAPI-T.2016.011.

[209] E. W. Stacy, « A Generalization of the Gamma Distribution », en, *in*: *The Annals of Mathematical Statistics* 33.*3* (Sept. 1962), pp. 1187–1192, ISSN: 0003-4851, DOI: 10.1214/aoms/1177704481, URL: http://projecteuclid.org/euclid.aoms/1177704481 (visited on 07/22/2021).

[210] Stephan Stanko et al., « SAR with MIRANDA - millimeterwave radar using analog and new digital approach », *in*: *2011 8th European Radar Conference*, Oct. 2011, pp. 214–217.

[211] M.B. Stevens and D.L. Perks, « I-Master RADAR: Recent trials results », *in*: *IET International Conference on Radar Systems (Radar 2012)*, 2012, pp. 1–5, DOI: 10.1049/cp.2012.1640.

[212] Christopher Stewart et al., « Deep Learning with Open Data for Desert Road Mapping », *in*: *Remote Sensing* 12 (July 2020), p. 2274, DOI: 10.3390/rs12142274.

[213] Craig Stringham and David G. Long, « Processing for UWB LFM-CW SAR », *in*: *2014 IEEE Geoscience and Remote Sensing Symposium*, 2014, pp. 1105–1108, DOI: 10.1109/IGARSS.2014.6946622.

[214] Xiao Sun et al., « Cascaded hand pose regression », *in*: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 824–832, DOI: 10.1109/CVPR.2015.7298683.

[215] Yuxi Sun et al., « Multisensor Fusion and Explicit Semantic Preserving-Based Deep Hashing for Cross-Modal Remote Sensing Image Retrieval », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 60 (2022), pp. 1–14, DOI: 10.1109/TGRS.2021.3136641.

[216] Sonia Tabti, Julien Rabin, and Abderrahim Elmoata, « Symmetric Upwind Scheme for Discrete Weighted Total Variation », *in*: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, ISSN: 2379-190X, Apr. 2018, pp. 1827–1831, DOI: 10.1109/ICASSP.2018.8461736.

[217] Yaniv Taigman et al., « DeepFace: Closing the Gap to Human-Level Performance in Face Verification », en, *in*: *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA: IEEE, June 2014, pp. 1701–1708, ISBN: 978-1-4799-5118-5, DOI: 10.1109/CVPR.2014.220, URL: https://ieeexplore.ieee.org/document/6909616 (visited on 10/29/2021).

[218] Yurun Tian, Bin Fan, and Fuchao Wu, « L2-Net: Deep Learning of Discriminative Patch Descriptor in Euclidean Space », *in*: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, ISSN: 1063-6919, July 2017, pp. 6128–6136, DOI: 10.1109/CVPR.2017.649.

[219] Ayşe Tombak et al., « Pixel-Based Classification of SAR Images Using Feature Attribute Profiles », *in*: *IEEE Geoscience and Remote Sensing Letters* 16.*4* (Apr. 2019), Conference Name: IEEE Geoscience and Remote Sensing Letters, pp. 564–567, ISSN: 1558-0571, DOI: 10.1109/LGRS.2018.2879880.

[220] Zhengzheng Tu et al., « Multimodal salient object detection via adversarial learning with collaborative generator », *in*: *Engineering Applications of Artificial Intelligence* 119 (2023), p. 105707, ISSN: 0952-1976, DOI: https://doi.org/10.1016/j.engappai.2022.105707, URL: https://www.sciencedirect.com/science/article/pii/S0952197622006972.

[221] Matthew Turk and Alex Pentland, « Eigenfaces for Recognition », *in*: *Journal of Cognitive Neuroscience* 3.*1* (Jan. 1991), pp. 71–86, ISSN: 0898-929X, DOI: 10.1162/jocn.1991.3.1.71, URL: https://doi.org/10.1162/jocn.1991.3.1.71 (visited on 08/23/2022).

[222] Eric Tzeng et al., « Adversarial Discriminative Domain Adaptation », *in*: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[223] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky, « Deep Image Prior », en, *in*: *International Journal of Computer Vision* 128.*7* (July 2020), pp. 1867–1888, ISSN: 0920-5691, 1573-1405, DOI: 10.1007/s11263-020-01303-4.

[224] Paul Viola and William M. Wells III, « Alignment by Maximization of Mutual Information », *in*: *International Journal of Computer Vision* 24.*2* (Sept. 1997), pp. 137–154, ISSN: 1573-1405, DOI: 10.1023/A:1007958904918, URL: https://doi.org/10.1023/A:1007958904918.

[225] Sergio Vitale, Giampaolo Ferraioli, and Vito Pascazio, « Multi-Objective CNN Based Algorithm for SAR Despeckling », *in*: *IEEE Transactions on Geoscience and Remote Sensing* (2020), pp. 1–14, ISSN: 0196-2892, 1558-0644, DOI: 10.1109/TGRS.2020.3034852.

[226] D.E. Wahl et al., « Phase gradient autofocus-a robust tool for high resolution SAR phase correction », *in*: *IEEE Transactions on Aerospace and Electronic Systems* 30.*3* (1994), pp. 827–835, DOI: 10.1109/7.303752.

[227] Lei Wang et al., « SAR-to-Optical Image Translation Using Supervised Cycle-Consistent Adversarial Networks », *in*: *IEEE Access* 7 (2019), pp. 129136–129149, DOI: 10.1109/ACCESS.2019.2939649.

[228] Puyang Wang, He Zhang, and Vishal M. Patel, « Generative adversarial network-based restoration of speckled SAR images », *in*: *2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Dec. 2017, pp. 1–5, DOI: 10.1109/CAMSAP.2017.8313133.

[229] Puyang Wang, He Zhang, and Vishal M. Patel, « SAR Image Despeckling Using a Convolutional Neural Network », *in*: *IEEE Signal Processing Letters* 24.*12* (2017), pp. 1763–1767, DOI: 10.1109/LSP.2017.2758203.

[230] Shuang Wang et al., « An improved watershed-based SAR image segmentation algorithm », *in*: *MIPPR 2009: Automatic Target Recognition and Image Analysis*, ed. by Tianxu Zhang et al., vol. 7495, Backup Publisher: International Society for Optics and Photonics, SPIE, 2009, 74951R, DOI: 10.1117/12.832892, URL: https://doi.org/10.1117/12.832892.

[231] Xiaying Wang et al., « HR-SAR-Net: A Deep Neural Network for Urban Scene Segmentation from High-Resolution SAR Data », *in*: *2020 IEEE Sensors Applications Symposium (SAS)*, 2020, pp. 1–6, DOI: 10.1109/SAS48726.2020.9220068.

[232] Yuanyuan Wang and Xiao Xiang Zhu, « The SARptical Dataset for Joint Analysis of SAR and Optical Image in Dense Urban Area », en, *in*: *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, Valencia: IEEE, July 2018, pp. 6840–6843, ISBN: 978-1-5386-7150-4, DOI: 10.1109/IGARSS.2018.8518298, URL: https://ieeexplore.ieee.org/document/8518298/ (visited on 06/29/2022).

[233] Kilian Q. Weinberger and Lawrence K. Saul, « Distance Metric Learning for Large Margin Nearest Neighbor Classification », *in*: *Journal of Machine Learning Research* 10.*9* (2009), pp. 207–244, ISSN: 1533-7928, URL: http://jmlr.org/papers/v10/weinberger09a.html (visited on 08/23/2022).

[234] Teresa White et al., « Gps-Denied Navigation Using Sar Images And Neural Networks », *in*: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 2395–2399, DOI: 10.1109/ICASSP39728.2021.9414421.

[235] Ben Willetts et al., « Spectrum analysis of high-resolution SAR data to obtain Bragg signatures of power cables », *in*: *IET Radar, Sonar & Navigation* 12.*8* (2018), pp. 839–843, DOI: 10.1049/iet-rsn.2017.0574.

[236] Fanzi Wu et al., « Cascaded regression using landmark displacement for 3D face reconstruction », *in*: *Pattern Recognition Letters* 125 (2019), pp. 766–772, ISSN: 0167-8655, DOI: https://doi.org/10.1016/j.patrec.2019.07.017, URL: https://www.sciencedirect.com/science/article/pii/S0167865518303933.

[237] Mike Wu and Noah Goodman, « Multimodal Generative Models for Scalable Weakly-Supervised Learning », *in*: *Advances in Neural Information Processing Systems*, ed. by S. Bengio et al., vol. 31, Curran Associates, Inc., 2018.

[238] Gui-Song Xia, Chu He, and Hong Sun, « A Rapid and Automatic MRF-Based Clustering Method for SAR Images », *in*: *IEEE Geoscience and Remote Sensing Letters* 4.*4* (Oct. 2007), Conference Name: IEEE Geoscience and Remote Sensing Letters, pp. 596–600, ISSN: 1558-0571, DOI: 10.1109/LGRS.2007.903065.

[239] Yuming Xiang, Feng Wang, and Hongjian You, « OS-SIFT: A Robust SIFT-Like Algorithm for High-Resolution Optical-to-SAR Image Registration in Suburban Areas », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 56.*6* (2018), pp. 3078–3090, DOI: 10.1109/TGRS.2018.2790483.

[240] Yuming Xiang et al., « Automatic Registration of Optical and SAR Images Via Improved Phase Congruency Model », *in*: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13 (2020), Conference Name: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, pp. 5847–5861, ISSN: 2151-1535, DOI: 10.1109/JSTARS.2020.3026162.

[241] Yuming Xiang et al., « SAR-PC: Edge Detection in SAR Images via an Advanced Phase Congruency Model », en, *in*: *Remote Sensing* 9.*3* (Mar. 2017), Number: 3 Publisher: Multidisciplinary Digital Publishing Institute, p. 209, ISSN: 2072-4292, DOI: 10.3390/rs9030209, URL: https://www.mdpi.com/2072-4292/9/3/209 (visited on 01/05/2023).

[242] Hua Xie, L.E. Pierce, and F.T. Ulaby, « Mutual information based registration of SAR images », *in*: *IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No.03CH37477)*, vol. 6, July 2003, 4028–4031 vol.6, DOI: 10.1109/IGARSS.2003.1295351.

[243] Tao Xiong et al., « Minimum-Entropy-Based Autofocus Algorithm for SAR Data Using Chebyshev Approximation and Method of Series Reversion, and Its Implementation in a Data Processor », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 52.*3* (2014), pp. 1719–1728, DOI: 10.1109/TGRS.2013.2253781.

[244] Wei Xiong et al., « A Deep Cross-Modality Hashing Network for SAR and Optical Remote Sensing Images Retrieval », *in*: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13 (2020), pp. 5284–5296, DOI: 10.1109/JSTARS.2020.3021390.

[245] Xin Xiong et al., « Rank-Based Local Self-Similarity Descriptor for Optical-to-SAR Image Matching », *in*: *IEEE Geoscience and Remote Sensing Letters* 17.*10* (Oct. 2020), Conference Name: IEEE Geoscience and Remote Sensing Letters, pp. 1742–1746, ISSN: 1558-0571, DOI: 10.1109/LGRS.2019.2955153.

[246] Xue and David A. Clausi, « Structure-preserving speckle reduction of SAR images using nonlocal means filters », en, *in*: *2009 16th IEEE International Conference on Image Processing (ICIP)*, Cairo, Egypt: IEEE, Nov. 2009, pp. 2985–2988, ISBN: 978-1-4244-5653-6, DOI: 10.1109/ICIP.2009.5414502, URL: http://ieeexplore.ieee.org/document/5414502/ (visited on 09/30/2021).

[247] Lingzi Xue, Xiaqing Yang, and Zongjie Cao, « Building Extraction of SAR Images Using Morphological Attribute Profiles », en, *in*: *Communications, Signal Processing, and Systems*, ed. by Qilian Liang et al., vol. 202, Series Title: Lecture Notes in Electrical Engineering, New York, NY: Springer New York, 2012, pp. 13–21, ISBN: 978-1-4614-5802-9 978-1-4614-5803-6, DOI: 10.1007/978-1-4614-5803-6_2, URL: http://link.springer.com/10.1007/978-1-4614-5803-6_2 (visited on 05/10/2021).

[248] Xufeng Han et al., « MatchNet: Unifying feature and metric learning for patch-based matching », en, *in*: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA: IEEE, June 2015, pp. 3279–3286, ISBN: 978-1-4673-6964-0, DOI: 10.1109/CVPR.2015.7298948, URL: http://ieeexplore.ieee.org/document/7298948/ (visited on 10/26/2021).

[249] Qi Yan et al., « CrossLoc: Scalable Aerial Localization Assisted by Multimodal Synthetic Data », *in*: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 17337–17347, DOI: 10.1109/CVPR52688.2022.01684.

[250] H. Yang, K. Lin, and C. Chen, « Supervised Learning of Semantics-Preserving Hash via Deep Convolutional Neural Networks », *in*: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 40.*02* (Feb. 2018), Place: Los Alamitos, CA, USA Publisher: IEEE Computer Society, pp. 437–451, ISSN: 1939-3539, DOI: 10.1109/TPAMI.2017.2666812.

[251] Yi Yang and Shawn Newsam, « Geographic Image Retrieval Using Local Invariant Features », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 51.*2* (Feb. 2013), Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 818–832, ISSN: 1558-0644, DOI: 10.1109/TGRS.2012.2205158.

[252] Famao Ye et al., « Remote Sensing Image Retrieval Using Convolutional Neural Network Features and Weighted Distance », *in*: *IEEE Geoscience and Remote Sensing Letters* 15.*10* (2018), pp. 1535–1539, DOI: 10.1109/LGRS.2018.2847303.

[253] Yuanxin Ye et al., « Fast and Robust Matching for Multimodal Remote Sensing Image Registration », en, *in*: *IEEE Transactions on Geoscience and Remote Sensing* 57.*11* (Nov. 2019), pp. 9059–9070, ISSN: 0196-2892, 1558-0644, DOI: 10.1109/TGRS.2019.2924684, URL: https://ieeexplore.ieee.org/document/8766118/ (visited on 07/18/2022).

[254] Yuanxin Ye et al., « Robust Optical-to-SAR Image Matching Based on Shape Properties », *in*: *IEEE Geoscience and Remote Sensing Letters* 14.*4* (2017), pp. 564–568, DOI: 10.1109/LGRS.2017.2660067.

[255] Yuanxin Ye et al., « Robust Registration of Multimodal Remote Sensing Images Based on Structural Similarity », *in*: *IEEE Transactions on Geoscience and Remote Sensing* 55.*5* (May 2017), pp. 2941–2958, ISSN: 0196-2892, 1558-0644, DOI: 10.1109/TGRS.2017.2656380.

[256] Michael Ying Yang, « Multiregion level-set segmentation of synthetic aperture radar images », *in*: *2009 16th IEEE International Conference on Image Processing (ICIP)*, ISSN: 2381-8549, Nov. 2009, pp. 1717–1720, DOI: 10.1109/ICIP.2009.5413378.

[257] Zhang Yongzhi et al., « Edge Extraction of Marine Oil Spill in SAR Images », *in*: *2010 International Conference on Challenges in Environmental Science and Computer Engineering*, vol. 1, Mar. 2010, pp. 439–442, DOI: 10.1109/CESCE.2010.200.

[258] J. C. Yoon et al., « KOMPSAT-5 SAR design and performance », *in*: *2011 3rd International Asia-Pacific Conference on Synthetic Aperture Radar (APSAR)*, 2011, pp. 1–4.

[259] R. Zabih and V. Kolmogorov, « Spatially coherent clustering using graph cuts », en, *in*: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.* Vol. 2, Washington, DC, USA: IEEE, 2004, pp. 437–444, ISBN: 978-0-7695-2158-9, DOI: 10.1109/CVPR.2004.1315196, URL: http://ieeexplore.ieee.org/document/1315196/ (visited on 05/10/2021).

[260] Sergey Zagoruyko and Nikos Komodakis, « Learning to Compare Image Patches via Convolutional Neural Networks », *in*: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[261] M.R. Zaman and C.R. Moloney, « A comparison of adaptive filters for edge-preserving smoothing of speckle noise », *in*: *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, ISSN: 1520-6149, Apr. 1993, 77–80 vol.5, DOI: 10.1109/ICASSP.1993.319751.

[262] Kai Zhang et al., « Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising », *in*: *IEEE Transactions on Image Processing* 26.*7* (July 2017), pp. 3142–3155, ISSN: 1057-7149, 1941-0042, DOI: 10.1109/TIP.2017.2662206.

[263] Qiang Zhang et al., « Learning a Dilated Residual Network for SAR Image Despeckling », en, *in*: *Remote Sensing* 10.*2* (Feb. 2018), Number: 2 Publisher: Multidisciplinary Digital Publishing Institute, p. 196, DOI: 10.3390/rs10020196, URL: https://www.mdpi.com/2072-4292/10/2/196 (visited on 08/25/2021).

[264] Qianqian Zhang et al., « A new road extraction method using Sentinel-1 SAR images based on the deep fully convolutional neural network », *in*: *European Journal of Remote Sensing* 52.*1* (Jan. 2019), Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/22797254.2019.1694447, pp. 572–582, ISSN: null, DOI: 10.1080/22797254.2019.1694447, URL: https://doi.org/10.1080/22797254.2019.1694447 (visited on 06/28/2021).

[265] Weixun Zhou et al., « Remote Sensing Image Retrieval in the Past Decade: Achievements, Challenges, and Future Directions », *in*: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 16 (2023), pp. 1447–1473, DOI: 10.1109/JSTARS.2023.3236662.

[266] Jun-Yan Zhu et al., « Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks », *in*: *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2242–2251, DOI: 10.1109/ICCV.2017.244.

[267] Sha Zhu, « A Bayesian Approach for Inverse Problems in Synthetic Aperture Radar Imaging », en, PhD thesis, Université Paris Sud - Paris XI, Oct. 2012, URL: https://theses.hal.science/tel-00844748 (visited on 12/14/2022).

[268] Huanxin Zou et al., « A Likelihood-Based SLIC Superpixel Algorithm for SAR Images Using Generalized Gamma Distribution », *in*: *Sensors* 16.7 (2016), ISSN: 1424-8220, DOI: 10.3390/s16071107, URL: https://www.mdpi.com/1424-8220/16/7/1107.

[269] Huanxin Zou et al., « A PDF-based SLIC superpixel algorithm for SAR images », *in*: *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2016, pp. 6823–6826, DOI: 10.1109/IGARSS.2016.7730781.

COLLEGES BRETAGNE LOIRE SCIENCES POUR L'INGENIEUR ET LE NUMERIQUE

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Titre : Navigation aérienne basée sur l'imagerie SAR et sur des données géospatiales de référence

Mot clés : navigation aérienne ; Radar à Synthèse d'Ouverture (RSO) ; fusion de données géospatiales ; régression de coordonnées de scène ; point proximal itératif

Résumé : Nous cherchons les moyens algorithmiques de déterminer l'état cinématique d'un appareil aérien à partir d'une image RSO observée et de données géospatiales de référence qui peuvent être RSO, optiques ou vectorielles. Nous déterminons la transformation qui associe les coordonnées de l'observation et les coordonnées de la référence et dont les paramètres sont l'état cinématique. Nous poursuivons trois approches. La première repose sur la détection et l'appariement de structures telles que des contours. Nous proposons un algorithme de type *Iterative Closest Point* (ICP) et démontrons comment il peut servir à estimer l'état cinématique complet. Nous proposons ensuite un système complet qui inclue un détecteur de contours multimodal appris. La seconde approche repote sur une métrique de similarité multimodale, ce qui est un moyen de mesurer la vraisemblance que deux restrictions locales de données géospatiales représentent le même point géographique. Nous déterminons l'état cinématique sous l'hypothèse duquel l'image SAR est la plus similaire aux données géospatiales de référence. La troisième approche repose sur la régression de coordonnées de scène. Nous prédisons les coordonnées géographiques de morceaux d'images et déduisons l'état cinématique à partir des correspondances ainsi prédites. Cependant, dans cette approche, nous ne satisfaisons pas l'hypothèse de multimodalité.

Title: Aerial Navigation Based on SAR Imaging and Reference Geospatial Data

Keywords: aerial navigation; Synthetic Aperture Radar (SAR); geospatial data fusion; Scene Coordinates Regression (SCR); Iterative Closest Point (ICP)

Abstract: We seek the algorithmic means of determining the kinematic state of an aerial device from an observation SAR image and reference geospatial data that may be SAR, optical or vector. We determine a transform that relates the observation and reference coordinates and whose parameters are the kinematic state. We follow three approaches. The first one is based on detecting and matching structures such as contours. We propose an iterative closest point algorithm and demonstrate how it can serve to estimate the full kinematic state. We then propose a complete pipeline that includes a learned multimodal contour detector. The second approach is based on a multimodal similarity metric, which is the means of measuring the likelihood that two local patches of geospatial data represent the same geographic point. We determine the kinematic state under the hypothesis of which the SAR image is most similar to the reference geospatial data. The third approach is based on scene coordinates regression. We predict the geographic coordinates of random image patches and infer the kinematic state from these predicted correspondences. However, in this approach, we do not address the fact that the modality of the observation and the reference are different.