Registration of egocentric views for collaborative localization in security applications

Huiqin Chen

université PARIS-SACLAY

Registration of egocentric views for collaborative localization in security applications

Alignement de vues pour la localisation collaborative dans des applications de sécurité

Thèse de doctorat de l'université Paris-Saclay

École doctorale n°580,
Sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat: Traitement du signal et des images
Unité de recherche : Université Paris-Saclay, ENS Paris-Saclay, CNRS, SATIE, 91190, Gif-sur-Yvette, France
Référent : Faculté des sciences d'Orsay

Thèse présentée et soutenue à Paris-Saclay, le 16 avril 2021, par

Huiqin CHEN

Composition du Jury

Arnaud Martin Professeur, Université de Rennes	Président
Catherine Achard Professeur, Sorbonne Université	Rapporteur & Examinatrice
Pascal Monasse Professeur, Ecole des Ponts ParisTech	Rapporteur & Examinateur
Michèle Gouiffès Maître de Conférences HDR, Université Paris-Saclay	Examinatrice
Antoine Manzanera Maître de Conférences HDR, ENSTA Paris	Examinateur

Direction de la thèse

Sylvie Le Hégarat-Mascle Professeur, Université Paris-Saclay	Directrice de thèse
Emanuel Aldea Maître de Conférences, Université Paris-Saclay	Co-Encadrant & Examinateur
Vincent Despiegel Responsable équipe de recherche, IDEMIA	Invité

Thèse de doctorat

NNT : 2021UPASG031

Abstract

This work focuses on collaborative localization between a mobile camera and a static camera for video surveillance. In crowd scenes and sensitive events, surveillance involves locating the wearer of the camera (typically a security officer) and also the events observed in the images (e.g., to guide emergency services). However, the different points of view between the mobile camera (at ground level), and the video surveillance camera (located high up), along with repetitive patterns and occlusions make difficult the tasks of calibration and localization.

We first studied how low-cost positioning and orientation sensors (GPS-IMU) could help refining the estimate of relative pose between cameras. We then proposed to locate the mobile camera using its epipole in the image of the static camera. To make this estimate robust with respect to outlier keypoint matches, we developed two algorithms: either based on a cumulative approach to derive an uncertainty map, or exploiting the belief function framework. Facing with the issue of a large number of elementary sources, some of which are incompatible, we provide a solution based on a belief clustering, in the perspective of further combination with other sources (such as pedestrian detectors and/or GPS data for our application). Finally, the individual location in the scene led us to the problem of data association between views. We proposed to use geometric descriptors/constraints, in addition to the usual appearance descriptors. We showed the relevance of this geometric information whether it is explicit, or learned using a neural network.

Résumé

Cette thèse s'intéresse à la localisation collaborative à partir d'une caméra mobile et d'une caméra statique pour des applications de vidéo-surveillance. Pour la surveillance d'évènements sensibles, la sécurité civile recours de plus en plus à des réseaux de caméras collaboratives incluant des caméras dynamiques et des caméras de surveillance traditionnelles, statiques. Il s'agit, dans des scènes de foules, de localiser d'une part le porteur de la caméra (typiquement agent de sécurité) mais également des évènements observés dans les images, afin de guider les secours par exemple. Cependant, les différences de point de vue entre la caméra mobile située au niveau du sol, et la caméra de vidéo-surveillance située en hauteur, couplées à la présence de motifs répétitifs et d'occlusions rendent les tâches de calibration et de localisation ardue.

Nous nous sommes d'abord intéressés à la façon dont des informations issues de capteurs de localisation et d'orientation (GPS-IMU) bas-coût, pouvaient contribuer à raffiner l'estimation de la pose relative entre les caméras. Nous avons ensuite proposé de localiser la caméra mobile par la localisation de son épipole dans l'image de la caméra statique. Pour rendre robuste cette estimation vis-à-vis de la présence d'outliers en termes d'appariements de points clés, nous avons développé deux algorithmes. Le premier est basé sur une approche cumulative pour construire la carte d'incertitude de l'épipole. Le second, qui exploite de cadre de la théorie des fonctions de croyance et de son extension aux cadres de discernement 2D, nous a permis de proposer une contribution à la gestion d'un grand nombre de sources élémentaires, dont certaines incompatibles, basée sur un clustering des fonctions de croyances, particulièrement intéressant en vue de la combinaison avec d'autres sources (comme les détecteurs de piétons et/ou données GPS pour notre application). Enfin, la dernière partie concernant la géolocalisation des individus dans la scène, nous a conduit à étudier le problème de l'association de données entre les vues. Nous avons proposé d'utiliser des descripteurs et contraintes géométriques, en plus des descripteurs d'apparence classiques, dans la fonction de coût d'association. Nous avons montré la pertinence de ces informations géométriques qu'elles soient explicites, ou apprises à l'aide un réseau de neurones.

Synthèse en Français

Cette thèse s'intéresse aux problèmes de localisation collaborative avec l'utilisation conjointe d'une caméra mobile et d'une caméra statique dans une application de surveillance. Un réseau de caméras collaboratives de caméras de vision dynamiques et égocentriques couplées aux caméras de surveillance traditionnelles dans des contextes à haut risque est devenu une avenue prometteuse pour faire progresser les tâches de sécurité publique et de sécurité publique, par fournir une localisation plus précise et une analyse plus fine des interactions individuelles. Dans les scènes urbaines encombrées, une photo prise par une caméra portable peut être utilisée par un agent des forces comme source supplémentaire d'informations pour la localisation lui-même ou la localisation de piétons cibles. Cependant, les différences variant entre les utilisateurs et la forme de surveillance entre la caméra mobile égocentrique au niveau du sol et la caméra de vidéosurveillance traditionnelle descendante soulèvent les défis pour leur intégration dans le système de surveillance public. En raison des changements à grande échelle et des changements de vue et de la présence fréquente de caractéristiques répétitives et d'occlusions, les tâches de localisation liées au porteur de la caméra et au piéton cible deviennent difficiles dans le réseau de caméras de la caméra égocentrique et de la caméra de vue statique dans les environnements urbains surpeuplés. L'objectif de cette étude est donc de fournir des solutions plus réalisables et plus fiables pour la géolocalisation d'agents équipées de caméras corporelles et la géolocalisation d'individuelles suspects à partir des images capturés à partir d'une caméra mobile et d'une caméra de surveillance statique.

D'abord, nous nous concentrons sur le problème de l'estimation de la pose relative qui est un problème proche de l'estimation de l'épipole en présence d'informations supplémentaires fournies par des capteurs de localisation et d'orientation de faible qualité. Une approche basée sur le M-estimateur offre une solution élégante pour la fusion entre les données inertielles et les données de vision, mais elle est sensible à l'importance préalable des correspondances visuelles entre les deux vues. En plus d'utiliser des indices extraits de la similarité visuelle locale, nous proposons de s'appuyer en même temps sur les associations apprises fournies par la cohérence géométrique globale. Un schéma de pondération conservateur pour combiner les deux types d'indices a été proposé et validé avec succès sur un ensemble de données urbaines.

Ensuite, nous travaillions sur la localisation de l'agent par la localisation de l'épipole et proposons une stratégie basée sur l'échantillonnage multimodal et le vote cumulé pour construire la carte d'incertitude de l'épipole afin de surmonter les défis soulevés par la présence de valeurs aberrantes parmi les points de correspondance. La méthode proposée est plus robuste en termes d'intégrité de la localisation de l'épipole que les approches standard. Nous proposons également un regroupement de fonctions de croyance et une fusion intra-groupe utilisant le cadre des fonctions de croyance 2D pour améliorer la précision et la fiabilité dans le cas d'un grand nombre de solutions, y compris les solutions aberrantes. L'algorithme proposé est plus robuste en termes d'exactitude et de précision que les approches standard qui fournissent des solutions singulières, notamment lorsqu'il est combiné à des sources supplémentaires comme les détecteurs de piétons et les données GPS.

Dernièrement, nous étudions le problème de l'association de données transversales pour la localisation des piétons. Nous proposons d'utiliser deux antécédents géométriques indépendants et de les intégrer avec les indices d'apparence classiques dans la fonction d'objection de l'algorithme d'association de données avec des règles de combinaison à la fois explicites et basées

sur l'apprentissage. Nos résultats montrent que la méthode proposée apporte une amélioration significative en termes de précision de l'association. Nous soulignons l'utilisation intéressante des indices géométriques dans l'association de données de vues croisées et son potentiel pour soutenir le suivi des piétons dans ce contexte. Nous suggérons d'utiliser la combinaison basée sur l'apprentissage pour les scores d'association a dans les scènes pour lesquelles une corrélation plus forte existe entre différentes caractéristiques lorsqu'un grand ensemble de données est disponible.

Introduction

Context

Video surveillance has been widely used in many countries as an important tool for safety and security in both private and public occasions. Today around 770 million surveillance cameras have been installed worldwide and one billion will be installed in 2021, being distributed in the United States, China, Germany and the United Kingdom. In terms of the total number of surveillance cameras and the number of surveillance cameras per 100 individuals as reported in Figure 1, the reported ratio in France is 2.46, far behind other countries, despite the concern of terrorism. Most of the worries are about the invasion of privacy and the potential institutional or personal abuse. Multiples bodies at national and European level regulate and enforce limitations on the use of CCTV systems. At the same time, there is often support from the public (the typical example being the Nice area) and from the research community for various benefits of the surveillance cameras. It was reported in [70] and [157] that camera surveillance can reduce the chance of undesirable behaviors and it is less likely for people to commit crimes and to harm others in the areas where their actions are being recorded. The surveillance cameras also provide a crucial piece of evidence and help the police trace the suspects' movements and identify criminals in the post-investigation.

However, the effectiveness of camera surveillance for crime prevention is limited to its recording functionality, and the actual exploitation of recorded data unleashes its true potential. When criminal activities happen, the surveillance cameras play a passive role, and typically a significant amount of human supervision is required at some point to intervene in the post-processing step. This passive search manner can lead to an important delay in dealing with emergency cases and results in the failure to prevent further crimes. With the development of deep learning and AI, more advanced and intelligent camera surveillance systems have been built upon facial recognition and automatic video content analysis. Different functions, including data recording, processing and decision making, are integrated into a single system on the cloud and an alert is sent to users when abnormal activities are detected. Such an intelligent surveillance system has been integrated into the concept of smart homes and cities for private and public use. It is also beneficial for monitoring the flow of crowds when human intervention becomes extremely difficult, for example in the case of crowds where the stampede can be avoided with the early invention of law enforcement when the abnormal areas are detected [54, 75, 135, 170].

In recent years, the camera surveillance systems start to consider progressively the mobile cameras as well, as shown in Figure 2. More and more countries, including France have started to equip their law enforcement with body-worn cameras since the first testing of body-worn cameras in the United Kingdom. It was reported in [21] that the police officers wearing cameras received fewer complaints and use less force than the officers without cameras. One of reasons may be that the presence of body-worn cameras on police potentially increases law enforcement's professionalism and makes citizens avoid the violence in the first place. Besides, the use of mobile cameras is not restricted to law enforcement. With the emergence of self-media and the widespread use of live-broadcast platforms, there are more and more available recordings in the public area from the citizens. These recordings make citizens involved in public surveillance with an active role. In some cases, they can provide valuable information as a witness when the recordings are not available for CCTV cameras and body-worn cameras of law enforcement. Compared to CCTV cameras,

Country	# of CCTV Cameras	# of People	# of CCTV Cameras per 100 People
United States	50 000 000	327,167,430	15.28
China	200 000 000	1,392,730,000	14.36
United Kingdom	5 000 000	66,488,990	7.5
Germany	5 200 000	82,927,920	6.27
Netherlands	1 000 000	17,231,020	5.80
Australia	1 000 000	24,992,370	4
Japan	5 000 000	126,529,100	3.95
Vietnam	2 600 000	95,540,400	2.72
France	1 650 000	66,987,240	2.46
South Korea	1 030 000	51,635,260	1.99

Figure 1: Top Ten countries on the number of surveillance cameras per 100 people[4].

(a) Body-born camera for law enforcement

(b) Mobile phone for citizens

(c) Drone

Figure 2: Egocentric mobile cameras for surveillance.

mobile cameras can provide many more details and higher resolution for individual targets, especially when the scene is very crowded.

While both fixed CCTV cameras and mobile cameras can provide useful recordings for the public spaces as surveillance devices, they can play different roles based on their specific strong points. Therefore, a collaborative camera networks of dynamic, egocentric view cameras coupled with the traditional overview surveillance cameras in high-risk contexts has become a promising avenue for advancing public safety and security applications. By offering a significant advantage due to their mobility and to the higher level of detail, the egocentric mobile cameras can provide dynamic viewpoints and finer analysis of individual interactions from the ground level, which enrich the context available using the static/third-view surveillance cameras for a significant number of problems related to direct risk assessment (i.e. detection of aggressive behavior, detection of phenomena that indicate danger for the crowd such as extreme density, tracking and identification of individuals who may represent a threat to others).

However, it is challenging to link effectively the mobile camera with the traditional CCTV camera due to their differences varying from the users to the form of surveillance. CCTV cameras are usually operated by the government or institutional agents and the mobile cameras can be used by law enforcement or individual citizens. The communication between different users is very often complicated and strongly regulated. In terms of the form of surveillance, the CCTV cameras are usually installed in a high place and provide a top-down form of recording, while the mobile cameras are held or worn by persons and provide a bottom-up form of images from the ground level. When applying the video content analysis or facial recognition in an advanced intelligent surveillance system, finding the correspondences between their images would be challenging due to the large scale and view changes.

To enhance the integration of mobile camera and fixed CCTV camera in the intelligent surveillance system for the safety and security of mass gatherings in complex scenarios and widespread urban environments, our PhD work has been developed along the "S^2UCRE" project within the French-German funding framework "Safety and Security of Urban Crowded Environments", which provided at the same time an opportunity to interact and collaborate with other academic and industrial partners. The objective is to develop an integrated surveillance system for different

sources to help law enforcement respond quickly when facing accidents, offensive behaviors or panic situations. The involved technology domains include:

1. Video-based crowd monitoring of distributed urban environments: crowd densities and crowd dynamics;
2. Short-term prediction of crowd behavior for fast and efficient evacuation;
3. Semi-automated suspicious behavior analysis for security applications;
4. Detection & (geo-) localization of perpetrators;
5. Self-localization of security and rescue team members and a communication platform for geo-registered information exchange between security and rescue staff.

Specifically, our efforts focus on the study of collaborative localization between the mobile camera and the static camera, related to the last two parts for geo-localization and geo-registration.

The camera based localization is involved in different surveillance tasks such as pedestrian tracking. It also plays a critical role in the environment perception, recognition and navigation in augmented reality, robotics and autonomous driving. Depending on camera configuration and the available scenes, the image-based localization can be divided into different sub-tasks, such as computing the camera pose (orientation and position) with respect to the world coordinate system for single or multiple camera networks or computing the relative camera pose for the stereo cameras system. Although it has been well studied in the community of computer vision, the camera localization for the mobile camera and a top-view static camera raises specific problems due to the large scale changes and view changes between their captured images. When it comes to the urban crowded environments, the task becomes more challenging due to the frequent presence of repetitive features and occlusions. To guarantee the efficiency of an integrated intelligent surveillance system between mobile cameras and static cameras, we require more accurate and robust algorithms for the localization task.

Outline of work

In this work, we deal with the localization problem between a mobile camera with egocentric view and a static camera. We aim to provide more feasible and reliable solutions for the geo-localization of agents equipped with body-worn cameras and the geo-localization of suspicious individuals from the images captured from a mobile camera and a static surveillance camera.

We develop a conceptual framework to apply the proposed algorithms for the surveillance tasks as illustrated in Figure 3. The scenario is designed for the agent equipped with a mobile camera moving into a region covered by a public static surveillance camera. Given an image from the mobile camera view, a prerequisite step which relies on the Video Management System (VMS) used by the CCTV operator is to find the corresponding frame from the video stream of the static camera through a synchronization mechanism. The proposed algorithms for the epipole localization are applied on the pair of images to estimate the 2D location of an agent in the view of the static camera, which can be transformed into 3D coordinates with the help of the geo-information related to the static camera. When a suspicious person is identified in the view of the mobile camera, the proposed association algorithm is used to find the corresponding detection of the target person in the static camera view. The 3D location of the target is then determined by the triangulation of the pair of 2D detections from different views. When additional pose sensors are available, the proposed algorithm for improvement of relative camera pose can be applied to enhance the accuracy of both agent localization and pedestrian localization.

The first part (Part II) deals with the relative pose estimation between the static camera and the mobile camera, which is involved in both the epipole estimation for the agent localization or the data association problem for the pedestrian localization. The standard image-based method

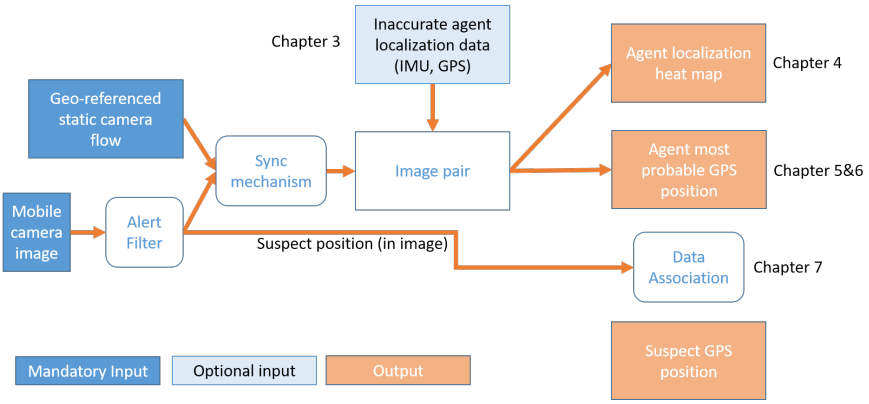


Figure 3: Overview of the global localization framework

for estimating the relative camera pose derives from correspondences of points from image to object or image to image, which are prone to be contaminated, especially when there is large scale change and view change and repetitive features in urban crowded environments. Relying on the visual features is not enough to overcome these challenges due to feature ambiguities. When the additional pose sensors are available, even though they may be noisy and inaccurate, they may still guide image-based estimation by integrating the priors as the regularization term into the optimization framework. For this purpose, we investigate the existing fusion framework between visual features and pose priors and propose to explore the combination of visual cues and the global geometrical coherence to improve the accuracy of relative camera pose in Chapter 3.

The second part of our work is devoted to the localization of an agent which refers to determining the position of agent in the 3D world coordinate system (see Part III). Although a position sensor such as GPS may be used to localize the agent, it suffers from low precision, especially in the urban environments where the GPS may be not available or severely perturbed. When an agent is equipped with a body-worn camera, the mobile views can help localize the agent with the help of the static surveillance views. As the 2D location of the camera wearer in the view of static camera is identified as the projection of mobile camera center, which is known as epipole in epipolar geometry between two views, the 3D localization of agent can be derived by the back-projection of the 2D location of epipole in the view of the static surveillance views into the 3D world coordinates according to the camera projection rule with the help of additional ground plane constraints. The epipole-based agent localization is feasible and light as it only requires a pair of two synchronized images from the mobile camera and the static camera. However, the epipole estimation may be unstable and unreliable due to the error introduced in the correspondences of the local visual features. To address this problem, we investigate the uncertainty estimation of epipole instead of single position which could be wrong and misleading. We first develop a multi-modal sampling strategy to increase the reliability for the confidence region of epipole compared to the standard propagation pipeline in Chapter 4. We then explore the fusion of multiple sources with a 2D belief function framework to improve the accuracy and precision for the epipole estimation (see Chapter 5 and Chapter 6).

In the last part (see Part IV), we work on the geo-localization of suspicious individuals which refers to the localization of individual pedestrians in the 3D world coordinate frame. When a target object is identified from the view of the agent, the 3D location of the object can be deduced by the triangulation of its projections on the 2D image plane for the mobile camera and the static camera, which requires a correct association of 2D detection of the object across different views. However, due to the strong scene scale changes and occlusions between the mobile egocentric camera on the ground level and the static camera at the top position, the association method relying on the visual features may not always be reliable. More robust and accurate methods are required to guarantee the efficiency of the data association. For this reason, we explore the use of geometric priors based on pedestrians' location and the camera calibration priors to improve the

performance of association based on the appearance-based features in Chapter 7. We investigate both explicit rule and the data-driven combination for the rule to combine the geometric cue and visual cues.

Both agent localization and the pedestrian localization tasks require a preliminary step for the geo-registration of a static camera to transform the 2D position to 3D coordinate in the real world. An easy way is to apply the algorithm PnP for 2D-3D correspondences by manually choosing 3D geo-referenced landmarks. As the camera is fixed, this calibration only needs to be done once. In this work, we consider the geo-registration of static camera as a well-built task and do not investigate it on the research level. We will present the calibration process for static camera and the transformation of 2D location to the 3D position in geographic coordinate system in Appendix.

Publications

The publications during this PhD work are listed as follows:

- Camera Localization based on Belief Clustering, Huiqin Chen, Sylvie Le Hégarat-Mascleand Emanuel AldeaProceedings of the 23rd International Conference on Information Fusion (FUSION), 2020 (see Chapter 6)
- Use of Scene Geometry Priors for Data Association in Egocentric Views, Huiqin Chen, Emanuel Aldea, Sylvie Le Hégarat-Mascleand Vincent Despiegel,Proceedings of the 8th International Workshop on Biometrics and Forensics (IWBF2020), 2020 (see Chapter 7)
- Determining Epipole Location Integrity by Multimodal Sampling, Huiqin Chen, Emanuel Aldea and Sylvie Le Hégarat-Mascle,Proceedings of the 16th IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS), The 3th International Workshop on Traffic and Street Surveillance for Safety and Security (IWT4S), 2019 (see Chapter 4)
- Integrating Visual and Geometric Consistency for Pose Estimation, Huiqin Chen, Emanuel Aldea and Sylvie Le Hégarat-Mascle,Proceedings of the 16th International Conference on Machine Vision Applications (MVA), 2019 (see Chapter 3)
- Belief Functions Clustering for Epipole Localization, Huiqin Chen, Sylvie Le Hégarat-Mascle and Emanuel Aldea, *submitted to International Journal of Approximate Reasoning, review ongoing*

Contents

List of Figures

List of Tables

Glossary

DoF - *degree of freedom*

SIFT - *scale invariant feature transform*

SSD - *sum squared difference*

CNN - *convolutional neural network*

RANSAC - *Random Sample Consensus*

SFM - *structure from motion*

ENU - *east-north-up*

GPS - *global positioning system*

MEMS - *microelectromechanical systems*

MLP - *multilayer perceptron*

LEA - *law enforcement agent*

SVD - *singular value decomposition*

MC - *monte-carlo simulation*

BBA - *basic belief assignment*

GSSF - *generalized simple support functions*

DF - *discernment frame*

DA - *data association*

Part I

Background

Chapter 1

Overview of multiple view geometry

Contents

Multiple view geometry is an important tool for dealing with the problems arising in a network of multiple cameras, for example a collaborative surveillance network with mobile and static surveillance cameras as in our case. This formalism describes the geometric relationship between the 3D world and the image plane, scene structure and cameras motion. In this chapter, we introduce the camera geometry and the epipolar geometry between two views presented in [60]. For the notations, we use the bold letter to represent vector and the normal letter for matrix and scalar value.

1.1 Coordinates representation

In the multiple view geometry, the coordinates for points, lines and planes are represented by the homogeneous coordinates. Compared to the Euclidean representation, the homogeneous coordinates can represent the infinity points using finite coordinates.

Definition 1.1.1. *The **homogeneous coordinates for a finite point** in Euclidean space $\hat{\mathbf{x}} \in \Re^N$, are defined as*

$$\mathbf{x} = (\hat{\mathbf{x}}^T, 1)^T = k(\hat{\mathbf{x}}^T, 1)^T,$$

where the scale $k \in \Re \backslash \{0\}$ can be ignored.

Definition 1.1.2. *The **homogeneous coordinates for an infinite point** in Euclidean space $\Re^N$ are defined as*

$$\mathbf{x} = (\hat{\mathbf{x}}^T, 0)^T.$$

Definition 1.1.3. *The **homogeneous coordinates for the line in Euclidean plane** satisfying $ax + by + c = 0$, are defined as*

$$\mathbf{l} = (a, b, c)^{\mathrm{T}} = k(a, b, c)^{\mathrm{T}},$$

where the scale k is ignored. The line at infinity is denoted by $\mathbf{l}_\infty = (0, 0, 1)^{\mathrm{T}}$. The definition of line in 2D spaces can be easily extended to a higher dimension space.

Definition 1.1.4. *A point $\mathbf{x}$ lies on the line $\mathbf{l}$ if and only if $\mathbf{x}^{\mathrm{T}}\mathbf{l} = 0$.*

Definition 1.1.5. *The **intersection between two lines** $\mathbf{l}$ and $\mathbf{l}'$ is computed as $\mathbf{x} = \mathbf{l} \times \mathbf{l}'$.*

Definition 1.1.6. *The **degree of freedom (DoF)** for a homogeneous vector $\mathbf{x}$ is $dim(\mathbf{x}) - 1$ as it is up to scale.*

1.2 Camera geometry

The camera model describes the mapping between the 3D real world and a 2D image. This mapping can be divided into the mapping between the world coordinate frame and the camera coordinate frame, and the mapping between the camera coordinate frame to the image plane. The mapping between the camera coordinate frame and the world coordinate frame is related by a rotation and a translation, as shown in Figure 1.1. In the following, we present the basic notations related to the camera geometry. We use $\mathbf{X}$ to denote homogeneous coordinates of the point in 3D and $\mathbf{x}$ for the point in 2D.

Given a 3D point in the world coordinate frame with the homogeneous coordinates $\mathbf{X} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 1)^{\mathrm{T}}$, the coordinates for the transformed point in the camera coordinate frame is

$$\mathbf{X_{cam}} = (\mathrm{X}', \mathrm{Y}', \mathrm{Z}')^{\mathrm{T}} = \mathrm{R}((\mathrm{X}, \mathrm{Y}, \mathrm{Z})^{\mathrm{T}} - \mathbf{C}) = [\mathrm{R} \mid -\mathrm{R}\mathbf{C}]\mathbf{X} = [\mathrm{R} \mid \mathbf{t}]\mathbf{X}, \tag{1.1}$$

where R is the rotation matrix from the world to the camera, $\mathbf{C}$ is the camera center in the world coordinate frame, and the translation vector $\mathbf{t} = -\mathrm{R}\mathbf{C}$. The rotation matrix R and the translation vector $\mathbf{t}$ are known as the extrinsic parameters of the camera.

The camera coordinate frame is defined as an Euclidean coordinate system with the origin at the centre of camera $\mathbf{C}$ and axis $\mathbf{Z}$ as the principle axis of camera, where plane $\mathrm{Z} = f$ is identified as the image plane or focal plane, and f is the *focal length* of camera. The intersection of the principal axis $\mathbf{Z}$ with the image plane is known as the *principal point*, which is also the projection of *camera center*. With the ideal pinhole camera model and similar triangle rules as shown in Figure 1.2, the projection of a point in the camera coordinate frame with the coordinates $\mathbf{X_{cam}} = (\mathrm{X}', \mathrm{Y}', \mathrm{Z}')^{\mathrm{T}}$ on the 2D image plane can be expressed with the following 2D homogeneous coordinates:

$$\mathbf{x} = (\frac{f\mathrm{X}'}{\mathrm{Z}'}, \frac{f\mathrm{Y}'}{\mathrm{Z}'}, 1)^{\mathrm{T}}, \tag{1.2}$$

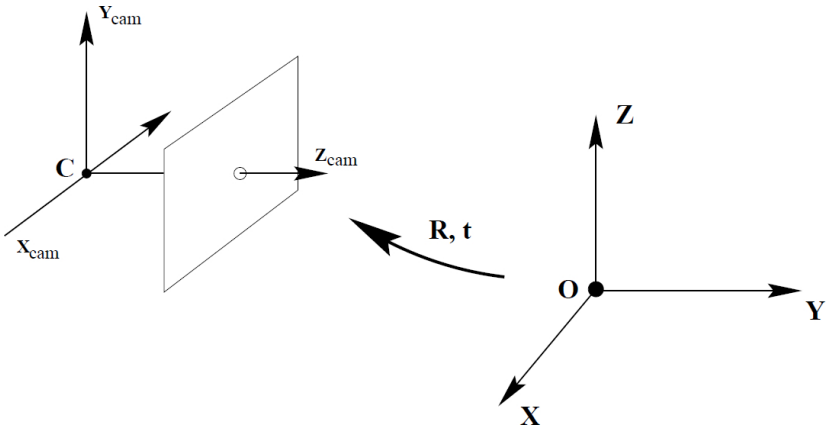


Figure 1.1: Euclidean transformation from the world to camera coordinate frame.

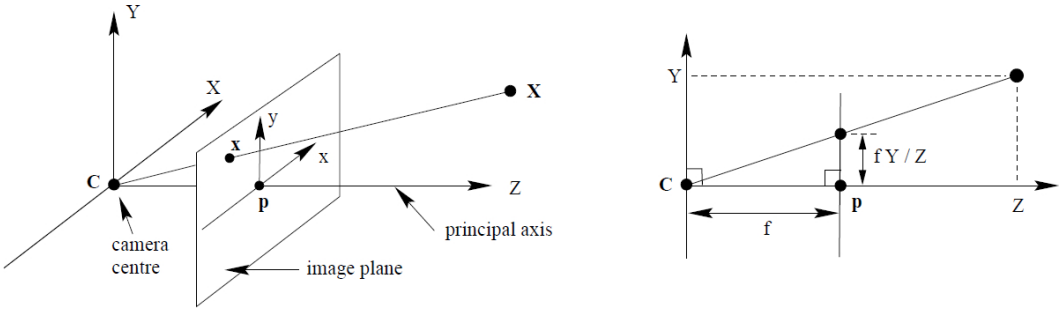


Figure 1.2: Pinhole camera model and the similar triangle rules for perspective projection from camera coordinate frame to image plane.

where the origin in the image plane is at the principal point with the coordinate $\mathbf{p} = (c_x, c_y)$ expressed in the general reference. By shifting the principal point offset, the projection of 3D point $\mathbf{X}$ in the image plane in general is

$$\mathbf{x} = (\frac{f\mathrm{X}'}{\mathrm{Z}'} + c_x, \frac{f\mathrm{Y}'}{\mathrm{Z}'} + c_y, 1)^{\mathrm{T}} = \frac{1}{\mathrm{Z}'}(f\mathrm{X}' + c_x\mathrm{Z}', f\mathrm{Y}' + c_y\mathrm{Z}', \mathrm{Z}')^{\mathrm{T}}. \tag{1.3}$$

It can be organized in the following matrix form:

$$\lambda\mathbf{x} = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathrm{X}' \\ \mathrm{Y}' \\ \mathrm{Z}' \end{bmatrix} = \mathrm{K}\mathbf{X}_{\mathbf{cam}}, \tag{1.4}$$

where $\lambda = \mathrm{Z}'$ is the scale, and K is the *camera calibration matrix* for the intrinsic parameters of camera in the case of the ideal pinhole camera model. The camera calibration matrix may have a more complex form depending on the specific camera model.

By replacing $\mathbf{x}_{\mathbf{cam}}$ in Equation (1.4) with Equation (1.1), we have the mapping from the 3D point in the real world to the pixel position in image plane:

$$\lambda\mathbf{x} = \mathrm{K}[\mathrm{R} \mid \mathbf{t}]\mathbf{X} = \mathrm{P}\mathbf{X}, \tag{1.5}$$

which is known as the projection model for the pinhole camera, and P is called the *projection matrix*.

1.3 Epipolar geometry

As illustrated in Figure 1.3, the epipolar geometry describes the projective geometry between two views from cameras, which only depends on the cameras' intrinsic parameters and the relative pose between two cameras. The algebraic representation for the epipolar geometry is known as the fundamental matrix and the essential matrix derived from correspondences of image points without knowing the scene structure. We introduce the necessary notations in epipolar geometry in the following.

Given the projections on the image planes of two cameras, denoted by $\mathbf{x_1}$ and $\mathbf{x_2}$ for a point $\mathbf{X} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 1)^{\mathrm{T}}$ in the world coordinate frame, the projection rays from camera centers corresponding to $\mathbf{x_1}$ and $\mathbf{x_2}$ can be denoted by the vectors $\overrightarrow{\mathbf{C_1X}}$ and $\overrightarrow{\mathbf{C_2X}}$, where $\mathbf{C_1}$ and $\mathbf{C_2}$ are the centers of the left and right cameras. The plane passing through the line $\overrightarrow{\mathbf{C_1X}}$, $\overrightarrow{\mathbf{C_2X}}$ and $\overleftrightarrow{\mathbf{C_1C_2}}$ is called the *epipolar plane*, including the *baseline* $\overleftrightarrow{\mathbf{C_1C_2}}$. The intersection of the baseline with the image plane is called *epipole*, which is the projection of one camera center on the view of other camera. By inverting the projective camera model in Equation (1.5), and the Euclidean transformation in Equation (1.1), the projection ray is expressed as

$$\overrightarrow{\mathbf{C_1X}} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z})^{\mathrm{T}} - \mathbf{C_1} = (\lambda_1 \mathrm{R}_1^{\mathrm{T}} \mathrm{K}_1^{-1}\mathbf{x_1} + \mathbf{C_1}) - \mathbf{C_1} = \lambda_1 \mathrm{R}_1^{\mathrm{T}} \mathrm{K}_1^{-1}\mathbf{x_1}, \tag{1.6}$$

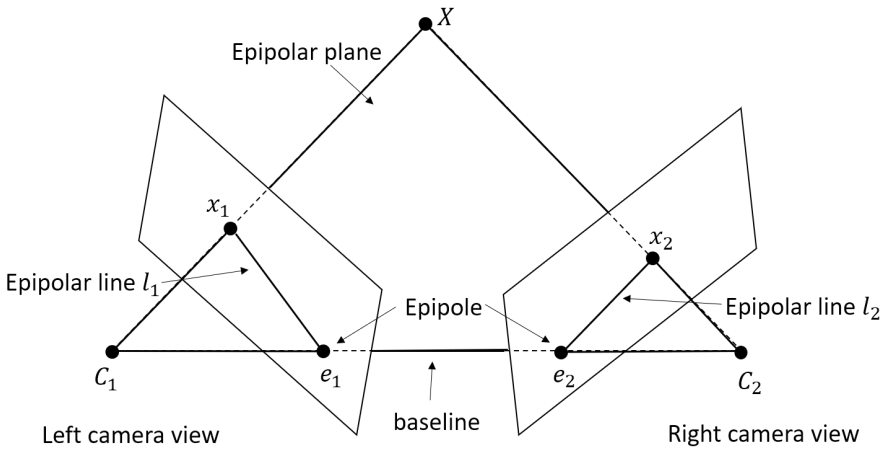


Figure 1.3: Epipolar geometry between two cameras.

and the similar expression for $\overrightarrow{\mathbf{C_2X}} = \lambda_2 \mathrm{R}_2^{\mathrm{T}} \mathrm{K}_2^{-1} \mathbf{x_2}$, where K_1 and K_2 are the camera calibration matrix for left and right cameras, λ_1, λ_2 are scale parameters, and R_1, R_2 are the rotation matrix for the left and right camera with respect to the world coordinate frame. As the vectors $\overrightarrow{\mathbf{C_1X}}$, $\overrightarrow{\mathbf{C_2X}}$ and $\overrightarrow{\mathbf{C_1C_2}}$ are co-planar, we have

$$\overrightarrow{\mathbf{C_2X}} \cdot (\overrightarrow{\mathbf{C_1C_2}} \times \overrightarrow{\mathbf{C_1X}}) = 0,$$

where '$\cdot$' presents the inner product between two vectors and '$\times$' presents the cross product between two vectors. By replacing $\overrightarrow{\mathbf{C_1X}}$ and $\overrightarrow{\mathbf{C_2X}}$, the epipolar constraint is described as

$$\lambda_2 (\mathrm{R}_2^{\mathrm{T}} \mathrm{K}_2^{-1} \mathbf{x_2})^{\mathrm{T}} [\overrightarrow{\mathbf{C_1C_2}}]_\times \lambda_1 \mathrm{R}_1^{\mathrm{T}} \mathrm{K}_1^{-1} \mathbf{x_1} = \mathbf{x_2}^{\mathrm{T}} \mathrm{K}_2^{-\mathrm{T}} \mathrm{R}_2 [\overrightarrow{\mathbf{C_1C_2}}]_\times \mathrm{R}_1^{\mathrm{T}} \mathrm{K}_1^{-1} \mathbf{x_1} = 0, \tag{1.7}$$

where $[\overrightarrow{\mathbf{C_1C_2}}]_\times$ is the skew-symmetric matrix related to the line vector $\overrightarrow{\mathbf{C_1C_2}} = (a, b, c)^{\mathrm{T}}$, defined as

$$[\overrightarrow{\mathbf{C_1C_2}}]_\times = \begin{bmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{bmatrix}.$$

This property can be represented by the fundamental matrix or the essential matrix depending on the knowledge of the camera calibration matrix K.

1.3.1 The Fundamental matrix

Definition 1.3.1. *The **Fundamental matrix** is defined as*

$$\mathrm{F} = \mathrm{K}_2^{-\mathrm{T}} \mathrm{R}_2 [\overrightarrow{\mathbf{C_1C_2}}]_\times \mathrm{R}_1^{\mathrm{T}} \mathrm{K}_1^{-1}.$$

We may write the Equation (1.7) with the Fundamental matrix. The fundamental matrix describes the geometric relation between the correspondences of image points $\mathbf{x_1}$ and $\mathbf{x_2}$. It has the following essential properties:

1. For a pair of correspondences $(\mathbf{x_1}, \mathbf{x_2})$, it gives

$$\mathbf{x_2}^{\mathrm{T}} \mathrm{F} \mathbf{x_1} = 0. \tag{1.8}$$

2. The rank of the fundamental matrix equals two, and the DoF is 7.

3. **Epipolar lines:** For $\mathbf{x_1}$, its corresponding point $\mathbf{x_2}$ is located on the *epipolar line* $\mathbf{l} = \mathrm{F}\mathbf{x_1}$, as $\mathbf{x_2}^{\mathrm{T}} \mathbf{l} = 0$, respective the epipolar line $\mathbf{l'} = \mathrm{F}^{\mathrm{T}} \mathbf{x_2}$ for $\mathbf{x_1}$. This relation can be used to constraint the search of correspondence on its epipolar line.

4. **Epipole:** The epipolar lines for all points other than epipole intersects at epipole. The epipole satisfies $\mathbf{e_2}^{\mathrm{T}}\mathrm{F} = 0$ and $\mathrm{F}\mathbf{e_1} = 0$, where $\mathbf{e_1}$ is the epipole in the first image and $\mathbf{e_2}$ is the epipole in the second image.

1.3.2 The Essential matrix

If the camera calibration matrix is known, we may also write Equation (1.7) with the essential matrix for the normalized coordinates.

Definition 1.3.2. *The **Essential matrix** is defined as*

$$\mathrm{E} = \mathrm{R}_2[\overrightarrow{\mathbf{C_1}\mathbf{C_2}}]_\times \mathrm{R}_1^{\mathrm{T}},$$

such that

$$\mathbf{x_2}^{\mathrm{T}}\mathrm{K}_2^{-\mathrm{T}}\mathrm{E}\mathrm{K}_1^{-1}\mathbf{x_1} = \hat{\mathbf{x}}_\mathbf{2}^{\mathrm{T}}\mathrm{E}\hat{\mathbf{x}}_\mathbf{1} = 0, \tag{1.9}$$

where $\hat{\mathbf{x}}_\mathbf{1} = \mathrm{K}_1^{-1}\mathbf{x_1}$ *and* $\hat{\mathbf{x}}_\mathbf{2} = \mathrm{K}_2^{-1}\mathbf{x_2}$ *are the normalized coordinates for a pair of image points.*

The essential matrix describes the geometric constraint between the correspondences of normalized image points. It is a rank 2 matrix with two equal singular values.

By considering the left camera coordinate frame as the world coordinate frame, and the Euclidean transformation from left camera to right camera is represented by the relative rotation matrix R and translation $\mathbf{t}$ between the left camera and right camera, we may write the fundamental matrix and the essential matrix as

$$\mathrm{F} = \mathrm{K}_2^{-\mathrm{T}}\mathrm{R}[\mathrm{R}^{\mathrm{T}}\mathbf{t}]_\times \mathrm{K}_1^{-1}, \quad \mathrm{E} = \mathrm{R}[\mathrm{R}^{\mathrm{T}}\mathbf{t}]_\times = [\mathbf{t}]_\times \mathrm{R}, \tag{1.10}$$

with $\overrightarrow{\mathbf{C_1}\mathbf{C_2}} = -\mathrm{R}^{\mathrm{T}}\mathbf{t}$, and the sign can be ignored as F and E are determined up to scale. So the essential matrix and the fundamental matrix can be derived from each other as follows:

$$\mathrm{E} = \mathrm{K}_2{}^{\mathrm{T}}\mathrm{F}\mathrm{K}_1, \quad \text{and} \quad \mathrm{F} = \mathrm{K}_2^{-\mathrm{T}}\mathrm{E}\mathrm{K}_1^{-1}. \tag{1.11}$$

1.3.3 Compute the fundamental/essential matrix

The fundamental matrix can be computed with the correspondence of image points using 8-point algorithm [62]. Given a pair of matching points $\mathbf{x_1} \leftrightarrow \mathbf{x_2}$ in two images, with $\mathbf{x_1} = (x_1, y_1, 1)^{\mathrm{T}}$ and $\mathbf{x_2} = (x_2, y_2, 1)^{\mathrm{T}}$, the definition of fundamental matrix gives an equation:

$$x_2x_1f_{11} + x_2y_1f_{12} + x_2f_{13} + y_2x_1f_{21} + y_2y_1f_{22} + y_2f_{23} + x_1f_{31} + y_1f_{32} + f_{33} = 0,$$

where f_{ij} is the coefficient of the fundamental matrix F. Given n pairs of matching points, we may write

$$\mathrm{A}\mathbf{f} = 0, \tag{1.12}$$

where A is the stack of $[x_2^ix_1^i,\ x_2^iy_1^i,\ x_2^i,\ y_2^ix_1^i,\ y_2^iy_1^i,\ y_2^i,\ x_1^i,\ y_1^i,\ 1]$ with $1 \le i \le n$, and $\mathbf{f}$ is the vector of coefficients for the fundamental matrix. As the number of unknown coefficients is 9 and the fundamental matrix is determined up to scale, it requires at least $n = 8$ pairs of points to find the fundamental matrix. When $n = 8$, there is a unique solution for $\mathbf{f}$ by solving the linear equation. When $n > 8$, $\mathbf{f}$ is over-determined and can be computed as the singular vector with the smallest singular value by applying SVD on A, with the constraint of $\| \mathbf{f} \| = 1$.

According to the definition of F, the rank of F equals to 2. To make the computed F satisfy this constraint, we can decompose the computed F by SVD, where $\mathrm{F} = \mathrm{U}\mathrm{diag}(s_1, s_2, s_3)\mathrm{V}^{\mathrm{T}}$. By replacing s_3 with 0, the new fundamental matrix is computed as $\mathrm{F}' = \mathrm{U}\mathrm{diag}(s_1, s_2, 0)\mathrm{V}^{\mathrm{T}}$.

The above method can also be used for the essential matrix estimation. As the essential matrix depends only on the 6DoF (degree of freedom) and is up to scale, the DoF for the essential matrix is reduced to 5. An efficient method based on 5-point algorithm can be found in [109].

1.3.4 Extraction of camera pose

The relative camera pose can be extracted from the essential matrix or the fundamental matrix if the camera calibration matrix K_1 and K_2 are known. As the essential matrix is the product of a skew-symmetric matrix and a rotation (orthogonal) matrix, the essential matrix shares the same properties related to singular values with the skew-symmetric matrix, which has two equal singular values with rank=2. Thus, the essential matrix can be decomposed with the following singular value decomposition(SVD):

$$E = U\text{diag}(1,1,0)V^T \quad \text{or} \quad E = U\text{diag}(-1,-1,0)V^T$$

where U and V are orthogonal. Using an orthogonal matrix W and a skew-symmetric matrix Z with the following definition:

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad Z = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

the essential matrix can be represented by

$$E = [\mathbf{t}]_\times R = UZWV^T = (UZU^T)(UWV^T) \quad \text{or} \quad E = [\mathbf{t}]_\times R = UZW^TV^T = (UZU^T)(UW^TV^T),$$

with $Z = \text{diag}(1,1,0)W^T$ and $Z = \text{diag}(-1,-1,0)W$. As W, U and V are orthogonal, and Z is the skew-symmetric, we can deduce that

$$[\mathbf{t}]_\times = UZU^T \quad \text{and} \quad R = UWV^T \quad \text{or} \quad R = UW^TV^T. \tag{1.13}$$

For the relative translation $\mathbf{t}$, we have $[\mathbf{t}]_\times \mathbf{t} = 0$ and $[\mathbf{t}]_\times = U\text{diag}(1,1,0)(UW)^T\mathbf{t}$, with $\|\mathbf{t}\| = 1$. Thus $\mathbf{t}$ can be computed as the singular vector with zero singular value and $\mathbf{t} = \pm U(0,0,1)^T = \pm\mathbf{u_3}$. In total, there are four possible solutions for the relative camera pose $[R \mid \mathbf{t}]$:

$$[UWV^T \mid \mathbf{u_3}] \quad \text{or} \quad [UW^TV^T \mid \mathbf{u_3}] \quad \text{or} \quad [UWV^T \mid -\mathbf{u_3}] \quad \text{or} \quad [UW^TV^T \mid -\mathbf{u_3}]. \tag{1.14}$$

The unique solution is determined by testing if a point is in front of both cameras.

1.4 Keypoints extraction and matching: SIFT

Up to this point, we have presented how to derive the fundamental matrix or the essential matrix from the correspondences of points in two images and how to extract the relative camera pose from them. The first and crucial step is thus to compute the interest points, known as the keypoint features in each image and then match them. Different methods have been developed for interest points detection and matching, either hand-crafted or in a learning-based manner. Here, we introduce the most popular hand-crafted method SIFT [95].

A keypoint is a locally distinctive image location in two dominant and orthogonal edge directions. It is invariant to translation, rotation and illumination change. One classical keypoint detection, known as Harris corner [59], consists of finding the keypoint through the search for the intensity changes in two directions by computing the sum squared difference (SSD) of neighbor pixels inside the local patch around (x, y), denoted as W_{xy}, with

$$\text{SSD}(x,y) = \sum_{(u,v)\in W_{xy}} (I(x+u, y+v) - I(x,y))^2.$$

With the Taylor approximation $I(x+u, y+v) \approx I(x,y) + [J_x, J_y][u,v]^T$, with $J_x = \frac{\partial I(x,y)}{\partial x}$ and $J_y = \frac{\partial I(x,y)}{\partial y}$, we may write $\text{SSD}(x,y)$ as

$$\text{SSD}(x,y) \approx [u,v] \begin{bmatrix} \sum_{W_{xy}} J_x^2 & \sum_{W_{xy}} J_x J_y \\ \sum_{W_{xy}} J_y J_x & \sum_{W_{xy}} J_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = [u,v] M [u,v]^T, \tag{1.15}$$

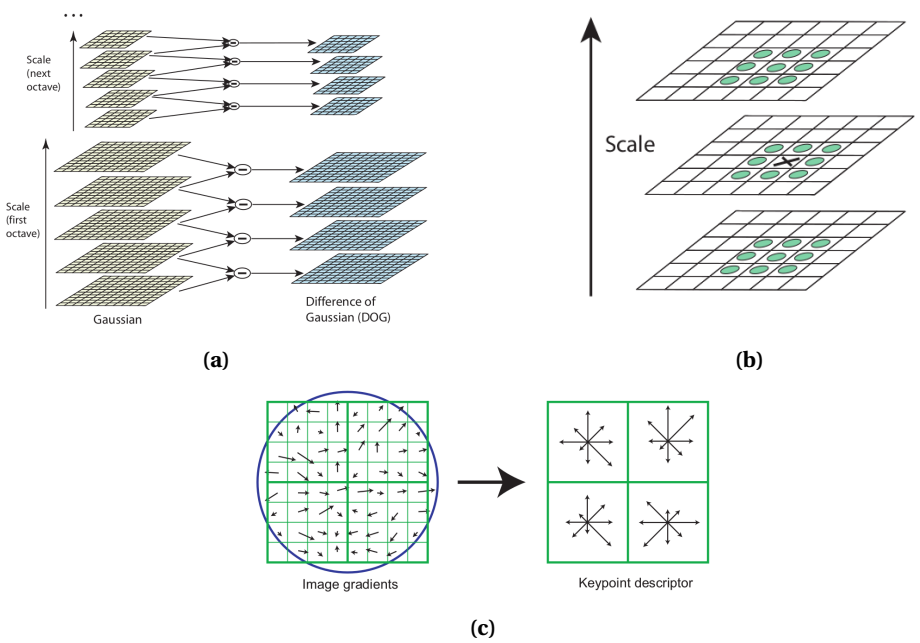


Figure 1.4: Keypoints and descriptor with SIFT [95]. (a) for the pyramid of different-of-Gaussian images, (b) for the extrema extraction in scale-space and (c) for the illustration of descriptor construction using 8×8 cells.

where M is called the structure matrix and computed from the image gradients. A pixel is considered as a keypoint or a corner if its structure matrix has two large eigenvalues , which means the dominant changes happen in two directions. A variant for this corner detection method is to detect the bloc with Differential of Gaussians, which is the basic idea of the widely used Scale-Invariant Image Features (**SIFT**).

By subtracting the blurred images with adjacent scale Gaussian smoothing, the method using SIFT builds a pyramid of the Difference-of-Gaussian images as the approximation of the scale-normalized Laplacian of Gaussian:

$$\mathrm{D}(x,y,\sigma) = (\mathrm{G}(x,y,k\sigma) - \mathrm{G}(x,y,\sigma)) * \mathrm{I}(x,y) = \mathrm{L}(x,y,k\sigma) - \mathrm{L}(x,y,\sigma), \tag{1.16}$$

where '$*$' presents the convolutional operation. The process is illustrated by Figure 1.4a The keypoint candidates can be identified as the local maxima and minima of $\mathrm{D}(x,y,\sigma)$ by comparing each sample point to its eight neighbors in the current image and nine neighbors in the scale above and below, as shown in Figure 1.4b. An accurate localization can be obtained with subpixel-subscale optimization. Considering a sample point as keypoint candidate and the Taylor expansion for $\mathrm{D}(x,y,\sigma)$ up to the second order at the sample point, the optimal offset from the sample point $\hat{\mathbf{x}}$ is derived with

$$\mathrm{D}(\mathbf{x}) = \mathrm{D} + \frac{\partial \mathrm{D}^{\mathrm{T}}}{\partial \mathbf{x}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathrm{T}}\frac{\partial^2 \mathrm{D}}{\partial \mathbf{x}^2}\mathbf{x}, \quad \text{s.t} \quad \frac{\partial \mathrm{D}}{\partial \mathbf{x}}(\hat{\mathbf{x}}) = 0,$$

where D is evaluated at the sample point as well as its derivatives and $\mathbf{x} = (x,y,\sigma)^{\mathrm{T}}$ is the offset from this point, which gives

$$\hat{\mathbf{x}} = -\frac{\partial^2 \mathrm{D}^{-1}}{\partial \mathbf{x}^2}\frac{\partial \mathrm{D}}{\partial \mathbf{x}}. \tag{1.17}$$

If any dimension of $\hat{\mathbf{x}}$ is larger than 0.5, the extremum is changed to a different sample point by adding the final offset $\hat{\mathbf{x}}$ to the location of its sample point. Furthermore, the keypoint candidates with low contrast are rejected if $\mathrm{D}(\hat{\mathbf{x}})$ is less than 0.03, assuming pixel range in $[0,1]$. In order to improve the stability, the edge responses are also eliminated by computing the Hessian matrix H

for D:

$$\mathrm{H} = \begin{bmatrix} \mathrm{D}_{xx} & \mathrm{D}_{xy} \\ \mathrm{D}_{xy} & \mathrm{D}_{yy} \end{bmatrix}. \tag{1.18}$$

and the candidates are discarded if

$$\frac{\mathrm{H})^2}{\mathrm{D}et(\mathrm{H})} > \frac{(r+1)^2}{r}, \tag{1.19}$$

where $r = 10$ in practice.

The orientation θ of a keypoint is identified as the dominant direction of the local gradient, which can be computed as the peak of the histogram for the direction of local gradients in the surrounding region of keypoint. A local keypoint extracted with SIFT is then represented by (x, y, σ, θ), including location, scale and orientation.

In order to match the keypoints in images, a distinctive and invariant descriptor is built upon the gradient orientation histogram around the keypoint, after the rotation of the coordinates and the gradient orientations of the surrounding region relative to the keypoint orientation in order to achieve orientation invariance. By choosing 4×4 grid of 4×4-pixel cells around the keypoint, the keypoint descriptor is a vector with the size of 128, by concatenating the gradient orientation histogram with 8 orientation bins ranging in $[0, 2\pi]$ for each cell, as shown in Figure 1.4c.

The matching step is then performed by searching for the nearest neighbor in terms of the Euclidean distance between descriptors in 128-D-space. To reduce the number of wrong correspondences, known as outliers, a ratio test for the distance between the nearest neighbor d_1 and the second nearest neighbor d_2 is applied by checking if $\frac{d_2}{d_1} < t$, where r is a threshold to be set by user.

Besides SIFT, there exists many other keypoints detector and descriptor such as SURF [11], ORB [133] or CNN-based LIFT [166]. We restrict our work on SIFT due to its competitive performance.

1.5 Robust estimation method

In practice, the estimated keypoints correspondences may be mismatched. The mismatched correspondences or outliers will make the fundamental matrix or the essential matrix not accurate and lead to relative pose error. We introduce two robust methods based on the RANdom SAmple Consensus (RANSAC) algorithm and the optmization framework with M-estimators to improve the tolerance for outliers.

1.5.1 RANSAC

The RANSAC (Random Sample Consensus) algorithm [48] is a robust method to estimate the parameters of a model with contaminated data including outliers, by working on the hypothesis verification framework. Given a set of putative correspondences of points, RANSAC randomly samples a subset of points with the size of the minimal number required to generate a hypothesis for the model parameters. The generated hypothesis is then verified against all data points by computing the support based on the total number of points which are consistent with the hypothesis. This sampling and checking process is repeated until the termination criteria is satisfied. The final solution is given by the hypothesis with the largest support. In order to ensure at least one of the selected minimal subsets does not include any outlier with a certain level of confidence η, the required minimum number of iterations k is determined by

$$k \geq \frac{log(1-\eta)}{log(1-\epsilon^m)}, \tag{1.20}$$

where ϵ is the correct matches ratio inside the input dataset, and m is the minimal number of points to generate the hypothesis for the model. In practice, the correct matches ratio ϵ is unknown

Table 1.1: Examples of M-estimator [176].

type	$\rho(x)$	$\psi(x)$	$w(x)$
L_2	$x^2/2$	x	1
L_1	$\lvert x\rvert$	$\mathrm{sgn}(x)$	$\frac{1}{\lvert x\rvert}$
L_1-L_2	$2(\sqrt{1+x^2/2}-1)$	$\frac{x}{\sqrt{1+x^2/2}}$	$\frac{1}{\sqrt{1+x^2/2}}$
L_p	$\frac{\lvert x\rvert^p}{p}$	$\mathrm{sgn}(x)\lvert x\rvert^{p-1}$	$\lvert x\rvert^{p-2}$
"Fair"	$c^2[\frac{\lvert x\rvert}{c}-\log(1+\frac{\lvert x\rvert}{c})]$	$\frac{x}{1+\lvert x\rvert/c}$	$\frac{1}{1+\lvert x\rvert/c}$
Huber $\begin{cases}\text{if} & \lvert x\rvert\le k\\ \text{if} & \lvert x\rvert\ge k\end{cases}$	$\begin{cases}x^2/2\\ k(\lvert x\rvert-k/2)\end{cases}$	$\begin{cases}x\\ k\mathrm{sgn(x)}\end{cases}$	$\begin{cases}1\\ k/\lvert x\rvert\end{cases}$
Cauchy	$\frac{c^2}{2}\log(1+(x/c)^2)$	$\frac{x}{1+(x/c)^2}$	$\frac{1}{1+(x/c)^2}$
Geman-McClure	$\frac{x^2/2}{1+x^2}$	$\frac{x}{(1+x^2)^2}$	$\frac{1}{(1+x^2)^2}$
Welsch	$\frac{c^2}{2}[1-\exp(-(x/c)^2)]$	$x\exp(-(x/c)^2)$	$\exp(-(x/c)^2)$
Tukey $\begin{cases}\text{if} & \lvert x\rvert\le c\\ \text{if} & \lvert x\rvert> c\end{cases}$	$\begin{cases}\frac{c^2}{6}(1-[1-(x/c)^2]^3)\\ c^2/6\end{cases}$	$\begin{cases}x[1-(x/c)^2]^2\\ 0\end{cases}$	$\begin{cases}[1-(x/c)^2]^2\\ 0\end{cases}$

and is approximated with the inlier ratio determined by the current hypothesis with the largest support.

There are also other variants of the RANSAC such as PROSAC [30], MLESAC [150], or deep learning-based DSAC [19] which shares the essential idea of inlier consensus. In our work, we adapt the standard RANSAC for the localization tasks.

1.5.2 M-estimator

M-estimator is another robust method to minimize a robust cost function for the residual errors:

$$\min \sum_i^N \rho(r_i^2), \tag{1.21}$$

where r_i^2 is the residual error data point x_i related to the model parameters, and ρ is a positive-symmetric function with a unique minimum at zero. It equals to solve the iterated re-weighted least-squares problem with:

$$\min \sum_i^N w(r_i) r_i^2, \tag{1.22}$$

where $w(\cdot)$ is the weight function defined as $\frac{1}{x}\frac{\partial \rho(x)}{\partial x}$, and $\psi(x) = \frac{\partial \rho(x)}{\partial x}$ is called the influential function. The function $\rho(x)$ is chosen to make the influential function to be bounded in order to limit the influence of any significant error related to a single observation on the offset of the final solution, and makes the estimation robust for outliers. Some commonly adopted choices for the function ρ are shown in Table 1.1.

1.6 Conclusion

In this chapter, we have introduced the projective camera geometry and the epipolar geometry between two views which are the foundation of our localization tasks studied in the following chapters. We have presented as well the estimation of the fundamental matrix and the essential matrix and the extraction of relative camera pose, followed by the detailed process of SIFT to build the

correspondences of local visual features which are crucial for the estimation of epipolar geometry. Two different robust methods are also highlighted for outliers rejections.

Chapter 2

Visual datasets and additional sensors

In this chapter, we introduce different datasets used in our work to evaluate the proposed algorithms for different tasks. Depending on the required scenario for the specific tasks, we may use both the available public datasets and the more specific datasets collected by ourselves. Most of them consist of pure vision data collected by cameras with a specific setup. In some datasets, we also collect additional measurements provided by GPS/IMU which are able to provide some priors on the relative pose. In the following chapters, the datasets introduced here will be referenced, and more details will be provided regarding the way they were used for a specific task.
Legal and ethical considerations In the last decade, the legal framework for recording data in public or private areas became significantly more restrictive across the globe, and specifically in EU. From a legal point of view, the datasets we use fall in the following four categories:

- dataset provided by other research groups, which are either directly accessible or accessible upon a request procedure that checks the academic status of the solicitor;
- dataset constructed by us, which has been manually anonymized (thus not usable for identification or re-identification tasks) and which is publicly available for the research community;
- dataset constructed by us, which may be provided to the academic community with restrictive clauses for the content use;
- dataset constructed by us for the final demonstrator of the S^2UCRE project, which was made accessible only to the involved project partners including the industrial ones, and which needs to be destructed at the end of the project.

2.1 Datasets

2.1.1 Central Railway Station

In order to evaluate the performance of relative camera pose with the additional pose priors in Chapter 3, we collected an urban scene dataset containing 32 images acquired with a smart phone with embedded pose sensors as mentioned in Section 2.2 in front of a central railway station, which presents a crowded urban environment. Some of the images were taken at ground level and some from the upper floors of a building, with a wide baseline.

The specific feature of these images is the fact that they contain additional data as an approximate localization using the synchronized IMU and GPS sensors which are available in the device. Following the same procedure as proposed in [52], we use the *GeoCam* application to record the pose prior information provided by the embedded sensors in the smartphone, including position and orientation. The GPS (longitude, latitude, attitude) and pose information (yaw, roll and pitch) are saved in the meta information associated with images. To be coherent with the camera-sensor

framework SOREPP [52], the geo-positions are converted to the coordinates in ENU coordinate frame. The dataset is provided to the academic community[1].

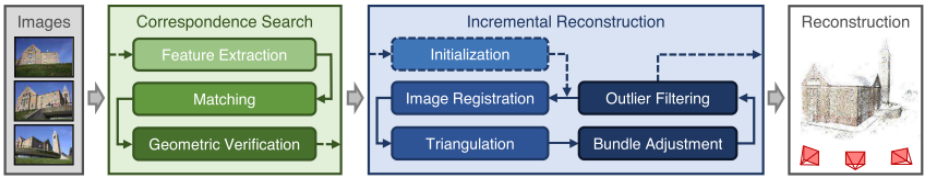


Figure 2.1: Pipeline for incremental structure from motion (source from [137].

Figure 2.2: Interface of *GeoCam*

For the ground truth calculation for relative pose estimation, we employ an structure from motion (SFM) framework on the whole dataset, as it has been performed for this task [52, 104]. Among the more commonly used frameworks [105, 134, 138, 161], we employed VisualSFM [161] for its usability and recent integration with the Linux repositories. The SFM is an integrated system including image matching, 3D triangulating and bundle-adjustment for the structure of scene and camera motion. It solves the problem of 3D point position and extrinsic camera parameters simultaneously given a collection of unordered images around the scene. The most widely used method is the incremental structure from motion. The standard pipeline can be divided into two stages, including image feature matching for each pair of images and the incremental reconstruction for all pairs, illustrated in Figure 2.2. The VisualSFM provides a feasible and efficient implementation for incremental SFM. By performing preemptive feature matching based on the scale order of features, it avoids unnecessary full matching for image pairs with insufficient matches and then accelerates the whole matching process. During the reconstruction process, the correct matches may be removed from the reconstruction due to inaccurate camera pose estimation in the early stage. To deal with this problem, the VisualSFM propose to re-triangulate all previous failed matches when the size of a model increases by 25%.

2.1.2 Building Entrance

We collected an outdoor scene dataset, denoted as *Building entrance*[2], containing a synchronized video stream acquired using a smartphone and a GoPro camera in an open outdoor environment. The GoPro camera is fixed on the upper floor of a building, while the phone is held by a moving

[1]The data used in this work may be found at: http://hebergement.u-psud.fr/emi/S2UCRE/ChenMVA19.zip
[2]Dataset contact form: http://hebergement.u-psud.fr/emi/S2UCRE/

pedestrian representing an agent equipped with a body-worn camera or a pedestrian recording during a particular event. This dataset can be considered as the simulation of the surveillance scenario for the real application. An example of image pairs is illustrated in Figure 2.3.

Both smartphone camera and GoPro camera are pre-calibrated in terms of intrinsic parameters.We compute the extrinsic parameters (rotation matrix and center) of the static camera by applying the algorithm PnP [49] on the manual landmarks. We annotated the ground truth for pedestrian detections and association labels using the annotation tool *LabelImg* [151] in order to evaluate the performance of the proposed association strategy in Chapter 7. Besides, we provide the ground truth location of the epipole corresponding to the projection of the moving smarthphone camera center in the view of the static GoPro camera, in order to evaluate the performance of epipole localization based on belief function theory in Chapter 6 by using the pedestrian detectors as the additional data source.

(a) Image from the Gopro camera **(b)** Image from the smartphone

Figure 2.3: An example of samples for the dataset *Building entrance*.

2.1.3 Saint Peters Square

The dataset *Saint Peters Square* is one of public urban dataset used in [167] for the evaluation of image matching with a geometric neural network. It collects thousands of images in front of the Saint Peters square from different views and scales. The ground truth of camera pose for each image is provided using SFM as well. We will use it for the evaluation of epipole localization based on uncertainty estimation in Chapter 4 as well as the epipole localization based on belief function theory in Chapter 6.

2.1.4 WildTrack

The *WildTrack* is also a public dataset[3] [25]. This dataset contains seven raw video recordings acquired with GoPro cameras mounted on tripods and having partly overlapping fields of view. Additionally, an annotated sequence of 400 synchronized frames is provided. The annotations represent the ground level projections for some of the pedestrians, the ones being present in the common field of view. The projections are expressed in a fixed, metric reference frame, and defined on a discrete grid of spatial resolution of 2.5 cm. The annotated detections in the form of boxes are also provided in the view of each camera. The camera intrinsic and extrinsic parameters are provided along with the visual data. The camera positions are expressed in the same reference frame as the one used for specifying the pedestrian locations. We will use it for the evaluation of across-view data association in Chapter 7 where a large number of synchronized image pairs and the annotation of pedestrians are required for training, as well as for the evaluation of epipole localization based on belief function theory in Chapter 6 in the case of temporal sequence.

[3] https://www.epfl.ch/labs/cvlab/data/data-wildtrack/

2.1.5 Final S^{2}UCRE demonstrator

The *Demo S^2 UCRE* dataset has been acquired in September 2020 in order to be used as a testbed for the final French-German demonstration of the project which took place successfully in October 2020. The sensors consist in two static GoPro cameras, along with four mobile sensors, namely two GoPro cameras and two smartphones. The static and mobile data transited for the biometry tasks to IDEMIA, which upon successful identification of perpetrators specified on a blacklist would send a geo-localization alert which would be further processed by our proposed algorithm. Unfortunately, the legal framework established between our University and the consortium, which allows industrial partners to use this dataset, restricts the data redistribution to any other entity, as well as its conservation beyond April 2021.

2.2 Pose measurement sensors

In addition to visual data, some other sensors may be placed on a camera in order to assist in localizing it in a 3D environment. A full pose measurement includes the position and the orientation of the camera sensors. All or some of the pose parameters are provided by different sensors including gyroscopes, magnetometers, accelerometers, satellite navigation systems. Different sensors can be integrated into a single hardware device called Inertial Measurement Unit (IMU), or the electronic devices such as tablet or mobile phone. In the following, we briefly introduce the working principle of these sensors.

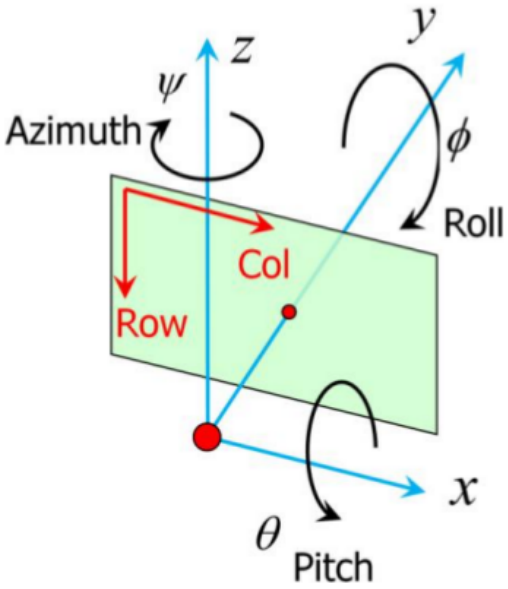


Figure 2.4: ENU coordinate system [52].

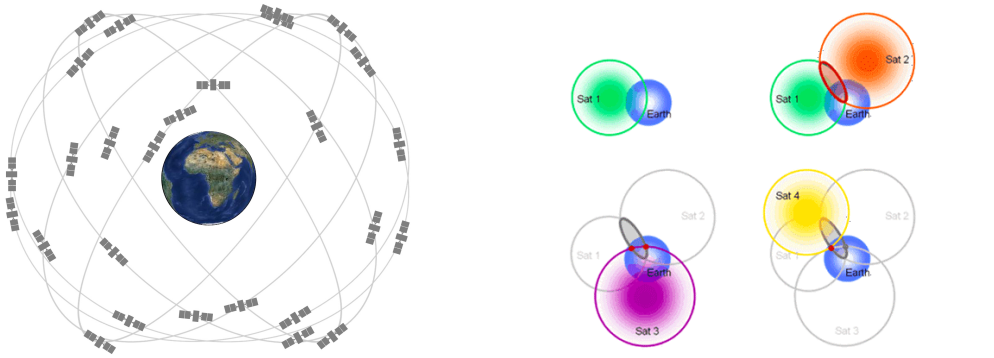
2.2.1 Coordinate system

We work with the East-North-Up (ENU) coordinate system as illustrated in Figure 2.4 and use the following convention for Euler rotation angles:

$$R(\psi,\theta,\phi) = R_Y(\phi)R_X(\theta)R_Z(\psi)$$

, with ϕ, θ and ψ correspond respectively to roll, pitch and yaw also called azimuth, and

$$R_Y(\phi) = \begin{bmatrix} cos\phi & 0 & -sin\phi \\ 0 & 1 & 0 \\ sin\phi & 0 & cos\phi \end{bmatrix}, R_X(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta & sin\theta \\ 0 & -sin\theta & cos\theta \end{bmatrix}, R_Z(\psi) = \begin{bmatrix} cos\psi & -sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.1)$$



(a) GPS satellites constellation [53]

(b) Trialteration for GPS location([136])

Figure 2.5: Global Positioning System (GPS)

2.2.2 Global Positioning System

The Global Positioning System (GPS) is a satellite-based navigation system that provides 3D position of the user including latitude, longitude and altitude. It is composed of three different segments:

1. The **space segment** consists of a constellation of at least 24 satellites distributed in 6 circular orbit 12,552 miles above Earth, illustrated in Figure 2.5a. Each satellite goes around the Earth about every 12 hours and transmit signals to Earth.

2. The **control segment** is the ground stations on Earth that monitor and track the transmission and the movement of the GPS satellites.

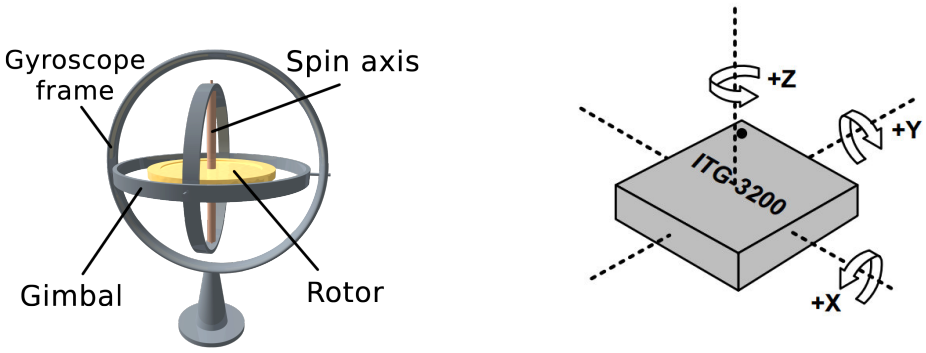
3. The **user segment** is the GPS receivers that process the signals from the GPS satellites and convert them into the 3D position and time estimates.

To locate a specific point on Earth, the GPS receiver collects signals from at least four different satellites and measure the distances (pseudo-range) to satellites using the received signals. Due to spherical symmetry, the receiver is localized at the intersection of spheres centered on a given satellite and having as radius the pseudo-range provided by this satellite. Since in addition to the receiver 3D coordiantes, the time bias between the satellites and the receive is unknwon, the researched state vector is 4D, so that at least four pseudo-range values are required to localize the receiver. Nevertheless, when more pseudo-range measurements are available, it allows for more robust and accurate localization. The trilateration process is illustrated in Figure 2.5b. The accuracy of GPS location is influenced by various factors such as the occlusions generated by buildings or mountains, and leading to Non-Line-Of-Sight or Multipath phenomena, and atmospheric conditions. In the urban environment, the GPS signal could be noisy or not available.

2.2.3 Gyroscope

The gyroscope is a device for measuring the angular velocity and orientation of an object. A classical mechanical gyroscope is composed of a spinning wheel or disc where the orientation of rotation axis is not affected by tilting according to the conservation of angular momentum [159], as shown in Figure 2.6a. An alternative to mechanical gyroscope is the microelectromechanical systems (MEMS) gyroscope, whose physical principle is those of vibrating structure gyroscopes, recalled just below. The MEMS gyroscopes are compact and small-integrated chips, as shown in Figure 2.6b. They do not require physical mobile parts so that they can be easily integrated to the electronic devices such as smartphones and cameras as well as gaming devices.

The vibrating structure gyroscope works on the detection of Coriolis forces for a rotated vibrating object. It is composed of a double-T structure crystal element containing a stationary sensing arm in the center and two symmetric drive arms on both sides. Both drive arms vibrates in a



(a) Mechanical gyroscope [159] **(b)** MEMS gyroscope [2]

Figure 2.6: Gyroscope

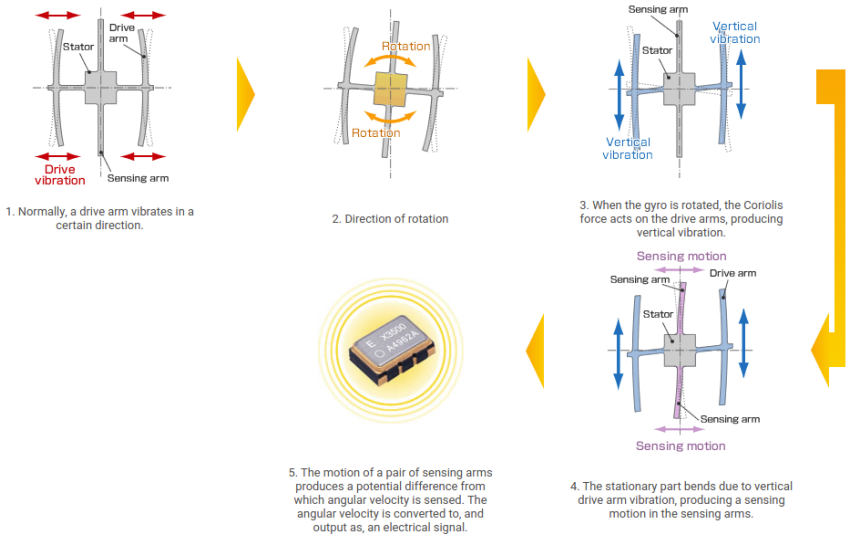


Figure 2.7: Vibrate gyroscope [1]

certain direction due to an alternating vibration electrical field. When the sensor is rotated, the Coriolis force on the drive arms leads to vertical vibration of the drive arms which causes the vibration of the stationary sensing arm. The amplitude of these vibrations is converted to electrical signal related to the angular rate for the rotation of object. The whole process is described in Figure 2.9. The rotation angle is then deduced by the integration of the angular rotation. By arranging such elements along three dimensions in space, the three rotation angles can be derived.

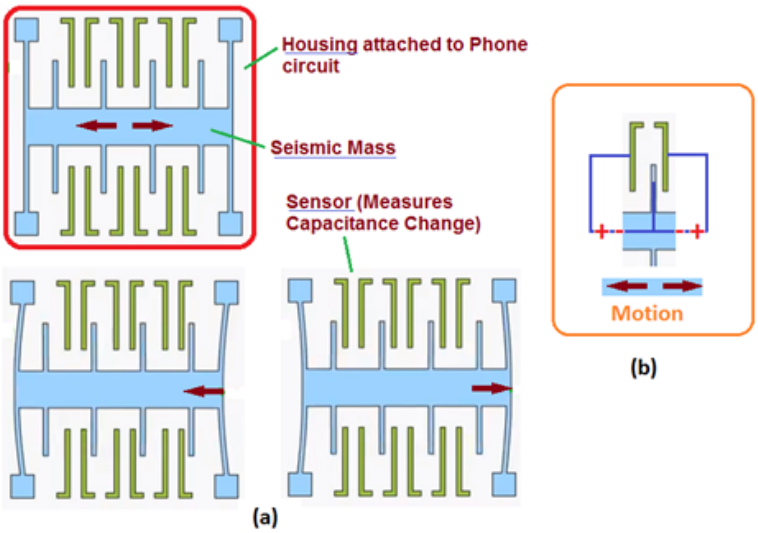


Figure 2.8: The principle of MEMS accelerometer [56]

2.2.4 Accelerometers

An accelerometer measures the acceleration (derivative of velocity) of its carrier object. The basic principle is to measure the force of the acceleration due to the gravity or external forces. There are different types of accelerometers depending on the sensing devise. The accelerometer used in the mobile phone is a MEMS-Based sensor related to the capacitance, illustrated in Figure 2.8. The acceleration is measured by the change of capacitance arising from the change of pose for the carrier object.

When there are no external forces on the object, the acceleration is opposite to gravity and points up in the world coordinate frame. The transformation between the measured acceleration in the local coordinate frame of an object and the world coordinate frame can be expressed as

$$\hat{\mathbf{a}} = \frac{\mathbf{a}}{\|\mathbf{a}\|} = \mathrm{R_Y}(\phi)\mathrm{R_X}(\theta)\mathrm{R_Z}(\psi)(0,0,1)^{\mathrm{T}} = (-cos(\theta)sin(\phi), sin(\theta), cos(\theta)cos(\phi),)^{\mathrm{T}}, \tag{2.2}$$

where $\mathbf{a}$ is the measured acceleration from 3-axis in the local coordinate frame, and ϕ, θ and ψ are respectively roll, pitch and yaw. As the component related to ψ is called out in the measured acceleration, the accelerometer data can only be used to estimate pitch and roll but not yaw. An additional sensor, namely a magnetometer, is needed to obtain the three rotation angles.

2.2.5 Magnetometers

A magnetometer or compass measures the orientation for direction to the magnetic north. The traditional compass tracks the orientation of a magnetic needle within the Earth's magnetic field without requirement of energy source. However, the magnetic sensor embedded in the mobile device is an electronic compass, whose measurement are based on the Hall-effect [58]. Such a sensor is composed of a semiconductor illustrated on Figure 2.9. When a magnetic field is present, a voltage is generated due to the movement of the electrons and holes inside the semiconductor. The strength of the magnetic field is proportional to the output of voltage which can be measured.

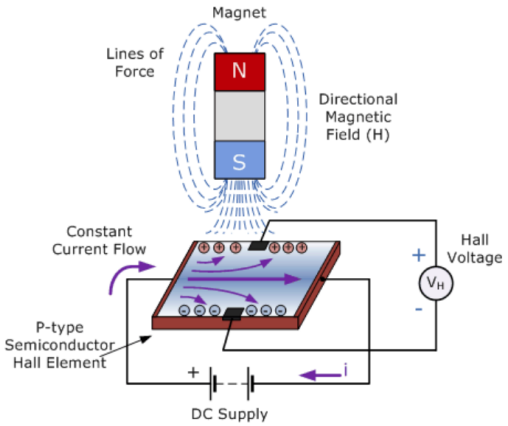


Figure 2.9: Hall effect sensor [3]

The orientation of the Earth's magnetic field can then be determined by measuring the magnetic field strengths from two perpendicular hall effect sensors inside the mobile phone.

The magnetometers can be complementary to the accelerometer by giving the information about yaw. However, yaw estimation is prone to be affected by metal and the distortions of the magnetic field and thus has a low accuracy in practical use.

Part II

Relative Camera Pose Estimation using Additional Sensors

Chapter 3

Improved relative pose estimation with visual and geometric consistency

Contents

3.1 Introduction

3.1.1 Motivation

Estimating the relative camera pose is a crucial step for both epipole localization that will be presented in Part III and the algorithm of data association which will be presented in Part IV. The relative camera pose between the mobile camera and the static camera helps to localize the mobile camera in the view of static camera. Besides, it also provides useful cues for the association of objects in the mobile view and the static view by building the 3D relationship among objects using 3D geometry. It is also a fundamental step for many computer vision tasks, including structure from motion (SFM), SLAM or object tracking, which play key roles for a wide range of applications, such as robot navigation and augmented reality.

However, urban areas raise specific difficulties for this pure image feature based methods. The performance of the estimation depends on the quality of the matching process. Due to large distances, textureless objects, repetitive structures and occlusions in the urban crowded environment, the matching process becomes difficult and inaccurate. To overcome these issues, alternative ways to allow for highly accurate relative camera pose estimation must be found. Nowadays,

sensors available in GPS receivers or IMUs are widely integrated with camera devices in smart phones. These sensors provide readily available camera pose information, albeit often very imprecise. Despite its uncertainty, the sensor prior can play a crucial role in improving pose estimation based on image feature. Previous results indicate that exploiting additional sensors injects helpful information for the pose estimation [52].

In our work, we intend to rely on the additional data sources (GPS and inertial sensors) to constrain and solve the pose estimation problem for wearable or UAV cameras in large scale urban scene. We do not assume a full cartography of the area using typically a SLAM algorithm is available or a video of the dynamic camera is available instead of just an image for registration. We tackle the problem of estimating the relative pose between two cameras in urban environments in the presence of additional information provided by low quality localization and orientation sensors like GPS, gyroscopes, magnetometers, giving a single pair of images. Our work aims to improve further the performance of pose estimation for the camera-sensor fusion framework. To this aim, we propose to use additionally the weights based on a global geometry consistency estimated by a deep network recently introduced in [104]. Then, we explore how to combine this additional criterion with the weighting strategy based on local classical features as well as the global geometric consistency.

3.1.2 Related works

Many research works have been devoted to the topic of the relative camera pose estimation. Most of them are based on purely visual estimation relying on local features. The classical pipeline for purely visual estimation relies on the local feature detectors and descriptors such as SIFT (see Section 1.4) to build the putative correspondences and then using the robust estimation method such as RANSAC (see Section 1.5.1) to filter out the false matches. Lots of efforts have been made to design more robust local feature detectors and descriptors with both hand-drafted manner (SIFT, SURF, Daisy, ORB) and the deep learning techniques [166] [29]. Many variants of RANSAC are also developed to improve the performance of robust estimation methods such as PROSAC [30], MLESAC [150]. The authors in [19] [20] attempt to integrate RANSAC process into the end-to-end CNN based neural network by making RANSAC differentiable, but does not improve the performance of RANSAC significantly. In practice, the combination of SIFT and RANSAC is still the standard process for relative camera pose estimation. Unlike the seperate process for feature matching and outliers filtering, the works in [102] [24] propose to directly predict the relative camera pose by taking RGB images from both cameras as an input of a convolutional neural network. Despite the outstanding performance of deep learning for many other computer vision tasks like image classification and object recognition, the direct regression of relative camera pose fails to achieve higher precision than the classical methods based on the local features due to the lousy ability to generalize and learn the continuous values.

An alternative way for relative camera pose estimation is to measure the camera pose with GPS/IMU sensors. Due to the limited precision for sensors, it is more practical to combine the sensor and image information to provide a more reliable estimation for the relative camera pose. There exist some fundamental approaches for combining the pose prior and the image information. The authors of [160] fuse the pose prior provided by IMU during the matching step in a process which can be considered as filtering for the relative pose estimation. They constrain the search area for correspondences around the epipolar line defined by sensor data to guide the matching. As mentioned in [52], this method is sensitive to sensor noise. For many works on visual inertial SLAM using *temporal* sequences, a popular fusion method is to constrain separately the image based estimation and pose prior as prediction step and correction step for Kalman filtering [124] [22]. Provided that video sequences cover in detail the area of interest, even pure visual SLAM may provide accurate results for registering cameras in urban scenes [125]. However, in our work we study a more constrained scenario in which a single pair of images is available, along with a low quality pose prior provided by low cost GPS and inertial sensors. Indeed, in metropolitan areas video-recording (especially using UAVs) is highly regulated, and even if videos from ground-

level dynamic cameras are available, they often have low quality and are heavily occluded.

In the last years, minimizing a loss function including both image features and pose prior by non linear optimization has been shown to be more accurate than Kalman filtering in visual-inertial SLAM [87]. The authors in [52] first proposed to extend a similar idea for the relative camera pose estimation for a single pair of views. Instead of using RANSAC, they propose an algorithm called SOREPP which relies on the fusion of putative correspondences and of noisy pose priors from sensors using the optimization framework based on M-estimator (see Section 1.5.2). However, the performance of SOREPP is sensible to the presence of a significant ratio of outliers as well as the high imprecision related to pose sensors.

Improving the quality of feature matching is crucial for either image-based estimation or camera-sensor based estimation as they are all influenced by the presence of outliers. Lots of works have also been done to improve local feature matching due to its importance for many tasks such as image retrieval as well as camera pose estimation. Most of them are based on visual cues. The classic strategy for SIFT+RANSAC based method is to apply the ratio test and the bidirectional check to reduce the ambiguities. However, a simple fixed threshold for the distance between the first nearest matches and the second nearest matches, potentially removes the false matches as well as the correct matches at the same time. A furthermore verification for matching can be performed with additional constraints such as consistency of supporting matches around the neighborhood region [14], the epipolar constraint for a correct correspondence [45, 119] (see Section 1.3), and the pedestrians-guided verification [6].

Deep learning has also been used in the image matching. In [171], the authors propose to predict the matching score by taking a pair of image patches as the input of a convolutional neural network with the supervised learning manner. Despite the advantage of supervised learning, the matching score is still limited to the local region around the keypoints and sensitive to repetitive features. The weakly supervised learning with the epipolar geometry is becoming a promising avenue to learn feature matching with neural networks [104] [129] [174]. By considering the putative matches as a 4D point cloud, they learn to predict a probability being inlier for each match, where the label for each match is assigned based on the epipolar distance. Without the visual information related to keypoints, the predicted weights depend on the global geometric consistency among all correspondences.

3.2 Image-based pose estimation

The relative camera pose estimation computes the rotation and translation from one camera to the other given a pair of images. It is based on the camera geometry and epipolar geometry between two cameras as well as the visual features extracted from the images captured by the two cameras. Traditional pose estimation relies on local image features: given a pair of images $(\mathrm{I}_a, \mathrm{I}_b)$, the standard procedure may be described as follows:

1. Keypoints extraction and matching:
   - Extracting keypoints for two images;
   - Assign a built-in descriptor such as SIFT descriptor for each keypoint;
   - Find the correspondences exhibiting smallest descriptor distance;
   - Remove the ambiguous correspondences with a fixed ratio test threshold.
2. Robust estimation:
   - Either
     + Apply RANSAC on correspondences to estimate the fundamental matrix or essential matrix with the 8-point algorithm (for instance);
     + Extract the relative camera pose;

- Or
  - + Estimate $(R, \mathbf{t})$ directly using a robust estimator like M-estimator.

When the RANSAC algorithm is applied, we need a measure to check if a data point is consistent with the hypothesis. In the case of pose estimation, our hypotheses deal with the fundamental matrix and we assess the consistency of a match with this hypothesis, the symmetric epipolar distance is widely used. It is defined as

$$d(\mathbf{x_1^i}, \mathbf{x_2^i}, F) = d(\mathbf{x_2^i}, F\mathbf{x_1^i}) + d(\mathbf{x_1^i}, F^T\mathbf{x_2^i}), \tag{3.1}$$

where $d(\mathbf{x_2^i}, F\mathbf{x_1^i})^2$ is the distance between the point $\mathbf{x_2^i}$ and the epipolar line associated to $\mathbf{x_1^i}$. Similar definitions hold for $d(\mathbf{x_1^i}, F\mathbf{x_2^i})^2$. An alternative distance measure is the Sampson distance defined as

$$d(\mathbf{x_1^i}, \mathbf{x_2^i}, F) = \frac{(\mathbf{x_2^i}^T F\mathbf{x_1^i})^2}{(F\mathbf{x_1^i})_1^2 + (F\mathbf{x_1^i})_2^2 + (F^T\mathbf{x_2^i})_1^2 + (F^T\mathbf{x_2^i})_2^2}, \tag{3.2}$$

where $(\mathbf{v})_j^2$ is the square of the j-th entry of the vector $\mathbf{v}$. If the distance related to the point pair $\mathbf{x_1^i} \leftrightarrow \mathbf{x_2^i}$ is less than a defined threshold which could be from 1 to 10 pixels depending the resolution of image, the point pair $\mathbf{x_1^i} \leftrightarrow \mathbf{x_2^i}$ is considered as an inlier with respect to F.

As an alternative, a M-estimator can be also used to directly estimate the parameters for relative camera pose, or the intermediate parameters related to the fundamental/essential matrix from which the relative camera pose is extracted.

3.3 Camera-sensor fusion framework: SOREPP

Given a single pair of images and the priors about camera pose, the algorithm SOREPP [52] provides a fusion framework of image-based estimation and the pose priors. We first present the basic idea of SOREPP and then discuss some possible improvements.

3.3.1 Relative pose representation

Considering two cameras with the Euler rotation angles $(\psi_1, \theta_1, \phi_1)$ and $(\psi_2, \theta_2, \phi_2)$ and centers in 3D coordinates $\mathbf{c_1} = (x_1, y_1, z_1)$ and $\mathbf{c_2} = (x_2, y_2, z_2)$ in the world coordinate system, the relative pose is computed as:

$$R_r = R(\psi_2, \theta_2, \phi_2)^T R(\psi_1, \theta_1, \phi_1), \mathbf{c_r} = R(\psi_1, \theta_1, \phi_1)(c_2 - c_1).$$

The relative rotation matrix R_r can be decomposed with respect to the relative Euler angles $(\psi_r, \theta_r, \phi_r)$ with Equation (2.1). The relative translation vector can be described using polar coordinates by defining the relative horizontal angle α and the relative vertical angle β a follows:

$$\alpha = \arctan2(\mathbf{c_r}(x), \mathbf{c_r}(y)), \beta = \arcsin(\frac{\mathbf{c_r}(z)}{\|\mathbf{c_r}\|}).$$

The parameters for relative pose between two cameras are then represented by a vector of angles

$$s = (\psi_r, \theta_r, \phi_r, \alpha, \beta)^T. \tag{3.3}$$

The uncertainty of pose measurements for each camera is described by a covariance matrix related to measurements, denoted by:

$$\Sigma_{i\in\{1,2\}} = \mathrm{diag}(\sigma_{\psi_i}, \sigma_{\theta_i}, \sigma_{\phi_i}, \sigma_{x_i}, \sigma_{y_i}, \sigma_{z_i}),$$

supposing that the noise is independent normally distributed. These noise values are provided by manufacturers along with other relevant specifications in the accompanying data-sheets. The uncertainty for the relative pose measurement can be then approximated with the first-order uncertainty propagation

$$\Sigma_s \approx J_1 \Sigma_1 J_1^T + J_2 \Sigma_2 J_2^T, \tag{3.4}$$

where J_i are the Jacobians of the relative pose vector with respect to each camera's pose parameters.

3.3.2 Optimization

Instead of using RANSAC, SOREPP relies on M-estimator by taking noisy pose priors given by GPS and IMU sensors as pose initialization and regularizing the solutions based on image-based estimation. By taking a set of putative correspondences $\mathscr{P}$ and the pose measurements for each camera, denoted by $(\psi_i, \theta_i, \phi_i, x_i, y_i, z_i)$ as well as the related covariance matrix Σ_i, with $i \in \{1,2\}$, SOREPP use the following M-estimator to find the optimal relative pose vector s:

$$\hat{s} = \underset{s}{\operatorname{argmin}} \left\{ c\left(\sum_{k \in \mathscr{P}} w(k)(1 - g(k,s)) \right) + \left(f(s, s_0, \Sigma_{s_0}) \right)^2 \right\}, \tag{3.5}$$

where

1. $g(k,s)$ is a Gaussian score evaluated for the correspondence k in $\mathscr{P}$ with respect to the relative pose s, defined as
$$g(k,s) = exp(\frac{-d^2(k,s)}{2\sigma_h^2}), \tag{3.6}$$
with $d(k,s)$ is the Sampson distance defined in Equation (3.2) and σ_h is a soft threshold on the Sampson distance associated with each correspondence.

2. $w(k)$ is a weight measuring the likelihood for each correspondence to be an inlier and can be empirically set equal to the ratio of the distances $d_{1\text{NN}}$ and $d_{2\text{NN}}$ from the first and second nearest neighbors of correspondence k
$$w(k) = 1 - \frac{d_{1\text{NN}}^2(k)}{d_{2\text{NN}}^2(k)}.$$

3. $f(s, s_0, \Sigma_{s_0})$ is a regularization term determined by a distance measure between the estimated s and the relative pose prior s_0:
$$f(s, s_0, \Sigma_{s_0}) = \frac{1}{5}\sqrt{(s - s_0)\Sigma_{s_0}^{-1}(s - s_0)}, \tag{3.7}$$
where s_0 is the relative pose priors computed from pose measurement for two cameras with Equation (3.3), and Σ_{s_0} is the covariance matrix of relative pose priors computed by Equation (3.4).

4. c is a parameter weighting the cost related to local visual features with respect to the cost related to the pose priors from sensors.

The pose priors s_0 is also used for the initialization of the optimization problem. By choosing different initializations around s_0, the optimal solution is determined with the minimal cost.

3.3.3 Performance

The performance of SOREPP depends on the sensor precision as well as on the quality of correspondences in the set $\mathscr{P}$ and the values of $w(\cdot)$. If the uncertainty of pose measurement is too high, the impact of $f(s)$ decreases. The presence of a significant ratio of outliers in $\mathscr{P}$ makes it difficult for the M-estimator to find good solutions, except with very precise pose priors. To guide the M-estimator, SOREPP estimates a correspondence weight in the form of a rough approximation of the prior likelihood of being an inlier [52] (see Section 3.4.1). Although it has the advantage of being simple, the proposed weighting tends to be unreliable in scenes with repetitive patterns such as in urban contexts. A feasible solution to improve the performance is to develop a more reliable weighting strategy, based on the problem of local feature matching.

3.4 Proposed combination matching strategy

We propose to explore a more reliable matching strategy exploiting local visual features from [52], as well as global geometry consistency as introduced in [104]. We integrate it into the camera-sensor based framework SOREPP to improve the accuracy of relative camera pose. We first explain how these two different matching methods guided by fundamentally different objectives may be used jointly to obtain robust correspondence weights.

3.4.1 Visual matching weight

The set of putative correspondences $\mathscr{P}$ is constructed using the classical SIFT nearest neighbor strategy [95] with a standard ratio threshold of 0.75. SOREPP defines then the weight $w(k)$ for each correspondence k based on the SIFT ratio test [95] :

$$\forall \mathbf{p} \in \mathscr{P}, w_v(\mathbf{p}) = 1 - \frac{d^2_{1\text{NN}}(\mathbf{p})}{d^2_{2\text{NN}}(\mathbf{p})}. \tag{3.8}$$

Such weights are thus independently evaluated for each correspondence, based on the assumption that the more distinctive the first nearest neighbor is, the more likely the matching corresponds to an inlier. However, this cue based only on the descriptor appearance, is not robust in presence of repetitive features and occlusion. Furthermore, while decreasing the number of outliers, the non-adaptive ratio threshold reduces as well the number of inliers, an outcome which is undesirable when inliers are scarce. Next we will discuss the weighting strategy based on global geometry consistency which avoids relying on a fixed ratio threshold.

3.4.2 Geometric matching weight based on neural network

Instead of taking images as input, the authors in [104] propose to directly deal with the putative 4D paires of corresponding points and predict the matching score for each correspondence by the weakly-supervised learning using the global epipolar geometry between the two cameras. Considering a pair of images $(\mathrm{I}_1, \mathrm{I}_2)$ and a set of normalized putative correspondences between them, denoted by

$$\mathscr{P} = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_\mathrm{N}, \}, \text{with} \quad \mathbf{p}_i = (x^i_1, y^i_1, x^i_2, y^i_2),$$

a deep network based on Multilayer Perceptron (MLP) takes the set of 4D point matches $\mathscr{P}$ as input and predict the matching score $w(\mathbf{p}_i) \in [0, 1)$ for each correspondence to classify them into inliers ($w(\mathbf{p}_i) > 0$) and outliers ($w(\mathbf{p}_i) = 0$).

As illustrated in Figure 3.1, the designed network is a 12-block ResNet [64] where each block is based on two sequential operations including a Perceptron layer with 128 neurons sharing weights (P) for each correspondence, a Context Normalization layer (CN), a Batch Normalization layer, and a ReLU. The Perceptron layer with 128 neurons sharing weights (P) allows us to deal with each correspondence independently so that the change in the point ordering will not influence the predicted matching score associated to each correspondence. To incorporate the global context information into the learned matching score, the authors in [104] develop the Context Normalization layer (CN) with the following definition:

$$\mathrm{CN}(o^l_i) = \frac{o^l_i - \mu^l}{\sigma^l},$$

where o^l_i is the output of layer l , $\mu^l = \frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} o^l_i$ and $\sigma^l = \sqrt{\frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} (o^l_i - \mu^l)^2}$ with the total number of correspondences N for each pair of images. By applying CN, the matching score for each individual correspondence depends on other correspondences as well.

The predicted matching scores are then used to derive the essential matrix $\mathrm{E} = g(\mathrm{X}, \mathbf{w})$ with the weighted 8-point algorithm reformulation by minimizing

$$\left\| \mathrm{X}^\mathrm{T} \mathrm{diag}(\mathbf{w}) \mathrm{X} \mathrm{Vec}(\mathrm{E}) \right\|,$$

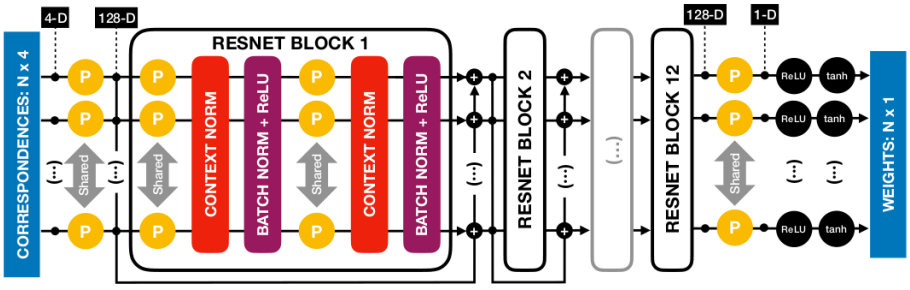


Figure 3.1: Architecture for geometric matching weight based on neural network [104]. The network consists of a sequential ResNet block and each block consists of two sequential units based on a weighting-shared Perceptron layer with 128 neurons (P) for each correspondence, followed by context normalization layer, batch normalization and ReLU.

where Vec(E) is the column vector for the coefficients of E, $\mathbf{w}$ is a diagonal matrix with the predicted matching scores and $\mathrm{X} \in \Re^{\mathrm{N}\times 9}$ is a matrix with each row computed as

$$\mathrm{X}^i = [x_1^i x_2^i, x_1^i y_2^i, x_1^i, y_1^i x_2^i, y_1^i y_2^i, y_1^i, x_2^i, y_2^i, 1],$$

with $(x_1^i, y_1^i, x_2^i, y_2^i)$ are the coordinates for the correspondence $\mathbf{p}_i$.

As it is not feasible to annotate each individual correspondence manually, the authors in [104] adapt a weakly-supervised strategy based on the ground truth essential matrix E_{gd}. The ground truth label y_i for each correspondence is inferred by comparing the distance between the corresponding epipolar line computed from E_{gd} and the corresponding point with a defined threshold.

For the training process, the authors use a hybrid loss function including the classification loss related to the predicted the matching score and the regression loss related to the estimated essential matrix from the predicted matching scores:

$$\mathscr{L}(\Phi) = \sum_{k=1}^{\mathrm{P}} (\alpha \mathscr{L}_x(\Phi, \mathscr{P}_k) + \beta \mathscr{L}_e(\Phi, \mathscr{P}_k)),$$

where Φ is the network parameters, $\mathscr{P}_k$ is the set of putative correspondences for image pair $k \in [1, \mathrm{P}]$ and P is the total number of image pairs. α and β are weighting parameters between classification loss $\mathscr{L}_x(\Phi, \mathscr{P}_k)$ based on binary cross entropy (see 7.14), and the essential matrix regression loss defined as:

$$\mathscr{L}_e(\Phi, \mathbf{x}_k) = \min \left\{ \| \mathrm{E}_k^* \pm g(\mathrm{X}_k, \mathbf{w}_k) \|^2 \right\},$$

where E_k^* is the ground truth essential matrix for each image pair k and $g(\mathrm{X}_k, \mathbf{w}_k)$ is the estimated essential matrix from the weighted correspondences for the image pair k.

The core idea is that inliers are geometrically structured, conversely to outliers. Unlike the weight based on the ratio test in Section 3.4.1 that depends only on the visual features of the considered correspondence, when considering geometric consistency among, all correspondences contribute to estimate $w(\mathbf{q}_i)$. Because this globally geometric weight is independent of appearance, it can potentially avoid the appearance ambiguity for repetitive features.

3.4.3 Exploiting both cues

It is useful to exploit jointly the local appearance cue and global geometric consistency cue because of their complementary information and their relative balance in performance and precision. In the following, different combinations of these cues are investigated and compared. For the sake of simplicity, w_v denotes the weight from classical visual features given by Equation (3.8), w_g the weight from the geometry network as presented in Section (3.4.2) and $w^{(i)}$ the combined weight to be investigated for a specific correspondence $\mathbf{p}$.

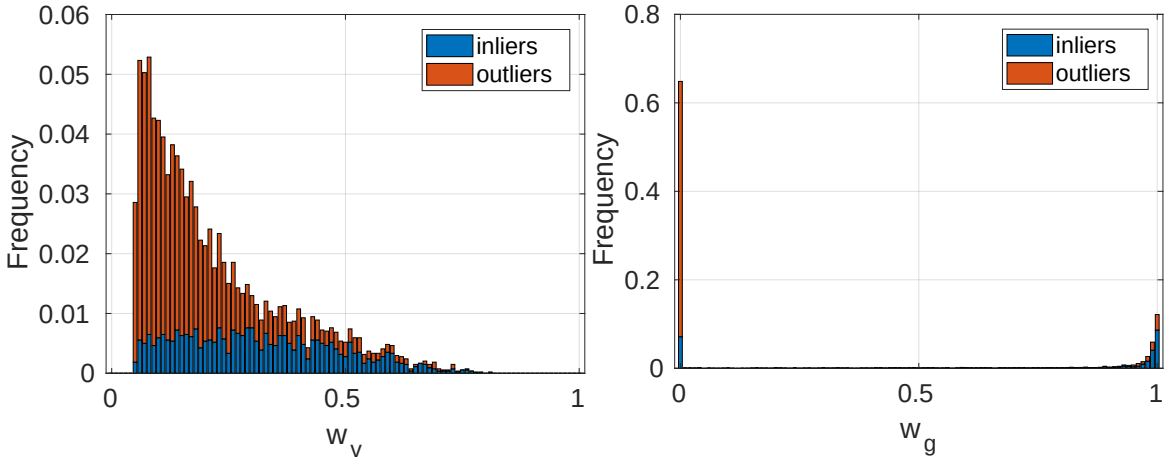


Figure 3.2: Histogram of inliers and outliers for w_v (above) and w_g (below). The neural network provides a much crisper output compared to a traditional appearance based criterion.

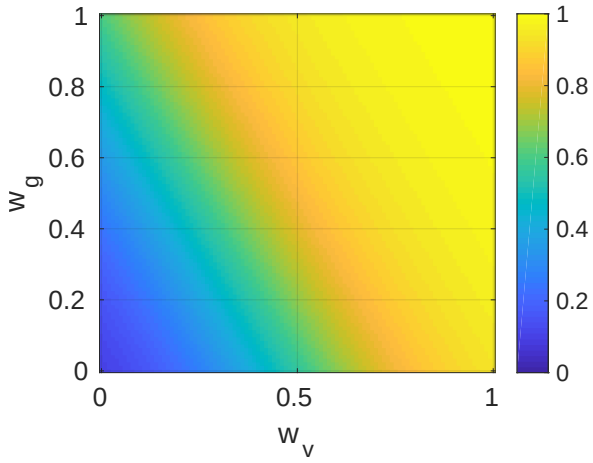


Figure 3.3: Logistic regression for w_v and w_g.

Average. The simplest fusion approach is to consider the average of w_g and w_v:

$$w^{(1)} = \frac{w_g + w_v}{2}, \tag{3.9}$$

Averaging can be seen as a weight filtering which allows us to keep considering correspondences having possible a very low weight w_v or w_g, while reinforcing correspondences with both high weights. In our case, this averaging approach is taken as the baseline.

Regression. Instead of applying a simple averaging operation, it is possible to learn automatically the fusion weight by logistic regression :

$$w^{(2)} = g(w_g, w_v), \tag{3.10}$$

where $g(\cdot)$ represents the bivariate logistic regression function.

In order to learn the parameters of this regression model, we choose image pairs in the dataset of [104] (see Section 2.1.3)and match keypoints using the classical nearest neighbor SIFT strategy. As expected, in our setting the resulting weight is close to 1 when w_v is larger than 0.8 irrespective of the value of w_g as shown in Figure 3.3. This seems to be reasonable since for few outliers the ratio test is higher than 0.8 according to Figure 3.2, upper histogram. Compared to averaging, regression tends overall to increase the combined weight, which might increase more the influence of potential inliers but also that of some outliers.

Conservative weighting. The previous strategies do not allow for a sufficient (for M-estimator efficiency) separation between inliers and outliers. In order to be more cautious since kept outliers are much more detrimental than removed inliers, we propose to adopt a conservative weighting based on a pessimistic function such as $\min(w_g, w_v)$. One significant consequence is that, in contrast to previous strategies, the min operation discards entirely the correspondences with $w_g = 0$, which exhibit a high ratio of outliers.

Since w_v seems to be more unreliable according to Figure 3.2-right, the symmetric weight from visual cues denoted by w_{vs} (computed by interchanging for the association the source and destination images in the pair) is also taken into account in order to constrain more the influence of visual part, and w_v is substituted by $min\{w_v, w_{vs}\}$. However, the geometry network exhibits already a conservative behaviour in the way it outputs the weights (see Figure 3.2-left histogram), thus the symmetric weight from the network denoted by w_{gs} is used conversely in order to allow for more prospective points being identified by geometric coherence: $\max\{w_g, w_{gs}\}$. Finally, by taking into account the particular behaviour of the two weighting algorithms, the fusion weight is computed using the following formula:

$$w^{(3)} = \min\{\max\{w_g, w_{gs}\}, \min\{w_v, w_{vs}\}\}, \tag{3.11}$$

The result in Equation (3.11) is also due to the fact that the loss function used in the neural network encourages a crisp decision regarding the nature of each input observation, a behavior which is clearly visible as well in Figure 3.2 when comparing the geometry weighting with the histogram of w_v values.

3.5 Experiments and results

To evaluate the proposed approach, we consider the dataset presented in Section 2.1.1. In the experimental part, we evaluate the performance with 992 pairs of images generated from the collected dataset. We compare the weighting strategies introduced in the previous section to SOREPP algorithm [52] and to the geometric network [104]. In this section, we first introduce the implementation details, then the evaluation metric, before discussing the results.

Given an image pair from the dataset, we compute two correspondence sets $\mathscr{P}_1$ and $\mathscr{P}_2$ using the SIFT nearest neighbor strategy with two different ratio test thresholds. A standard ratio test threshold 0.75 is chosen for $\mathscr{P}_1$. For $\mathscr{P}_2$, we choose a very high value of 0.95 which almost amounts to cancelling the ratio test, but it actually still helps eliminating a number of false correspondences. We compute the weight w_v as in Equation (3.8) for each correspondence in $\mathscr{P}_1$ and $\mathscr{P}_2$ during the matching step. Furthermore, $\mathscr{P}_2$ is set as the input of the geometric network in [104] and we get a geometric weight w_g for each correspondence in $\mathscr{P}_2$. We use $\mathscr{P}_2^*$ to denote the subset of $\mathscr{P}_2$ where w_g is greater than 0 and each correspondence in $\mathscr{P}_2$ is converted to normalized coordinates by $\hat{\mathbf{q}} = \mathrm{K}^{-1}\mathbf{q}$ where K is the intrinsic matrix of camera as introduced in Section 1.2. We applied the model trained by the authors of [104] on outdoor scenes. Then, from w_v and w_g, we compute the combined weights $w^{(1)}$, $w^{(2)}$ and $w^{(3)}$ for any correspondence in $\mathscr{P}_2$, using respectively Equation (3.9), (3.10) and (3.11). According to the various inputs and algorithms, we classify the different estimation methods as shown in Table 3.1.

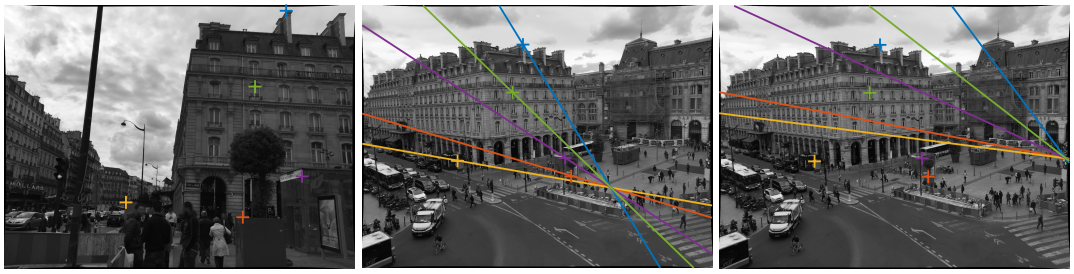
Given the estimated rotation matrix and translation vector $(\mathrm{R}_{es}, t_{es})$ and the ground truth $(\mathrm{R}_{gd}, t_{gd})$ for relative camera poses, we compute the rotation error $\delta\mathrm{R}$ and translation error δt as follows. For $\delta\mathrm{R}$, we firstly compute the relative rotation matrix $\Delta\mathrm{R}$ between R_{es} and R_{gd}, with $\Delta\mathrm{R} = \mathrm{R}_{es}^{\mathrm{T}} \cdot \mathrm{R}_{gd}$. $\Delta\mathrm{R}$ thus represents the rotation error under the form of a matrix, which can also be encoded as a rotation of angle ϕ around a vector v. Smaller is ϕ, closer R_{es} is to R_{gd}. Then, for sake of simplicity, we focus on ϕ as a measure of the rotation error $\delta\mathrm{R}$. Following [60] (see page 584), ϕ is computed as:

$$\delta\mathrm{R} \approx \phi = \arccos\left(\frac{\mathrm{Trace}(\Delta\mathrm{R}) - 1}{2}\right), \tag{3.12}$$

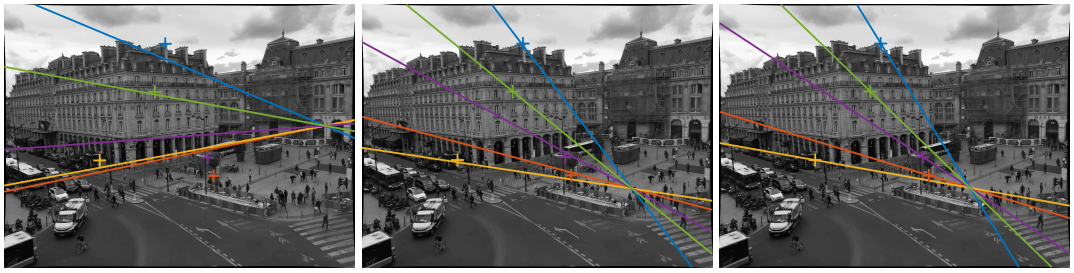
with $\mathrm{T}r(\cdot)$ the trace of a matrix.

Table 3.1: Estimation method

Name	Algorithm	Input
Sensor	-	-
SIFT-RANSAC	RANSAC	$\mathscr{P}_1$
GEO-RANSAC [104]	RANSAC	$\mathscr{P}_2^*$
SIFT-SOREPP [52]	SOREPP	$\mathscr{P}_1$, w_v, GPS-IMU estimate
GEO-SOREPP	SOREPP	$\mathscr{P}_2$, w_g, GPS-IMU estimate
Ours-LO	SOREPP	$\mathscr{P}_2$, $w^{(1)}$, GPS-IMU estimate
Ours-REG	SOREPP	$\mathscr{P}_2$, $w^{(2)}$, GPS-IMU estimate
Ours-CNSV	SOREPP	$\mathscr{P}_2$, $w^{(3)}$, GPS-IMU estimate



(a) Image at ground level **(b)** Ground truth **(c)** Sensor provided pose



(d) GEO-RANSAC [104] **(e)** SIFT-SOREPP [52] **(f)** Ours-CNSV

Figure 3.4: Illustration of pose estimation results between a ground level view (a) and an overview camera. The quality may be assessed visually based on the position of the manually defined control points with respect to their epipolar lines and based on the location of the epipole.

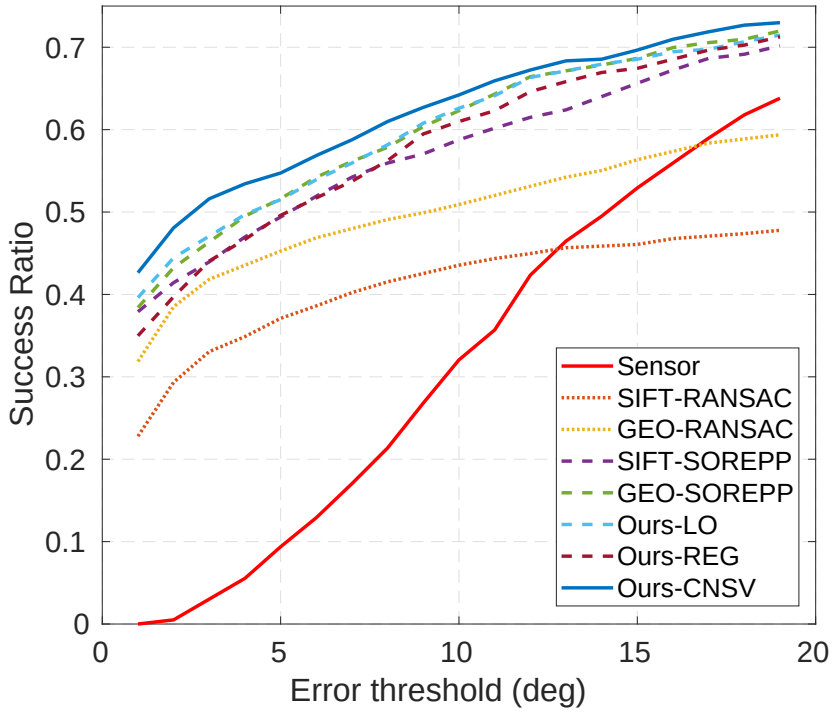


Figure 3.5: Rotation and translation success ratio of various pose estimation algorithms versus the error threshold.

The translation error δt is computed as the angle between t_{es} and t_{gd} :

$$\delta t = \arccos\left(\frac{t_{es} \cdot t_{gd}}{\|t_{es}\| \|t_{gd}\|}\right), \tag{3.13}$$

In the following, we take the pose estimation error as the maximum value between δR and δt. To evaluate the performance of each method on each pair of images of the whole dataset, we consider, like [14, 104], the cumulative density function (CDF) of pose estimation error. Then, for each evaluated method, the corresponding curve presents the percentage of image pairs whose pose is successfully estimated with respect to a given error threshold. In Figure 3.5, this metric is called "success ratio". Figure 3.5 shows a relative pose estimation for an image pair, and the corresponding epipolar lines and epipole locations for the different methods considered.

Then, Figure 3.5 provides a quantitative evaluation of the obtained results. When comparing the weighting strategies and the classic approaches, the so-called Ours-CNSV approach which uses the conservative fusion rule shows the best performance. The benefit of pose priors, even if noisy, is confirmed by the SOREPP based estimation (SIFT-SOREPP) which improves the pure vision results achieved by SIFT-RANSAC and the geometry based network (GEO-RANSAC). The average weighting and the bivariate regression weights are less effective in supporting the M-estimator. Indeed, recall that according to Equation (3.5), the computed weight keeps influencing the pose estimation. This influence is visible especially in the high precision range (error smaller than 5 degrees) which is required for the accurate localization of elements of interest in the camera fields of view. Our experiments confirm systematically that relying on the more permissive $\mathscr{P}_2$ threshold *for visual filtering* is more effective than using the stricter $\mathscr{P}_1$, because the *global* geometric consistency provided by the neural network identifies more reliably the inliers than the *local* fixed ratio test threshold used classically for SIFT matching.

3.6 Conclusion

In this chapter, we propose a strategy based on both local information provided by visual similarity, and a global geometrical coherence of the transformation between two views for relative camera pose estimation using additional sensors. The proposed combination for the two types of cues along with localization and orientation information within an M-estimator, achieves excellent results on real data acquired in an urban scenario.

The relative pose estimation is a basic element in the camera network for our localization tasks related to camera wearer and the other pedestrians. In the following chapters, we start to go into details for these two localization problem based on epipole localization and data association.

Part III

Agent Localization

Chapter 4

Epipole localization based on uncertainty estimation

Contents

4.1 Introduction

4.1.1 Motivation

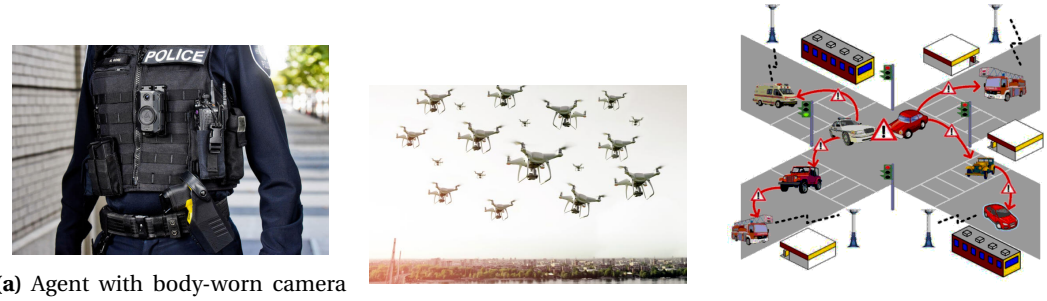
When a law enforcement agent (LEA) is equipped with the body-worn camera, the images taken from this device may be used as an additional source of information to provide an accurate location of themselves, which is useful when the GPS is not available in cluttered urban scenes.

This strategy can be exploited to pinpoint any camera wearer based on a photo taken within the field of view of a reference camera by determining the epipole in the reference view. Besides the agent localization in the surveillance task, localizing the person wearing the camera can become important when the wearer follows a specific event in the crowd and needs to localize his event accurately in an absolute reference system. More generally, the epipole localization problem is of significant interest for a wide range of tasks involving embedded cameras, such as locating LEAs or pedestrian camera wearers in the field of view of static security cameras, and mutual localization inside swarms of coordinating UAVs or in vehicular networks.

In these real applications, estimating the confidence region as a candidate search area is more beneficial than a single epipole position since a false epipole estimation may mislead completely the localization of the target. It is well known that the epipole estimation is unstable, depending on image content and interest point detection noise, but also due to the remaining outliers and to possible degenerate configurations. However, few works have been done for the reliability of its uncertainty estimation, which is vital in practice. The influence of outlier observations missed by RANSAC outlier rejection on the uncertainty has also been underlined in [147], when estimating the uncertainty of the simpler homography transform for image registration. Assuming that inliers can be identified by robust estimation such as RANSAC, and that the noise values follow a normal distribution with a small variance as typically $\sigma \in [0, 1]$ px, the standard propagation pipeline proposed in [115] underestimates the epipole location uncertainty and yields a low level of integrity for covering the correct epipole location due to the remaining false matches, as illustrated by Figure 4.2.

For these above reasons, the problem we address here is more general than the epipole estimation itself, and it regards how to improve its reliability based on the uncertainty estimation. To set up a more reliable epipole location map, we develop a method based on multimodal sampling of the RANSAC process. In accordance with the considered application of wearable camera localization, we focus on the case when the epipole is visible in the reference view. The basic idea is to empirically evaluate the uncertainty by simulating stochastic realizations of the matched point set. Our work focuses on a family of techniques which avoid relying explicitly on error models for outlier observations during uncertainty evaluation. Each of these realizations is processed following the standard pipeline to compute the epipole location and its uncertainty to provide an elementary geometric model. Global uncertainty is then derived from the fusion of all these elementary models using a tractable voting strategy.

The benefits of the proposed stochastic approach are twofold. First, we propose an epipole localization map which increases the accuracy in locating the true epipole while avoiding at the same time to underestimate the underlying uncertainty, contrarily to existing approaches. Second, our method has a low computational burden as the sampling process exploits intermediate results performed nevertheless during the robust estimation step for outlier rejection.



(a) Agent with body-worn camera in surveillance network

(b) UAVs swarm network

(c) Vehicular Ad Hoc Network

Figure 4.1: Applications of epipole localization

Figure 4.2: Illustration of an epipole uncertainty estimation. For a given pair of images, the matches selected as inliers by RANSAC (91 out of 208 initial matches) are shown with green and red lines. There is only one false positive match which is illustrated by the red line, all the other matches being true positives. In the figure on the right (close-up of the scene in the reference view), the estimated epipole uncertainty on all inliers (presented by blue ellipse) predicts a small standard deviation of the estimated epipole, and misses the true epipole (the red dot). When the single false positive is removed, the epipole ellipse (the green ellipse) predicts the uncertainty in a more reliable way.

4.1.2 Related works

There are several approaches for the estimation of epipole. The most widely used approaches for epipole estimation rely on the relation between epipole **e** and the fundamental matrix F, with F**e** = 0. By computing first the fundamental matrix, the epipole is derived from the fundamental matrix estimation using singular value decomposition (SVD). The authors in [97] attempted to use a direct method based on the invariant cross-ratio, but failed to surpass the fundamental matrix method using the eight-points algorithm. In [28], the authors proposed a robust algorithm to estimate the fundamental matrix based on a nonlinear constraint associated with the epipoles from the epipolar geometry using four corresponding points. The epipoles are estimated by solving the optimization problem for the non linear constraint. In [16], the authors proposed to compute the epipole geometry based on virtual parallax using the parametrisation of epipole and a 3D homography for the fundamental matrix.

The epipole can also be directly estimated using the linear equation system under the epipolar geometry [27], when the fundamental matrix can be simplified as a skew-matrix related to epipole in the case where the motion between cameras is a pure translation and two cameras share the same intrinsic parameters, which is not the case in general.

In [76], a conic equation related to the epipole is derived from four correspondences and a given epipole in one view based on the invariant cross-ratio of epipolar line. The epipole is estimated by the intersection of conics determined by the available correspondent points which requires the number of correspondences less than the classical eight points algorithm. The reduction of the number of correspondences is beneficial for two reasons. First, it is more feasible to mark the correspondences manually. Besides, it reduces the chance to include the wrong correspondences which is crucial for the performance. However, it still does not eliminate the influence of potential wrong correspondences for the methods with automatic features matching.

In [92], the authors proposed to estimate the epipole based on image motion measurements or optical flow for the omnidirectional camera in which the image is a sphere. The epipole is determined by the intersection of two circles, where each circle is the intersection of image sphere and a plane derived from the optical flows of a pair of antipolar points. However, this method is

only limited to the omnidirectional camera and requires the optical flow measurements.

The above studies are only related to the estimation of epipole itself but do not involve its uncertainty estimation which is crucial for the reliability. In [84, 94, 115], the uncertainty of epipole is discussed briefly. The authors in [115] develop a closed-form solution for the Jacobian of the SVD and make it possible to estimate the epipole uncertainty with the first order propagation from the fundamental matrix, whose uncertainty is well developed in many existing works [32, 148, 177]. Alternatively, in [94], the epipole is determined by two or more measurements of motion parallax which presents the displacement of image velocities for two points whose projections coincide at the same point on image, and the uncertainty of epipole is then estimated with uncertainty propagation assuming Gaussian errors on the feature measurement. However, the latter method based on motion parallax is not feasible as the motion parallax is challenging to measure or estimate accurately. Our work still focuses on the first method using the Jacobian of the SVD from the fundamental matrix and studying the uncertainty problem.

4.2 Uncertainty quantification

Uncertainty quantification is a well-studied topic involved in almost all scientific fields such as physics, chemistry and biology, computational engineering, applied mathematics, statistics, computational science as well as finance. It is a complex problem due to various known or unknown errors including measurement errors, system errors or random experimental errors. When it comes to the computer vision, it is a problem more about quantifying the uncertainty of the predictions for a mathematical model or transformation from the noisy data, such as the transformation related to epipolar geometry, which can be generalized as a problem related to data modeling (see Chapter 15 in [154]).

Data modeling can be described as estimating a model's parameters by fitting a set of observations into the aforementioned model. An optimal estimation for the parameters relies on minimizing the disagreement between the observations and the model associated with chosen parameters. A simple example may be fitting a set of points into a straight line for which, due to the measurement errors related to the input data, the estimated parameters may be not correct even with an optimal (in the sense of an error minimization criterion) fitting of the observations. An important consideration is to estimate as well the uncertainty of the inferred parameters , in order to provide at the same time a measure of the confidence in these parameters. Assuming that the model is appropriate for the task at hand, the primary source of error considered in the model fitting problem is represented by the measurement errors on the observations.

Depending on our knowledge about the nature of measurement errors, there are different techniques for the uncertainty quantification of the estimated parameters. They can be divided into two categories: propagation and sampling. Here we introduce several commonly used techniques and discuss them in terms of the applicability as well as the limits for dealing with our problem related to epipole estimation.

4.2.1 Uncertainty propagation

When the measurement errors on the observations are known, as for example, following independent Gaussian distribution, the forward and backward propagations provide an analytical solution for approximating the covariance matrix of the estimated parameters by propagating the covariance matrix of the observations.

Definition 4.2.1. ***Forward propagation.*** *Let* $\mathbf{x} = \{x_1, x_2, ..., x_i, ..., x_N\} \in \mathbb{R}^N$ *denote a vector of random variables and* $f(\mathbf{x}) \in \mathbb{R}^M$ *be a mapping from* $\mathbb{R}^N$ *to* $\mathbb{R}^M$*, which can be approximated by* $f(\mathbf{x}) \approx \mathbf{x_0} + \mathrm{J}(\mathbf{x}-\mathbf{x_0})$ *using a first-order Taylor series expansion, where* $\mathrm{J}_\mathbf{x}$ *is the Jacobian matrix of* $f(\cdot)$ *with respect*

to $\mathbf{x}$*, computed from the first order of partial derivative* $\frac{\partial f}{\partial \mathbf{x}}$ *as follows:*

$$\mathrm{J}_{\mathbf{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_i} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_i} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_i}{\partial x_1} & \frac{\partial f_i}{\partial x_2} & \cdots & \frac{\partial f_i}{\partial x_i} & \cdots & \frac{\partial f_i}{\partial x_n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_i} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}. \tag{4.1}$$

Given the covariance matrix $\Sigma_{\mathbf{x}}$ *associated with* $\mathbf{x}$*, the covariance matrix of* $f(\mathbf{x})$ *can be expressed as*

$$\Sigma_{f(\mathbf{x})} = \mathrm{J}_{\mathbf{x}} \Sigma_{\mathbf{x}} \mathrm{J}_{\mathbf{x}}^{\mathrm{T}}. \tag{4.2}$$

Definition 4.2.2. ***Backward propagation.*** *Let* $\mathbf{p} = (p_1, p_2, ..., p_i, ..., p_{\mathrm{M}}) \in \mathbb{R}^{\mathrm{M}}$ *denote a vector of parameters of a model and* $\mathbf{x} = f(\mathbf{p})$ *be a mapping from the parameter space* $\mathbb{R}^{\mathrm{M}}$ *to the data space* $\mathbb{R}^{\mathrm{N}}$*, where* $\mathbf{x} = \{x_1, x_2, ..., x_i, ..., x_n\} \in \mathbb{R}^{\mathrm{N}}$ *is a vector of random variables for the observations. The covariance associated with the parameters* $\mathbf{p}$ *can be computed as*

$$\Sigma_{\mathbf{p}} = (\mathrm{J}_{\mathbf{p}}^{\mathrm{T}} \Sigma_{\mathbf{x}}^{-1} \mathrm{J}_{\mathbf{p}}))^{-1}, \tag{4.3}$$

where $\mathrm{J}_{\mathbf{p}}$ *is the Jacobian matrix derived from the first order partial derivative as follows:*

$$\mathrm{J}_{\mathbf{p}} = \begin{bmatrix} \frac{\partial f_1}{\partial p_1} & \frac{\partial f_1}{\partial p_2} & \cdots & \frac{\partial f_1}{\partial p_i} & \cdots & \frac{\partial f_1}{\partial p_m} \\ \frac{\partial f_2}{\partial p_1} & \frac{\partial f_2}{\partial p_2} & \cdots & \frac{\partial f_2}{\partial p_i} & \cdots & \frac{\partial f_2}{\partial p_m} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_i}{\partial p_1} & \frac{\partial f_i}{\partial p_2} & \cdots & \frac{\partial f_i}{\partial p_i} & \cdots & \frac{\partial f_i}{\partial p_m} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_m}{\partial p_1} & \frac{\partial f_n}{\partial p_2} & \cdots & \frac{\partial f_m}{\partial p_i} & \cdots & \frac{\partial f_m}{\partial p_m} \end{bmatrix}. \tag{4.4}$$

The limitation for the propagation-based method is the assumption of linearity based on the first order of Taylor series expansion. The estimation of the covariance matrix may be not accurate when the linearity assumption does not hold well. In addition, the propagation of covariance matrix is limited to the Gaussian noise for which the uncertainty can be quantified by the covariance matrix. If the noise follows a more complex distribution or is unknown, the above analytical propagation pipeline will become challenging.

4.2.2 Monte-Carlo simulation

When it comes to an error propagation of a more complex observation noise model than Gaussian noise, an alternative method for uncertainty quantification is based on numerical simulation, which is called Monte-Carlo (MC) simulation. Given a set of input dataset as well as the assumed noise distribution model, the process of Monte-Carlo can be described in the following steps:

1. Draw a realization of noise for each point in the input dataset from the assumed noise distribution independently;
2. Create a synthetic dataset by adding the noise to the original dataset;
3. Estimate the parameters with the synthetic dataset;

4. Repeat the above steps to obtain multiple estimations of parameters. The statistical distribution of these parameters estimations from the synthetic dataset gives the uncertainty of quantification for the estimated parameters.

Compared to method based on the uncertainty propagation, the Monte-Carlo simulation can be adapted to different noise models at a more expensive computation cost. However, both of them ask for an explicit noise model for the observations which is not available or difficult to estimate in practice.

4.2.3 Bootstrap method

When the nature of measurement errors is unknown, bootstraping [44] is a powerful tool to quantify the uncertainty of estimated parameters. Unlike the uncertainty propagation or MC simulation method, the bootstrap method does not require a specific noise distribution model. The only assumption is the independent and identically distributed nature of the data points for the input observations. The core idea of the bootstrap method is to derive the distribution of parameters by exploring the original observation using a resampling strategy without generating or adding new data.

In terms of popularity, bootstrapping has been longly used and studied by the robust statistics community [44, 100, 145], and became increasingly popular in the fields of machine learning [35, 55, 144] and computer vision [81, 82, 180]. More recently, the technique of bootstrapped ensembles has been applied to the uncertainty estimation of some classes of deep architectures as well, with applications in deep reinforcement learning [113, 114] and computer vision [68, 89]. Despite the fact that for the specific cases of deep architectures, recent results hint that the benefit of resampling the training dataset might be reduced [110], bootstrapping remains an accessible and well studied method for characterizing the uncertainty of a model.

Assuming the observations represent a set of independent and identically distributed data points $\mathscr{D}$, the bootstrap process consists of the following steps as illustrated in Figure 4.3:

1. Create a synthetic dataset with the same size as the original dataset, by randomly sampling the data points from the original dataset with replacement;
2. Estimate the parameters with the synthetic dataset;
3. Repeat the above steps;
4. The uncertainty of parameters is then deduced from all parameters estimated from the synthetic datasets.

Although it belongs to the stochastic approach category as well, the bootstrap differs from MC approaches, since the solution diversity does not derive from any noise assumption but dataset intrinsic variability. Instead of sampling data points by following a probably wrong noise distribution, the bootstrap method samples points exclusively among the actual observation dataset. As mentioned in [126], the basic idea is that the dataset itself is considered as the best and at the same time the only available estimator for the underlying probability distribution when the noise is unknown.

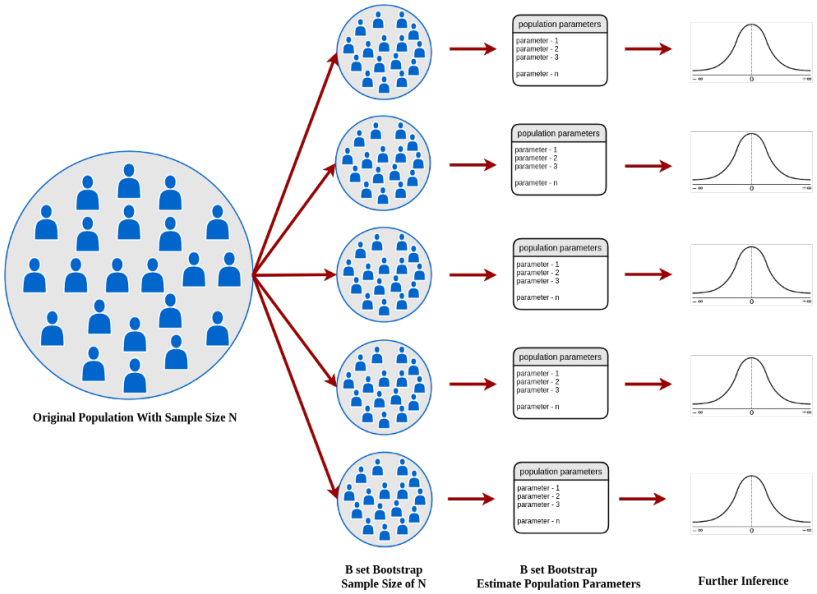


Figure 4.3: Bootstrap method([127]).

4.3 Computing the Jacobian of SVD

The singular value decomposition (SVD) is a widely used technique in computer vision for solving the least square problem such as the estimation of the fundamental matrix for more than eight point matches. The computation of the Jacobian matrix for SVD is crucial for the analytical uncertainty estimation using the forward or backward propagation formula. Its explicit expression was first clearly developed in [115]. Let A denote a matrix with the size of $M \times N$ with $M \geqslant N$. The SVD of A gives

$$A = UDV^T = \sum_{i=1}^{N} d_i U_i V_i^T, \tag{4.5}$$

where U is an $M \times N$ orthogonal matrix and U_i is its column vector, V is an $N \times N$ orthogonal matrix and V_i is its column vector, D is an $N \times N$ diagonal matrix with the singular values as diagonal element d_i.

The Jacobian of the SVD consists in computing the partial derivatives of U, V and D with respect to each element a_{ij} in A, denoted by $\frac{\partial U}{\partial a_{ij}}$, $\frac{\partial V}{\partial a_{ij}}$ and $\frac{\partial D}{\partial a_{ij}}$. By computing the derivative of Equation (4.5) with respect to a_{ij}, we can write

$$\frac{\partial A}{\partial a_{ij}} = \frac{\partial U}{\partial a_{ij}} DV^T + U \frac{\partial D}{\partial a_{ij}} V^T + UD \frac{\partial V^T}{\partial a_{ij}}, \tag{4.6}$$

with $\frac{\partial a_{ij}}{\partial a_{ij}} = 1$ and $\frac{\partial a_{kl}}{\partial a_{ij}} = 0$ for $\forall (k,l) \neq (i,j)$. By multiplying U^T and V from the left and right respectively, it gives that

$$U^T \frac{\partial A}{\partial a_{ij}} V = U^T \frac{\partial U}{\partial a_{ij}} D + \frac{\partial D}{\partial a_{ij}} + D \frac{\partial V^T}{\partial a_{ij}} V^T = \Omega_U^{ij} D + \frac{\partial D}{\partial a_{ij}} + D \Omega_V^{ij}, \tag{4.7}$$

where $\Omega_U^{ij} = U^T \frac{\partial U}{\partial a_{ij}}$ and $\Omega_V^{ij} = \frac{\partial V^T}{\partial a_{ij}} V$.

As U and V are orthogonal matrices, they satisfy

$$U^T U = UU^T = I \quad \text{and} \quad V^T V = VV^T = I.$$

The derivative gives that

$$\frac{\partial U^T}{\partial a_{ij}} U + U^T \frac{\partial U}{\partial a_{ij}} = (\Omega_U^{ij})^T + \Omega_U^{ij} = 0$$

and

$$\frac{\partial \mathrm{V}^{\mathrm{T}}}{\partial a_{ij}}\mathrm{V}+\mathrm{V}^{\mathrm{T}}\frac{\partial \mathrm{V}}{\partial a_{ij}}=\Omega_{\mathrm{V}}^{ij}+(\Omega_{\mathrm{V}}^{ij})^{\mathrm{T}}=0.$$

That indicates that Ω_{U}^{ij} and Ω_{V}^{ij} are anti-symmetric matrices with null diagonal elements. This anti-symmetric property has two effects associated with Equation (4.7):

1. Regarding the diagonal elements, it gives that

$$u_{ik}v_{jk}=0+\frac{\partial d_k}{\partial a_{ij}}+0. \tag{4.8}$$

2. Regarding the off-diagonal elements, it gives that

$$\begin{cases} u_{ik}v_{jl}=d_l(\Omega_{\mathrm{U}}^{ij})_{kl}+d_k(\Omega_{\mathrm{V}}^{ij})_{kl} \\ u_{il}v_{jk}=d_l(\Omega_{\mathrm{U}}^{ij})_{lk}+d_k(\Omega_{\mathrm{V}}^{ij})_{lk}=-d_l(\Omega_{\mathrm{U}}^{ij})_{kl}-d_k(\Omega_{\mathrm{V}}^{ij})_{kl} \end{cases} \tag{4.9}$$

By solving Equation (4.8) and the 2×2 linear systems in Equation (4.9), the derivative $\frac{\partial \mathrm{D}}{\partial a_{ij}}$ and Ω_{V}^{ij} and Ω_{U}^{ij} can be computed. Then the derivative $\frac{\partial \mathrm{U}}{\partial a_{ij}}$ and $\frac{\partial \mathrm{V}}{\partial a_{ij}}$ can be deduced from Ω_{V}^{ij} and Ω_{U}^{ij} with

$$\frac{\partial \mathrm{U}}{\partial a_{ij}}=\mathrm{U}\Omega_{\mathrm{U}}^{ij}\quad\text{and}\quad\frac{\partial \mathrm{V}}{\partial a_{ij}}=\mathrm{V}(\Omega_{\mathrm{V}}^{ij})^{\mathrm{T}}=-\mathrm{V}\Omega_{\mathrm{V}}^{ij}. \tag{4.10}$$

4.4 Fundamental matrix uncertainty

As the epipole estimation is based on the fundamental matrix, we introduce first different existing estimations for the uncertainty of the fundamental matrix quantified by the covariance matrix, before we estimate the uncertainty of epipole.

4.4.1 Uncertainty of the 8-point algorithm

One of computations for the fundamental matrix uncertainty is based on the 8-point algorithm [148]. Given a set of $n = 8$ point matches between two views, denoted by $\mathrm{X} = \{x_i, y_i, x_i', y_i'\}_{1\le i\le 8}$, the fundamental matrix F may be derived by solving the linear system $\mathrm{A}\mathbf{f}=\mathbf{c}$ with

$$\mathrm{A}=\begin{bmatrix} x_1'x_1 & x_1'y_1 & x_1' & y_1'x_1 & y_1'y_1 & y_1' & x_1 & y_1 \\ x_2'x_2 & x_2'y_2 & x_2' & y_2'x_2 & y_2'y_2 & y_2' & x_2 & y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_8'x_8 & x_8'y_8 & x_8' & y_8'x_8 & y_8'y_8 & y_8' & x_8 & y_8 \end{bmatrix}, \tag{4.11}$$

where $\mathbf{f}=(\mathrm{F}_{11},\mathrm{F}_{12},\mathrm{F}_{13},\mathrm{F}_{21},\mathrm{F}_{22},\mathrm{F}_{23},\mathrm{F}_{31},\mathrm{F}_{32})^{\mathrm{T}}$, $\mathbf{c}=-(1,1,1,1,1,1,1,1)^{\mathrm{T}}$ and F_{33} is set to be 1. Under this assumption and with the forward propagation, we get

$$\mathbf{f}=\mathrm{A}^{-1}\mathbf{c},\quad \Sigma_{\mathbf{f}}=\mathrm{J}_{\mathrm{X}}\Sigma_{\mathrm{X}}\mathrm{J}_{\mathrm{X}}^{\mathrm{T}}, \tag{4.12}$$

where J_{X} is the Jacobian matrix of $\mathbf{f}$ with respect to the set of point matches X, which can be computed with the chain rule. In order to impose the constraint of rank 2 for the fundamental matrix, the smallest singular value derived by SVD of F is forced to be 0. We use F_{rank2} to denote the rank 2 fundamental matrix. If

$$\mathrm{F}=\begin{bmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & 1 \end{bmatrix}=\mathrm{UDV}^{\mathrm{T}}, \tag{4.13}$$

then

$$\mathrm{F}_{rank2} = \mathrm{U}\mathrm{D}\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\end{bmatrix}\mathrm{V}^{\mathrm{T}}. \tag{4.14}$$

Then the covariance matrix for F_{rank2} can be derived as

$$\Sigma_{\mathrm{F}_{rank2}} = \mathrm{J}_{\mathrm{F}_{rank2}/\mathbf{f}}\begin{bmatrix}\Sigma_{\mathbf{f}} & 0\\ 0 & 0\end{bmatrix}\mathrm{J}^{\mathrm{T}}_{\mathrm{F}_{rank2}/\mathbf{f}}, \tag{4.15}$$

where $\mathrm{J}_{\mathrm{F}_{rank2}/\mathbf{f}}$ is the Jacobian matrix of F_{rank2} with respect to $\mathbf{f}$. The explicit computation for J_{X} and $\mathrm{J}_{\mathrm{F}_{rank2}/\mathbf{f}}$ can be summarized as follows:

$$\frac{\partial \mathrm{F}_{rank2}}{\partial f_i} = \frac{\partial \mathrm{U}}{\partial f_i}\mathrm{D}\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\end{bmatrix}\mathrm{V}^{\mathrm{T}} + \mathrm{U}\frac{\partial \mathrm{D}}{\partial f_i}\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\end{bmatrix}\mathrm{V}^{\mathrm{T}} + \mathrm{U}\mathrm{D}\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\end{bmatrix}\frac{\partial \mathrm{V}^{\mathrm{T}}}{\partial f_i}, \tag{4.16}$$

where $\frac{\partial \mathrm{U}}{\partial f_i}$, $\frac{\partial \mathrm{D}}{\partial f_i}$ and $\frac{\partial \mathrm{V}}{\partial f_i}$ are computed from the jacobian of SVD as presented in Section 4.3.

Uncertainty for least squares (n > 8). The input of the above method is limited to sets of 8-point matches. When $n > 8$, the fundamental matrix F can be estimated by least squares minimization and its covariance matrix can be derived from points by applying the Jacobian of the SVD as mentioned in [115]. F can be computed as in [60] by SVD from the matrix B which is the stack of $[x'_i x_i, x'_i y_i, x'_i, y'_i x_i, y'_i y_i, y'_i, x_i, y_i, 1]_{1\leq i\leq n}$. By setting $\mathbf{f} = (\mathrm{F}_{11}, \mathrm{F}_{12}, \mathrm{F}_{13}, \mathrm{F}_{21}, \mathrm{F}_{22}, \mathrm{F}_{23}, \mathrm{F}_{31}, \mathrm{F}_{32}, \mathrm{F}_{33})^{\mathrm{T}}$ with $\|\mathbf{f}\| = 1$, the covariance matrix of $\mathbf{f}$ is

$$\Sigma_f = \mathrm{J}_{\mathrm{SVD}}\Sigma_{\mathrm{X}}\mathrm{J}^{\mathrm{T}}_{\mathrm{SVD}}. \tag{4.17}$$

One can use the same strategy as with the 8 point algorithm to impose the rank-2 constraint.

From Equation (4.15), we can see that the uncertainty of the fundamental matrix is based on the uncertainty in match localization due to detector noise, as well as on their geometrical structure specified by their coordinates.

4.4.2 Uncertainty for nonlinear minimization estimation

The fundamental matrix can be written as follows:

$$\mathrm{F} = \begin{bmatrix} a & b & -ax - by \\ c & d & -cs - dy \\ -ax' - cy' & -bx' - dy' & (ax + by)x' + (cx + dy)y' \end{bmatrix}, \tag{4.18}$$

which is associated to eight variables $(a, b, c, d, x, y, x', y')$. As the fundamental matrix is defined up to the scale, the fundamental matrix can then be represented by a vector of seven parameters $\mathbf{f} = [f_1, f_2, f_3, f_4, f_5, f_6, f_7]^{\mathrm{T}}$. The fundamental matrix can be estimated iteratively by minimizing the following criterion:

$$\min_{\mathbf{f}} \sum_{i=1}^{n} \mathrm{C}_i^2(\mathbf{x}_i, \mathbf{f}), \tag{4.19}$$

where $\mathbf{x}_i$ is a pair of point matches and $\mathrm{C}_i^2(\mathbf{x_i}, \mathbf{f})$ is a cost function related to epipolar distance measure in Equation (3.2) and Equation (3.1). The covariance of the fundamental matrix F may be computed with the formula presented in [177].

Assuming $\mathbf{f}^*$ is the optimal solution for $\mathbf{f}$ by solving the non linear optimization problem in Equation (4.19), it cannot use an explicit function to quantify the mapping from the input observations and the output $\mathbf{f}^*$ for the optimization process. In this case, it involves an implicit function $\mathbf{f}^* = \varphi(\mathbf{x})$, with $\mathbf{x} = [\mathbf{x}_1, ..., \mathbf{x}_i, ..., \mathbf{x}_n]^{\mathrm{T}}$. With the forward propagation formula in Equation (4.2), the uncertainty of $\mathbf{f}^*$ can be computed as

$$\Sigma_{\mathbf{f}^*} = \mathrm{J}_\varphi(\mathbf{x}) \Sigma_{\mathbf{x}} \mathrm{J}_\varphi(\mathbf{x})^{\mathrm{T}}, \tag{4.20}$$

where $\mathrm{J}_\varphi(\mathbf{x})$ is the Jacobian of the implicit function $\mathbf{f}^* = \varphi(\mathbf{x})$ with respect to the observation $\mathbf{x}$. The implicit function theorem gives

$$\mathrm{J}_\varphi(\mathbf{x}) = -\mathrm{H}^{-1} \frac{\partial \Phi}{\partial \mathbf{x}}, \tag{4.21}$$

with

$$\Phi = \frac{\partial \sum_{i=1}^{n} \mathrm{C}_i^2}{\partial \mathbf{f}} = 2 \sum_i \mathrm{C}_i \frac{\partial \mathrm{C}_i^{\mathrm{T}}}{\partial \mathbf{f}} \tag{4.22}$$

and

$$\mathrm{H} = \frac{\partial \Phi}{\partial \mathbf{f}} \approx 2 \sum_i \frac{\partial \mathrm{C}_i^{\mathrm{T}}}{\partial \mathbf{f}} \frac{\partial \mathrm{C}_i}{\partial \mathbf{f}}, \quad \frac{\partial \Phi}{\partial \mathbf{x}} \approx 2 \sum_i \frac{\partial \mathrm{C}_i^{\mathrm{T}}}{\partial \mathbf{f}} \frac{\partial \mathrm{C}_i}{\partial \mathbf{x}}. \tag{4.23}$$

Assuming the noises on each observation are independent, by replacing Equation (4.23) and Equation (4.21) in Equation (4.20), the uncertainty of $\mathbf{f}^*$ becomes

$$\Sigma_{\mathbf{f}^*} = 4\mathrm{H}^{-1} \sum_i \frac{\partial \mathrm{C}_i^{\mathrm{T}}}{\partial \mathbf{f}} \frac{\partial \mathrm{C}_i}{\partial \mathbf{x}_i} \Sigma_{\mathbf{x}_i} \frac{\partial \mathrm{C}_i}{\partial \mathbf{x}_i}^{\mathrm{T}} \frac{\partial \mathrm{C}_i}{\partial \mathbf{f}} \mathrm{H}^{-\mathrm{T}} = 4\mathrm{H}^{-1} \sum_i \frac{\partial \mathrm{C}_i^{\mathrm{T}}}{\partial \mathbf{f}} \Sigma_{\mathrm{C}_i} \frac{\partial \mathrm{C}_i}{\partial \mathbf{f}} \mathrm{H}^{-\mathrm{T}}. \tag{4.24}$$

As the covariance Σ_{C_i} can be approximated by its sample variance, denoted by $\frac{\sum_i \mathrm{C}_i^2}{n-|\mathbf{f}|}$, where $|\mathbf{f}| = 7$ is the size of $\mathbf{f}$, the Equation (4.24) becomes

$$\Sigma_{\mathbf{f}^*} = 2\frac{\sum_i \mathrm{C}_i^2}{n-|\mathbf{f}|} \mathrm{H}^{-1} 2 \sum_i \frac{\partial \mathrm{C}_i^{\mathrm{T}}}{\partial \mathbf{f}} \frac{\partial \mathrm{C}_i}{\partial \mathbf{f}} \mathrm{H}^{-\mathrm{T}} = \frac{2\sum_i \mathrm{C}_i^2}{n-|\mathbf{f}|} \mathrm{H}^{-1} \mathrm{H} \mathrm{H}^{-\mathrm{T}} = \frac{2\sum_i \mathrm{C}_i^2}{n-|\mathbf{f}|} \mathrm{H}^{-\mathrm{T}}. \tag{4.25}$$

The covariance of the fundamental matrix F is then deduced from $\mathbf{f}$ with the forward propagation formula as follows:

$$\Sigma_{\mathrm{F}} = \frac{\partial \mathrm{F}}{\partial \mathbf{f}} \Sigma_{\mathbf{f}^*} \frac{\partial \mathrm{F}}{\partial \mathbf{f}}^{\mathrm{T}}, \tag{4.26}$$

It can be seen that the uncertainty of F for nonlinear minimization estimation depends mainly on the residual error and on the number of point matches.

4.5 Epipole uncertainty

4.5.1 Epipole estimation

The epipole of an image pair is defined as the point of intersection of the baseline with the image plane and corresponds to the projection of the camera center in the image of other camera, as illustrated in Figure 1.3, where $\mathbf{e}$ denotes the epipole on the left image I_1 and $\mathbf{e}'$ for the epipole on the right image I_2.

Given the fundamental matrix F computed from a set of n putative point matches between two views $\mathrm{M} = \{x_i, y_i, x_i', y_i'\}_{1 \le i \le n}$, the epipolar geometry gives the epipolar lines $\mathbf{l}_{\mathbf{x}'} = \mathrm{F}\mathbf{x}$ and $\mathbf{l}_{\mathbf{x}} = \mathrm{F}^{\mathrm{T}}\mathbf{x}'$ for a pair of points $(\mathbf{x}, \mathbf{x}')$. As the epipole line $\mathbf{l}_{\mathbf{x}'}$ contains the epipole $\mathbf{e}$ and $\mathbf{l}_{\mathbf{x}}$ contains the epipole $\mathbf{e}'$ for any pair of points $(\mathbf{x}, \mathbf{x}')$, the epipoles $\mathbf{e}$ and $\mathbf{e}'$ satisfy that $\mathbf{e}'^{\mathrm{T}}\mathbf{l}_{\mathbf{x}} = (\mathbf{e}'^{\mathrm{T}}\mathrm{F})\mathbf{x} = 0$ for all $\mathbf{x}$ and $\mathbf{l}_{\mathbf{x}'}^{\mathrm{T}}\mathbf{e} = \mathbf{x}'^{\mathrm{T}}(\mathrm{F}\mathbf{e}) = 0$ for all $\mathbf{x}'$. Thus, it gives that

$$\mathbf{e}'^{\mathrm{T}}\mathrm{F} = 0 \quad \text{and} \quad \mathrm{F}\mathbf{e} = 0.$$

The singular value decomposition of the fundamental matrix F gives

$$\mathrm{F} = \begin{bmatrix} \mathbf{u_1} & \mathbf{u_2} & \mathbf{u_3} \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{v_1^T} \\ \mathbf{v_2^T} \\ \mathbf{v_3^T} \end{bmatrix},$$

where $\mathbf{u_i}$ are called the left-singular vectors, $\mathbf{v_i}$ are called the right-singular vectors, and σ_i are the non null singular values. The epipole $\mathbf{e}$ is then the right null vector $\mathbf{v_3}$ and $\mathbf{e}'$ is the left null vector $\mathbf{u_3}$.

Building upon the strategies mentioned above, there are two fundamental approaches to characterizing the uncertainty of the epipole: the analytical solution based on the introduced covariance matrix estimation for the fundamental matrix, and a statistical method based on Monte Carlo simulation.

4.5.2 Analytical pipeline

Given the estimation of the fundamental matrix F and its covariance Σ_{F}, the epipole $\mathbf{e}$ and the corresponding $\mathbf{e}'$ are computed by performing the SVD of F, as the epipoles satisfy the constraints $\mathrm{F}\mathbf{e} = 0$ and $\mathbf{e}'\mathrm{F}^{\mathrm{T}} = 0$ respectively. For the epipole covariance, the authors in [115] derive the analytical solution by computing the Jacobian of the SVD:

$$\Sigma_{\mathbf{e}} = \mathrm{J}_{\mathrm{SVD}} \Sigma_{\mathrm{F}} \mathrm{J}_{\mathrm{SVD}}^{\mathrm{T}}. \tag{4.27}$$

Thus, the analytical pipeline consists of first estimating Σ_{F} with the previous presented methods on the inlier set I, by assuming that the noise of point matches in I follows an independent Gaussian distribution with a small variance. Then $\Sigma_{\mathbf{e}}$ is obtained with Equation (4.27). Specifically, the epipole and its covariance matrix can be directly extracted from the parameters F_p and its covariance Σ_{F_p} in Equation (4.26) as the parameterization of F_p is based on epipoles (we refer the reader to [177] for more details).

To visualize the epipole uncertainty related to the estimates $\mathbf{e}$ and $\Sigma_{\mathbf{e}}$, we use the k-hyper-ellipsoid based illustration [126] defined by the following equation:

$$(\mathbf{x} - \mathbf{e})^{\mathrm{T}} \Sigma_{\mathbf{e}}^{-1} (\mathbf{x} - \mathbf{e}) = \kappa^2. \tag{4.28}$$

By choosing $\kappa^2 = 5.991$, the ellipse represents the 95% confidence region according to the probability $\mathrm{P}_{\chi^2}(\kappa, 2)$ for the epipole to lie inside.

4.5.3 Simulation based estimation

An alternative solution to estimate uncertainty is a statistical method based on a standard Monte-Carlo (MC) simulation [60]. Considering the input point matches contaminated with Gaussian noise, one can draw multiple point realizations by adding to the input points some noise following the given noise distribution, and estimate the epipole with a given estimation method. The epipole uncertainty is characterized by the distribution of all the computed epipole realizations. Compared to the analytical solution, the MC approach avoids the computation of derivatives and the assumption of linearity at the cost of more expensive computations.

4.6 Proposed multi-modal sampling strategies

Similar to the fundamental matrix, the estimation of epipole is also sensible to the quality of the input point matches. With the presence of noise on the measurements of feature points, the epipole may be also perturbed. The epipole uncertainty associated to erroneous matches can be described by a covariance matrix using the first order propagation rule. However, when the correspondences

are contaminated with wrong correspondences, using a simple error model for characterizing the errors introduced by erroneous matches leads to the failure of the epipole uncertainty propagation rule. To overcome this challenge, our main aim in this part of the work is to propose a more reliable uncertainty estimation.

We argue that both the analytical solution and the MC techniques based on simulating the measurement noise will not avoid the issue highlighted in Figure 4.2, since the final error is not due to the covariance matrix approximations, but to the incorrect modelling of the observation noise. In fact, the assumption of Gaussian noise with small variance is not realistic at all for the outliers.

Therefore, we propose in the following section a solution which overcomes the above limitations by avoiding to rely on explicit assumptions about outlier noise. In order to deal with the unknown error distribution introduced by outliers, we adopt an approach for epipole uncertainty estimation inspired by the bootstrap method. A key point is that it manages to exploit part of the computation performed for outlier rejection.

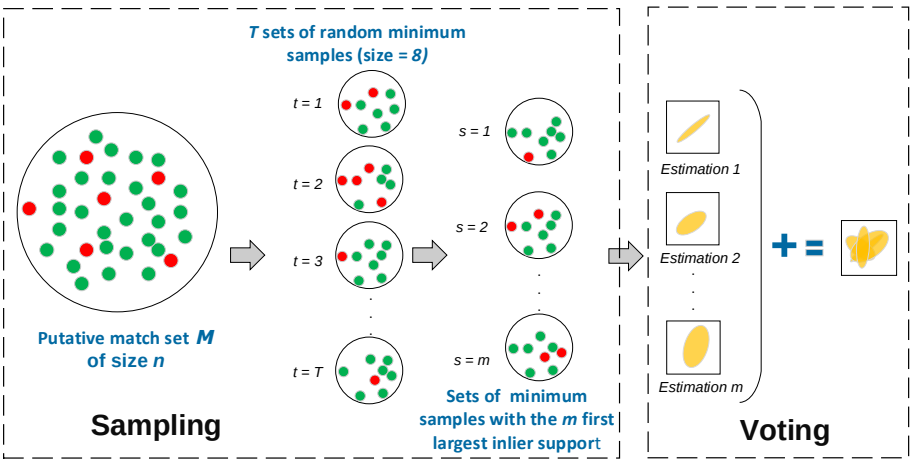


Figure 4.4: Method overview. The proposed algorithm is divided into two stages: Sampling and Voting. Given the input of putative match set M, containing true point matches (green dots) and false point matches (red dots), we randomly sample T sets of minimum number of point matches and then select the sets with the m first largest inlier support during the stage of sampling. The second stage constructs the final map by cumulative voting of the estimations from these m sets.

4.6.1 Method

The core idea is to draw different realizations by sampling point matches inside the given dataset itself. A straightforward way is to generate sample models by sampling different sets of minimal number of point matches directly from M and draw the distribution of epipoles estimated from each realization. Although this mechanism is closely related to RANSAC, the difference is that the latter process considers all the models instead of only the best one. For evaluating the uncertainty of the solution, it is not reliable to rely only on the most consensual model as we illustrated in Figure 4.2. However, considering any model is not a wise choice either, since many of them do not provide useful information because they capture few inliers, especially in difficult visual contexts.

In our work, we propose a solution which performs an accelerated model set simulation by considering models with a preference based on the number of captured inliers, illustrated by Figure 4.4. Let $\mathrm{S} = \{\mathrm{S}_1, \mathrm{S}_2, ..., \mathrm{S}_k, ..., \mathrm{S}_\mathrm{T}\}$ be the set of all sets of minimal number of point matches randomly sampled from M during RANSAC, ranked in decreasing order with respect to the size of their inlier support, denoted by $\left|\mathrm{I}_{\mathrm{S}_k}\right|$. The epipole uncertainty estimation is based on $m \ll \mathrm{T}$ models estimated from m first sets of point matches in S: $\mathrm{S}^m = \{\mathrm{S}_1, \mathrm{S}_2, ..., \mathrm{S}_k, ..., \mathrm{S}_m\} \subset \mathrm{S}$. In order to provide a continuous probability map for the epipole location over the image domain, we consider each model with the epipole $\mathbf{e_k}$ and the covariance matrix $\Sigma_{\mathbf{e_k}}$, where $\Sigma_{\mathbf{e_k}}$ is computed with the propagation formula related to 8-point algorithm (see Equation (4.15) and Equation (4.27)). For each

model, the uncertainty map of epipole localization is computed by

$$\mathrm{P}_k(\mathbf{p}) \propto exp\left\{-\frac{1}{2}(\mathbf{p}-\mathbf{e_k})^{\mathrm{T}}\Sigma_{\mathbf{e_k}}^{-1}(\mathbf{p}-\mathbf{e_k})\right\}, \tag{4.29}$$

where $\mathbf{p}$ is the pixel position in image domain $\mathscr{P}$ of dimension $[0,w]\times[0,h]$. $\mathrm{P}_k(\mathbf{p})$ denotes the probability of $\mathbf{p}$ being the true epipole coordinate given the epipole and its covariance matrix estimated on model S_k. Finally, we compute the global uncertainty map as the cumulative vote of the uncertainty map based on all m models, followed by a normalization by dividing with the maximum value over the map:

$$\mathrm{P}(\mathbf{p}) = \frac{\sum_{k=1}^{m}\mathrm{P}_k(\mathbf{p})}{\max\limits_{\mathbf{p}\in\mathscr{P}}\sum_{k=1}^{m}\mathrm{P}_k(\mathbf{p})}. \tag{4.30}$$

This uncertainty map P exhibits the coherence with the distribution of $\{\mathbf{e_k}\}_{1\leq k\leq m}$ as the maximum value is still located in the position of $\mathbf{e_k}$ for each single uncertainty map $\mathrm{P}(k)$. The cumulative voting guarantees to follow the global consistency among all considered models instead of only one single estimation. Although more complex fusion strategies exist, voting is is widely used due to its tractability for large numbers of hypotheses.

4.6.2 Discussion on parameters

The proposed approach has one key parameter, namely m the number of considered models. The value of m is expected to relate to the distribution of the support sizes $\left|\mathrm{I}_{\mathrm{S}_i}\right|$ for the most consensual models, as well as to the largest number of inliers $\left|\mathrm{I}_{\mathrm{S}_1}\right|$ captured during random sampling process. Otherwise stated, m is set to make sure that all the chosen models capture a sufficiently large number of inliers with respect to the best model. Following our validation in difficult visual contexts, we set a fixed, reasonably large value $m = 1000$ during the RANSAC sampling stage. This choice is well adapted in the case of significant, frequent ambiguities among models which capture similar numbers of inliers which are also close to $\left|\mathrm{I}_{\mathrm{S}_1}\right|$. In less ambiguous contexts, when the process is most certain about the best model and the values $\left|\mathrm{I}_{\mathrm{S}_i}\right|$ decay faster, the choice of a fixed value m may introduce more unreliable, low quality models. This problem may be easily avoided by setting a threshold τ for the number of inliers during the voting stage. If $\left|\mathrm{I}_{\mathrm{S}_i}\right| < \tau \times \left|\mathrm{I}_{\mathrm{S}_1}\right|$, then the model estimated from the set of point matches S_i will be discarded during the cumulative voting (See Equation (4.30)). Following our experimental validation in various contexts, the threshold τ is suggested to be set in the range 0.7 to 0.9.

4.6.3 Computational complexity

A significant computational advantage of our approach is that it is tailored to benefit from the intermediate evaluations performed by RANSAC, that are generally wasted by pose estimation algorithms. Therefore, we make a clear difference below between the sampling cost which is contained within the RANSAC algorithm, and the additional incurred cost with respect to outlier rejection.

As it is well known, the number of iterations T performed during RANSAC depends on the approximate ratio of true inliers among the observations, irrespective of our method. The sorting algorithm required for selecting the m most consensual models may be performed by insertion operations in a sorted vector, with an overall underlying complexity of $\mathrm{O}(\mathrm{T}\log(m))$. This additional cost is negligible with respect to the F estimation and evaluation of its support performed at each RANSAC iteration.

For the epipole location voting step (the vote domain being the image domain as well), the naive complexity will be $\mathrm{O}(m\times w\times h)$ which may be significant for large images. However, when applying Equation (4.29) in order to compute the uncertainty map for each model, one may constrain the domain of pixels to be updated p from $\mathscr{P}$ to the confidence region $\mathscr{C}$, where $\mathscr{C}$ is the k-hyper-ellipsoid defined in Equation (4.28). With this acceptable approximation, the complexity may be significantly reduced to $\mathrm{O}(m\times a\times b)$, with a and b being the main axes of the uncertainty

ellipse. As for most voting approaches, this second step which is specific for our algorithm may be trivially parallelized if real time constraints are critical.

4.7 Experiments and results

In the experiment part, we test the performance of the proposed method for epipolar uncertainty estimation with the public urban dataset used in [167]. Four variants based on the existing literature are compared with the proposed method in terms of accuracy and precision for qualitative and quantitative evaluation.

We firstly choose images from the dataset *Saint peters square* (see Section 2.1.3), with various view changes, and generate different pairs of images. We keep the ones whose epipole is visible in the reference view. In order to guarantee sufficient overlap between two views with a fair estimation of the epipolar geometry, we remove image pairs whose number of inliers found by RANSAC is less than 10 or below 20% of putative matches. The final number of image pairs is 1118. The ground truth epipole location is computed from the provided calibration information.

Let M_{SIFT} denote the set of putative point matches between two views, computed by SIFT with standard ratio test 0.75 [95] and I_{SIFT} the inlier set selected by RANSAC from M_{SIFT}. In order to compare with the method in [167], we apply their geometric neural network on M_{SIFT} to filter out false matches and generate the inlier set I_{NN}. The proposed method is applied directly on M_{SIFT}. During sampling stage, we set the number of RANSAC iterations T = 100000 based on the expected ratio of outliers. The number of considered models $m = 1000$. For voting, τ is set to be 0.9. The output is an uncertainty map for epipole location computed by Equation (4.30). Under this configuration, the standard RANSAC process takes on average 21.3s, with an additional 2.2s during our algorithm voting stage on the domain of size 768×1024 (a standard implementation run on an i7-7700HQ CPU). To compare the proposed uncertainty estimation of epipole with the standard pipeline based on the analytical solution (See Section 4.5.2), we develop four variants depending on the input point matches and uncertainty estimation method, detailed in Table 4.1. We compute a simple uncertainty map with estimated epipole $\mathbf{e}$ and its covariance matrix $\Sigma_{\mathbf{e}}$ using Equation 4.29 for these variants.

4.7.1 Evaluation Metric

Given the estimated epipole uncertainty map P and the ground truth epipole location $\mathbf{e_{gd}}$, the evaluation is based on accuracy as well as precision, defined as follows.

Accuracy. The accuracy represents the predicted likelihood for the true epipole and the precision underlines how tightly the candidate area is delineated inside the image space. The accuracy is measured with the score

$$s = \mathrm{P}(\mathbf{e_{gd}}), \tag{4.31}$$

A large score is expected for a prediction with high confidence level. We compute first the score for each pair of images and then compute the success ratio by counting the percentage of image pairs whose score is higher than a chosen threshold τ_s. By setting different values for τ_s, we obtain the curve of success ratio for each method.

Precision. The precision is measured in terms of similarity between the estimated uncertainty map P and ground truth map P_{gd}, which is defined by setting a value $\mathrm{P}_{gd}(\mathbf{e_{gd}}) = 1$, and null values elsewhere over the domain of image.

This measure is complementary to accuracy, since high scores across large areas of the image space increase the accuracy. In the following, we present two measures considered for assessing the similarity between the two maps. For both these measures, we first normalize the estimated uncertainty map across the image space in order to convert it to a proper probability distribution function.

Algorithm	Input	Estimation Method
Least **S**quares SIFT	I_{SIFT}	LS (Equation (4.17)), Equation (4.27)
Least **S**quares NN	I_{NN}	LS (Equation (4.17)), Equation (4.27)
Minimization SIFT	I_{SIFT}	Min (Equation (4.26)), Equation (4.27)
Minimization NN	I_{NN}	Min (Equation (4.26)), Equation (4.27)
Ours SIFT	M_{SIFT}	Sampling and Voting

Table 4.1: Summary of compared algorithms.

- **KL divergence based distance.** Given distributions G and Q, the KL divergence is defined as:

$$D_{KL}(G \parallel Q) = \sum_{x \in \chi} G(x) log \frac{G(x)}{Q(x)}. \tag{4.32}$$

  In order to get a symmetric measure, one may also compute the reverse $D_{KL}(Q \parallel G)$ and take the average of $D_{KL}(G \parallel Q)$ and $D_{KL}(Q \parallel G)$. The Kullback-Leibler (KL) divergence is widely used for measuring to what extent two probability distributions are related (a divergence of zero indicating actually identical distributions). However, in the case of an indicator ground truth distribution, $D_{KL}(P_{gd} \parallel P)$ is null $\forall \mathbf{p} \in \mathscr{P} \setminus \{\mathbf{e_{gd}}\}$, as $P_{gd}(\mathbf{p} \neq \mathbf{e_{gd}}) = 0$. Thus, it only depends on the value of $P(\mathbf{p} = \mathbf{e_{gd}})$. Conversely, $D_{KL}(P \parallel P_{gd})$ is not sensitive to the values of $P(\mathbf{p})$ for the same reason. Therefore, in our setting the KL divergence provides a good estimate for accuracy (as the score s), but fails to take into account the overall spatial geometry of the uncertainty map in order to provide helpful information about the localization precision (i.e. the desirable property according to which the inferred likelihood map P outputs significant values around the true epipole location).

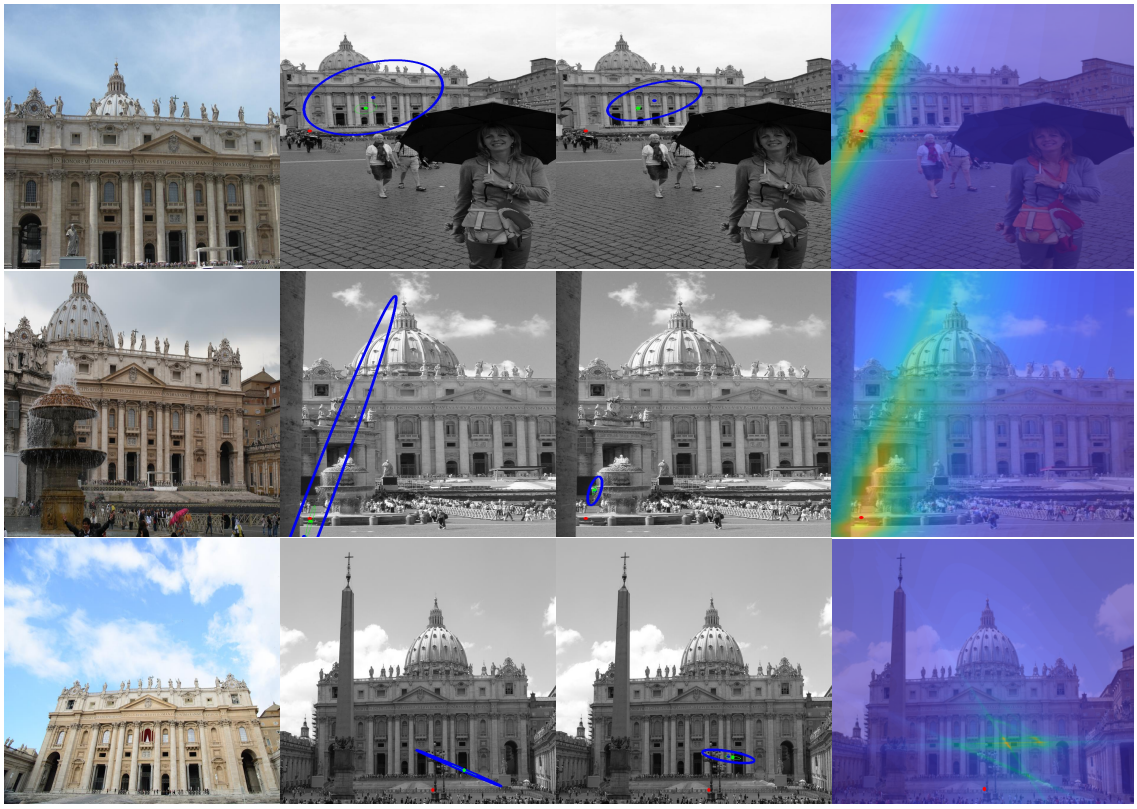
- **Optimal transport based distance.** Optimal transport (OT) has emerged as a powerful tool to evaluate the similarity of two distributions based on their spatial layout [33, 155], by computing the minimal cost in order to transform a distribution into the other one. In spite of its suitable behaviour, the use of OT has been limited by its computational cost in multidimensional spaces (more details about the underlying formalism may be found in [33]). However, in our case the optimal transport based distance also has a simplified form due to the indicator value in P_{gd} (viewed as a degenerated point distribution), and it may be conveniently computed in an exact form as follows:

$$D_{OT}(P, P_{gd}) = \frac{\sum_{\mathbf{p} \in \mathscr{P}} P(\mathbf{p}) \parallel \mathbf{p} - \mathbf{e_{gd}} \parallel_2}{\sum_{\mathbf{p} \in \mathscr{P}} P(\mathbf{p})}, \tag{4.33}$$

  when considering the Euclidean norm on $\mathbb{R}^2$, denoted by $\parallel . \parallel_2$. Hence, in our particular case in which the ground truth is an indicator function, the optimal transport may be expressed as the expectation of the localization error $\parallel \mathbf{p} - \mathbf{e_{gd}} \parallel_2$. However, in case of a different ground truth distribution, as for example in the case of an imprecise annotation, one may resort to the general OT formula.

  Unlike the KL divergence, the OT distance is able to discriminate among multiple distributions which, for the same score s, place more likelihood in the proximity of the ground truth epipole location.

Based on the characteristics of the two distance measures, we choose D_{OT} as the measure of precision. A small value presents a high level of precision. We compute D_{OT} between P and P_{gd} for each pair of image and draw the normalized histogram of D_{OT} over all image pairs.



(a) Source image **(b)** LS/Min SIFT **(c)** LS/Min NN **(d)** Ours SIFT

Figure 4.5: Qualitative performance. (a) presents the source image taken in the reference view. (b), (c) and (d) show the epipole uncertainty estimation in the reference view with different algorithms as mentioned in Table 4.1. The red dot is the ground truth epipole location. In (b) and (c), the blue ellipse presents 95% confidence region for the epipole location based on least squares estimation and the green ellipse for minimization. The proposed estimation is illustrated in (d) by a heat color map (the yellow color corresponds to the highest probability).

4.7.2 Results

Qualitative Evaluation. We present qualitative evaluation results using three examples of image pairs in Figure 4.5. The above row presents a successful scenario in which the proposed algorithm performs very well in terms of accuracy and precision with respect to existing algorithms. In this case, the most consensual model (selected by RANSAC) is invalid, therefore standard pipelines provide a false uncertainty estimation. In the middle row, we present a case where the methods we considered for comparison exhibit very variable levels of performance, despite the fundamentally similar pipeline used for uncertainty propagation. In this case, our method still exhibits a very good accuracy. Finally, the last row shows a failure case, for which all methods provide wrong evaluations due to the degeneracy of the fundamental matrix evaluation, since the inlier set is located exclusively on the same facade.

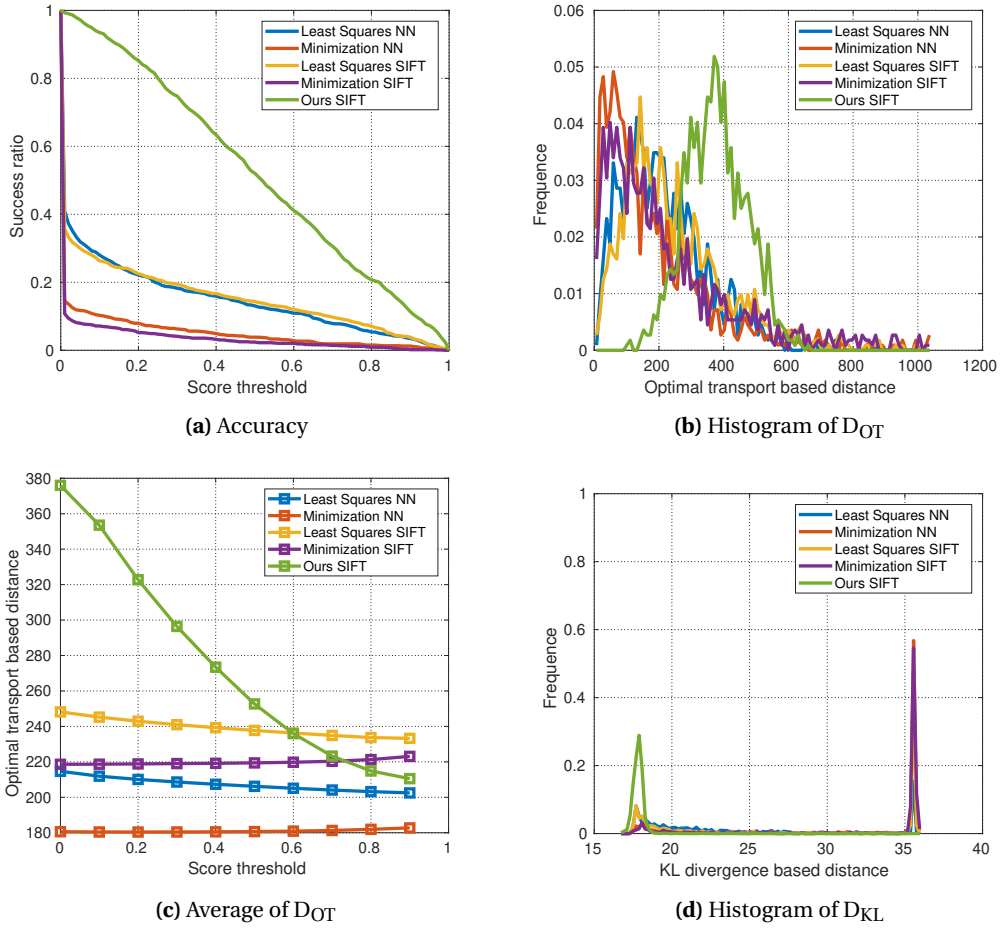
Overall, the integrity of our method is excellent. In difficult scenes, it tends to enlarge the uncertainty area which is a desirable behavior, and will fail only in the presence of degenerate geometric configurations (very limited overlap or in the presence of a single planar dominant structure).

Quantitative Evaluation. The quantitative evaluation is based at the same time on the accuracy as well as on the precision of the localization.

- **Accuracy** The accuracy evaluation is presented in Figure 4.6a. The proposed method achieves the best performance on accuracy. For other methods based on inliers of standard RANSAC, even with a very small threshold of score, their success rate is still below 40%. On the contrary, the proposed method achieves 40% success rate even with a large threshold such as 0.6. In

agreement with the behaviour predicted in Section 4.7.1, Figure 4.6d shows the KL divergence based distance for the different methods, which correlates with the score performance.

- **Precision** The evaluation on precision is shown in Figure 4.6b. Due to a more conservative estimation of the localization, the proposed method exhibits the largest distance histogram due to small value predictions over the image domain, for which the OT distance is very sensitive. However, Figure 4.6c shows the evolution of the average D_{OT} when the estimated probability maps are set to zero below a threshold (specified on the horizontal axis). Once the smallest values are gradually removed, the average distance of the proposed method approaches the distance of others methods, a fact which underlines that the peaks of $P(\mathbf{p})$ are closely located to the real locations.



(a) Accuracy

(b) Histogram of D_{OT}

(c) Average of D_{OT}

(d) Histogram of D_{KL}

Figure 4.6: Quantitative Evaluation. For all evaluated algorithms, (a) compares the percentage of image pairs whose score is larger than the correspondent threshold. (b) illustrates the normalized histogram of the optimal transport based distance for the precision evaluation. (c) is the average of the optimal transport based distance over the tested image pairs with different score threshold. (d) compares the histogram of the KL divergence based distance.

We present qualitative evaluation results using three examples of image pairs in Figure 3.4. The above row presents a successful scenario in which the proposed algorithm performs very well in terms of accuracy and precision with respect to existing algorithms. In the middle row, we present a case where the methods we considered for comparison exhibit very variable levels of performance, despite the fundamentally similar pipeline used for uncertainty propagation. In this case, our method exhibits a very good accuracy, despite a larger uncertainty area. Finally, the last row shows a failure case, for which all methods provide wrong evaluations due to the degeneracy of the fundamental matrix evaluation, since the inlier set is located exclusively on the same facade. Our

proposed method predicts robustly a pretty high probability for the true epipole when other methods on inliers of RANSAC fail to capture the true epipole or not, depending on the configuration of scene.

Both qualitative and quantitative evaluations reveal that the proposed method improves the integrity of the estimated epipole. Compared to the standard pipeline based on a single estimation of inliers found by RANSAC, which yields a low integrity of the estimated epipole due to the presence of outliers, the proposed multimodal sampling method explores different model estimations on potentially valid sets of point matches. The estimation on such sets of matches helps to decrease the chance to include outliers. The voting strategy by multiple models follows the global tendency and finally outputs a reliable prediction for epipole localization.

4.8 Conclusion

In this chapter, we introduced a strategy for estimating the epipole location which may be interpreted as an extension of a RANSAC approach by bootstrapping. Being still based on the standard RANSAC back-bone, the proposed method has a low computational cost but, compared to RANSAC, we are able to improve the estimation from an integrity point of view. In terms of the agent localization application which guides our effort, this means that we are able to provide - still based only on purely visual data - a potential search area for the mobile agent, and that the extent of this area may be controlled by the operator's aversion to risk.
In the following parts, we will focus on the joint exploitation on additional sources of information which may guide the epipole (and implicitly the LEA) localization within the Belief Function framework.

Chapter 5

2D belief function framework

Contents

In the previous chapter, we have presented how to combine multiple uncertainty maps for epipole location built from different samplings of the input point matches using the accumulated voting strategy. It can also be considered as a general problem of combining different imprecise sources for 2D localization, which can be presented by the general formalism with the Belief function theory. In this chapter, we first present the necessary notations and operations of the general Belief function theory and then introduce the 2D Belief function framework explicitly developed for the problem of the 2D localization.

5.1 Belief function theory

Belief function theory, known as Dempster-Shafer theory or Evidence theory, is a general formalism for representing imprecision and uncertainty due to lack of knowledge. It can be seen as the extension of both sets and probability measures. By considering different pieces of imprecise sources or information as evidences, it allows one to represent the degree of belief for each piece of evidence and combine all of them to make a final consistent decision with these pieces of evidence. The major work can be then divided into belief representation for the degree of belief and the combination of evidences followed by the decision making.

5.1.1 Belief representation

5.1.1.1 Mass function

Considering a variable x taking values in a finite set Ω, denoted as the *frame of discernment*, and the power set 2^{Ω} which is the set of all subsets related to Ω, a piece of evidence about X can be

represented by a ***mass function*** $m : 2^{\Omega} \rightarrow [0,1]$ such that

$$\sum_{A \subseteq \Omega} m(A) = 1, \tag{5.1}$$

where A is an element in 2^{Ω} i.e. a subset of Ω, and represents a *hypothesis* for possible solutions of the considered problem. Specifically, every A with $m(A) > 0$ is called the ***focal element*** of the mass function. The *focal set* is then the set of the focal elements.

The mass function allows for assigning a mass value as the degree of support to each hypothesis. There are several properties about the mass function.

Definition 5.1.1. *A mass function is said* ***normalized*** *if:*

$$m(\emptyset) = 0,$$

where the empty set $\emptyset$ is the hypothesis that the solution lies outside Ω. The unnormalized mass function can be converted into normalized mass function as follows:

$$m'(A) = \begin{cases} \dfrac{m(A)}{1 - m(\emptyset)} & \text{if } A \neq \emptyset, \\ 0 & \text{if } A = \emptyset. \end{cases} \tag{5.2}$$

Definition 5.1.2. *A mass function is said* ***logical*** *if* $m(A) = 1$ *where* A *is the only focal element. In this case, the evidence indicates that the solution is inside* A *for sure.*

Definition 5.1.3. *A mass function is said* ***vacuous*** *if* $m(\Omega) = 1$*, which assigns a total mass over the frame of discernment and represents a total ignorance for the considered piece of evidence. A vacuous mass function is a special case of logical mass function*

Definition 5.1.4. *A mass function is said* ***categorical*** *if the mass function is logical but not vacuous.*

Definition 5.1.5. *A mass function is said* ***simple*** *if the mass function has at most two focal elements includingg the frame of discernment* Ω *such that*

$$m(A) = 1 - w, \quad m(\Omega) = w,$$

where $w \in [0,1]$.

Definition 5.1.6. *A mass function is* ***dogmatic*** *if the frame of discernment* Ω *is not a focal element, i.e.* $m(\Omega) = 0$.

Definition 5.1.7. *A mass function is said* ***Bayesian*** *if all focal elements are singular values or singletons. In this case, the mass function is equivalent to a probability distribution.*

Definition 5.1.8. *A mass function is said* ***consonant*** *if its focal elements* $A_0, A_1 ... A_k$ *satisfied that* $A_0 \subset A_1 \subset ... \subset A_k$*, which means all focal elements are nested.*

5.1.1.2 Alternative representations

Belief function and plausibility function are three alternative representations for evidence. They are equivalent to the mass function and can be deduced from each other due to the one-to-one relationship between them.

Definition 5.1.9. *Given a mass function m for an evidence, the* ***belief function*** Bel *is defined as:*

$$Bel(A) = \sum_{B \subseteq A, B \neq \emptyset} m(B), \quad \forall A \subseteq \Omega,$$

which measures the degree that the evidence induces the focal element A *and does not support the complement* $\overline{A}$.

Definition 5.1.10. *Given a mass function m, the **Plausibility function** Pl is defined as*

$$Pl(A) = \sum_{B \cap A \neq \emptyset} m(B), \quad \forall A \subseteq \Omega,$$

which measures the degree that the evidence is consistent with A, *where* $Pl(A) = Bel(\Omega) - Bel(\overline{A})$.

Definition 5.1.11. *Given a mass function m, the **commonality function** q is defined as*

$$q(A) = \sum_{B \supseteq A} m(B), \quad \forall A \subseteq \Omega.$$

The belief function Bel and plausibility function Pl can be interpreted as the lower bound and the upper bound for the probability such that:

$$Bel(A) \leqslant P(A) \leqslant Pl(A), \quad \forall A \subseteq \Omega,$$

where the equality holds in the case of a Bayesian mass function and they are equivalent to a probability function.

The mass function can be also deduced from the belief function and plausibility function and commonality function as follows:

$$\begin{aligned} m(A) &= \sum_{B \subseteq A} (-1)^{|A \setminus B|} Bel(B), \\ m(A) &= \sum_{B \subseteq A} (-1)^{|A \setminus B|} (1 - Pl(\overline{B})), \\ m(A) &= \sum_{B \supseteq A} (-1)^{|A \setminus B|} q(B). \end{aligned}$$

5.1.1.3 BBAs comparison

The mass function specifies a ***Basic Belief Assignment(BBA)***. The comparison of two BBAs involves two types of measures. The first is to compare the commitment of information which suggests how much informative a mass function is. The other is to compute the distance of similarity between two BBAs which quantifies the consistency between two pieces of evidence.

Regarding the information commitment, there are several orderings between BBAs. These orderings can be useful when constructing belief functions by selecting the least informative BBA according to the *Least Commitment Principle*.

Definition 5.1.12. *Given two BBAs m_i and m_j, if $q_i(A) \leq q_j(A), \forall A \in 2^\Omega$, where $q(A) = \sum_{B \supseteq A} m(B)$, then m_i is **q-more committed** (more informative) than m_j, denoted by $m_i \sqsubseteq_q m_j$.*

Definition 5.1.13. *Given two BBAs m_i and m_j, if $Pl_i(A) \leq Pl_j(A)$ or equivalently $Bel_i(A) \geq Bel_j(A)$ $\forall A \in 2^\Omega$, then m_i is **Pl-more committed** (more informative) than m_j, denoted by $m_i \sqsubseteq_{Pl} m_j$.*

Regarding the distance or dissimilarity between BBAs, various distances or dissimilarity measures have been proposed [74]. The smaller the distance is, the larger is the consistency between two pieces of evidences. Our work considers Jousselme's one for its simplicity, understandable behavior and well-established mathematical properties.

Definition 5.1.14. *Given two BBAs m_i and m_j, **Jousselme's distance** between them is defined as*

$$d_J(m_i, m_j) = \sqrt{\frac{1}{2}\left(\langle m_i, m_i \rangle_J + \langle m_j, m_j \rangle_J - 2\langle m_i, m_j \rangle_J\right)}, \tag{5.3}$$

where $\langle m_i, m_j \rangle_J$ is computed as

$$\langle m_i, m_j \rangle_J = \sum_{A \in 2^\Omega} \sum_{B \in 2^\Omega} \frac{|A \cap B|}{|A \cup B|} m_i(A)\, m_j(B).$$

5.1.2 Belief function combination

Given a number of mass functions representing different independent pieces of information, denoted by $m_1, m_2, ..., m_i, ..., m_k$, several combination rules allow us to successively fuse two mass functions in order to provide a single mass function m gathering all available information. Here, we introduce the most popular rules including Dempster's rule [140], the conjunctive rule [141] and the disjunctive rule [142].

Definition 5.1.15. *Given two independent mass functions m_i and m_j obtained from two independent sources, the **unnormalized conjunctive rule** of combination is defined as*

$$m_i \textcircled{\cap} m_j(\mathrm{A}) = \sum_{\mathrm{B}\cap\mathrm{C}=\mathrm{A}} m_i(\mathrm{B})\, m_j(\mathrm{C}), \forall \mathrm{A} \in 2^{\Omega}.$$

Definition 5.1.16. *Given two independent mass functions m_i and m_j obtained from two independent sources, the **Dempster's rule** of combination is defined as*

$$m_i \oplus m_j(\mathrm{A}) = \begin{cases} \dfrac{1}{1-k} \displaystyle\sum_{\mathrm{B}\cap\mathrm{C}=\mathrm{A}} m_i(\mathrm{B})\, m_j(\mathrm{C}) & \forall \mathrm{A} \in 2^{\Omega} \setminus \{\emptyset\}, \\ 0 & \textit{if } \mathrm{A} = \emptyset, \end{cases}$$

*where κ is the **degree of conflict** between two mass functions and defined as*

$$\kappa = \sum_{\mathrm{B}\cap\mathrm{C}=\emptyset} m_i(\mathrm{B})\, m_j(\mathrm{C}).$$

The unnormalized conjunctive rule and the Dempster's rule are commutative and associative. The difference between them consists in dealing with the case for the empty set to which the mass value corresponds to the conflict between two mass functions. The unnormalized conjunctive rule allows for preserving the mass value for the empty set representing the reject hypothesis under open world assumption, while the normalized version, i.e, the Dempster's rule assumes a closed world and normalizes the mass function by assigning the mass for empty set over all other focal elements.

Definition 5.1.17. *Given two independent mass functions m_i and m_j obtained from two independent sources, the **disjunctive rule** of combination is defined as*

$$m_i \textcircled{\cup} m_j(\mathrm{A}) = \sum_{\mathrm{B}\cup\mathrm{C}=\mathrm{A}} m_i(\mathrm{B})\, m_j(\mathrm{C}), \forall \mathrm{A} \in 2^{\Omega}.$$

Like the conjunctive rule, the disjunctive rule is commutative and associative. The difference between these two rules is that the disjunctive rule relies on the union operation ($\mathrm{B} \cup \mathrm{C}$) instead of the intersection operation as for the conjunctive rule. Thus, the disjunctive rule of combination is more suitable when only one of the pieces of evidence is reliable.

5.1.3 BBA simplification

As combinations are performed, the belief becomes more fragmented across more focal elements (FE). Then, mainly for numerical reasons, the BBA has to be approximated to keep a controlled number of FEs. This process is often called BBA simplification, and in the perspective of further combination, the approaches providing a generalization of the initial BBA are favored, in particular those aggregating some FEs while following the least commitment principle.

Iterative aggregation technique [37] is one of the approximation methods. Based on a specific criterion, it selects iteratively the elements to be aggregated until the number of focal elements for a BBA reduces to the desired number. As in [8], we use the Jousselme's distance between two focal elements as the criterion in our work.

Definition 5.1.18. *Given a mass function m and two focal elements, the Jousselme's distance between two focal elements* A *and* B

$$d_{\mathrm{J}}^2(\mathrm{A},\mathrm{B}\mid m)=\left(1-\frac{|\mathrm{A}|}{|\mathrm{A}\cup\mathrm{B}|}\right)m^2(\mathrm{A})+\left(1-\frac{|\mathrm{B}|}{|\mathrm{A}\cup\mathrm{B}|}\right)m^2(\mathrm{B})\,. \tag{5.4}$$

The BBA simplification can be then described in Algorithm 1. If the above BBA simplification is performed along with BBA combination, the associativity will be not such a simplification process breaks the associativity (if it existed) of the combination.

Algorithm 1 BBA simplification

Input:
A mass function m with n focal elements $\{f_i\}_{1\leq i\leq n}$ for a piece of evidence;
The desired number of focal elements N_f;
Output: The mass function m with the number of focal elements reduced to N_f;

N_{cur} equals to the number of focal elements related to m;
while $\mathrm{N}_{cur}>\mathrm{N}_f$ **do**
  $d_{min}=1$
  for $i=0$ to N_{cur} **do**
    for $j=i+1$ to N_{cur} **do**
      if $d_{\mathrm{J}}^2(f_i,f_j|m)<d_{min}$ **then**
        $s_1=i$;
        $s_2=j$;
        $d_{min}=d_{\mathrm{J}}^2(f_i,f_j|m)$;
      end if
    end for
  end for
  Add focal element $f_{s_i}\cup f_{s_j}$ with $m(f_{s_i}\cup f_{s_j})+=m(f_{s_i})+m(f_{s_j})$;
  Remove the focal elements f_{s_i} and f_{s_j} from m;
  $\mathrm{N}_{cur}=\mathrm{N}_{cur}-1$ if $f_{s_i}\cup f_{s_j}$ does not exist, otherwise $\mathrm{N}_{cur}=\mathrm{N}_{cur}-2$;
end while
return m;

5.1.4 Decision making

After performing the combinations on the BBAs from all available sources, a decision can be made based on the resulting BBA. It is generally done by translating belief function model to probability model following the maximum operation over all singleton elements. We introduce here two most popular transformations based on piginistic probability and plausibility.

The piginistic transformation is first introduced in [143]. It is the most widely used method for probability transformation. The basic idea is to accumulate the positive degree of supports from all focal elements in which the singleton element is involved with the split of mass for each focal element over its cardinality, with the following definition:

Definition 5.1.19. *Given a mass function m, the* ***piginistic probability*** *for a singleton element x is defined as follows:*

$$\mathrm{Bet\mathrm{P}}(x)=\frac{1}{1-m(\emptyset)}\sum_{x\in\mathrm{B},\mathrm{B}\subseteq\Omega}\frac{m(\mathrm{B})}{|\mathrm{B}|},\forall x\in\Omega \tag{5.5}$$

with its unnormalized version gives

$$\mathrm{Bet\mathrm{P}}_{un}(x)=\sum_{x\in\mathrm{B},\mathrm{B}\subseteq\Omega}\frac{m(\mathrm{B})}{|\mathrm{B}|}. \tag{5.6}$$

An alternative transformation based on plausibility is proposed in [31], which may be more consistent with the Dempster's rule with the following definition:

Definition 5.1.20. *Given a mass function m, the **plausibility probability**, called also the **contour function**, for a singleton element x is defined as follows:*

$$\mathrm{P}l\_\mathrm{P}(x) = \frac{\mathrm{P}l(x)}{\sum_{y\in\Omega}\mathrm{P}l(y)}, \forall x \in \Omega \tag{5.7}$$

along with its unnormalized version $\mathrm{P}l(x)$.

5.2 2D Belief function framework: 2CoBel

Although the belief function theory provides a general framework for modeling imprecision sources, it suffers from the expensive computation burden due to massive sets operations such as intersection with the increase of the size of the discernment frame. Such issue is more challenging to handle the information fusion using belief function framework for various tasks relying on the discernment frame in 2D or higher dimension spaces, such as object detection, tracking and localization. The belief function framework for 2D discernment frame mainly involves in manipulating the 2D focal elements. There are different representations proposed for the focal elements in 2D space including discrete bitmap representation in [8], the representation using a set of boxes in [132], as well as the polygon-based representation used in [121]. Among them, the polygon-based representation in [121] provides more precision and flexibility, as well as the full scalability with respect to the cardinality of 2D discernment frame. For our work, we use their open source library 2CoBel ("A scalable belief function representation for 2D discernment frames") using polygon representation to deal with our problem related to 2D epipole localization in image domain. In the following, we briefly introduce the essential parts about this 2D belief function framework, with the process illustrated in Figure 5.1.

5.2.1 Polygons based representation

The 2D focal element is taken as a general geometric region in 2D space and represented by a set of generic polygons.The polygons for each focal element have a closed simple path without crossing among the edges and are represented by a minimum number of vertexes using the coordinates of integer values to ensure the numerical stability. The vertexes are organized in the topological ordering of counter-clockwise for positive areas and clockwise for holes. The basic operations for focal elements during combination such as intersection, union, difference and XOR can be then directly performed on the vertexes of polygons using 2D polygon clipping algorithms developed in [72, 153].

There are several advantages using the polygon representation:

1. The set of vertexes for a set of polygons give a unique and precise representation for a focal element having 2D geometric shape. In our case, the input sources are represented by a confidence region using ellipse, which can be easily approximated by a polygon for geometric computation.

2. The polygon representation allows for focal elements having multiple components or focal elements with hole inside.

3. As the operations are performed on the vertexes of polygons, the computational cost is independent of the cardinality of the discernment frame and only depends on the number of vertexes of polygons. A fast comparison and lookup of focal elements can be furthermore achieved with hashing table.

4. The scalability is guaranteed in terms of basic operations as the coordinates can be rescaled at any desired level of precision under the memory limit of computational hardware.

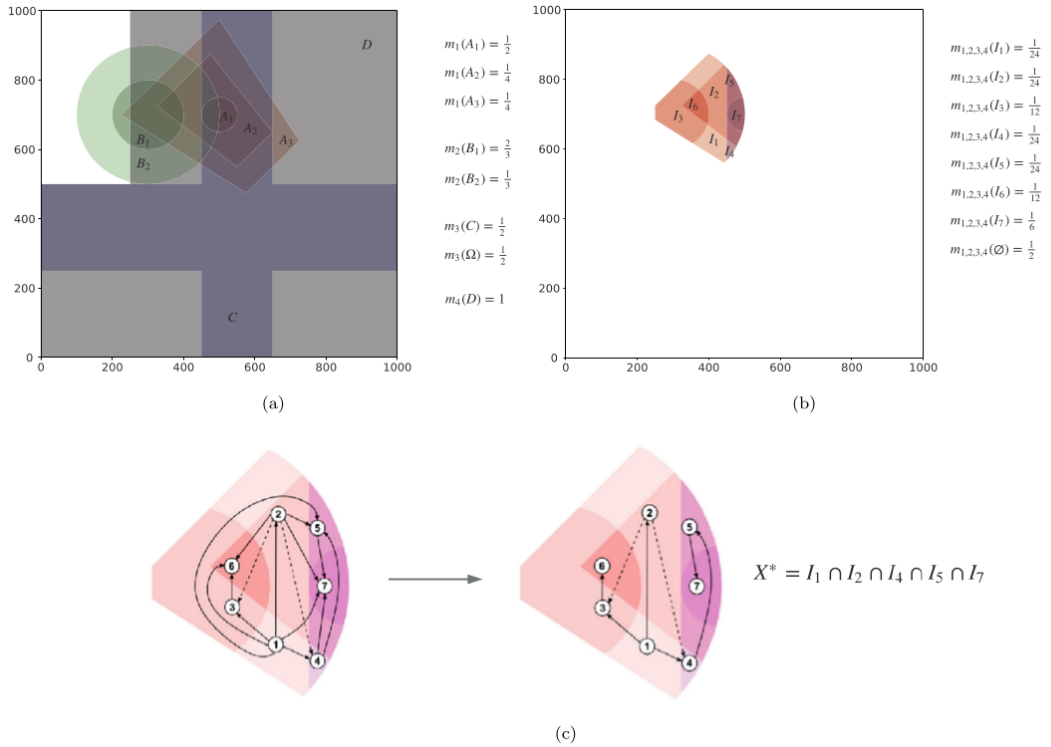


Figure 5.1: Illustration of the framework 2CoBel for 2D localization problem [121]. There are several BBA definitions from different sources for the localization problem, represented by polygons and illustrated with different colors in (a). (b) presents the BBA after the combination rules. (c) illustrate the intersection-inclusion graph, with the solid lines for the inclusion relationship and the dashed lines for the intersection relationship. (c) presents the decision making results X^* with the maximum intersection rules for BetP value.

However, the polygon representation does not guarantee the scalability when it comes to dealing with singleton hypothesis for decision making algorithms.

5.2.2 The intersection-inclusion graph

For efficient decision making, the authors proposed a high-level graph based description for the interactions between focal elements. Given a number of focal elements, represented as sets of vertexes of polygons, denoted by $A = \{A_1, A_2, ..., A_i, ..., A_n\}$ ranked by decreasing cardinality, i.e. $|A_i| \geq |A_j|$ for $i < j$, a direct acyclic graph (DAG) $G = (V, E)$ is built to describe the intersection and inclusion relationships between them. Each node $v_i \in V$ is a focal element A_i and each edge $e_{ij} \in E$ represents a link $v_i \rightarrow v_j$ when the intersection between their correspondent focal elements is not empty, i.e $A_i \cap A_j \neq \emptyset$. Besides, each node v_i is associated with a reference to the lowest l_i^l and highest label l_i^h of the ordered nodes that include the current node.

The intersection-inclusion graph allows us to explore all the possible intersections between focal elements. The objective is to find an subset that is an intersection between a high number of focal elements so that it gathers a maximal sum of mass values (of all the focal elements including it). Let $P_k^{(m)} = \{v_{k,1}, v_{k,2}, ..., v_{k,m}\}$ be a path including a set of nodes in the intersection-inclusion graph $G = (V, E)$, it represents the intersection between all the focal elements represented by nodes $v_{k,j}$.

Definition 5.2.1. *A path $P_k^{(m)}$ is called as **dead** if the intersection among all the focal elements corresponding to its nodes is the empty set.*

Definition 5.2.2. *A path $P_k^{(m)}$ is called as **redundant** if there exists another path leading to the same intersection.*

Definition 5.2.3. *Given two paths* $\mathrm{P}_k^{(m)}$ *and* $\mathrm{P}_h^{(n)}$, $\mathrm{P}_k^{(m)}$ *is called* ***superpath*** *of* $\mathrm{P}_h^{(n)}$ *if* $m > n$ *and:*

$$\forall v \in \mathrm{P}_h^{(n)} \Rightarrow v \in \mathrm{P}_k^{(m)}.$$

Conversely, $\mathrm{P}_h^{(n)}$ *is called* ***subpath*** *of* $\mathrm{P}_k^{(m)}$.

It can be shown that the researched intersection can be only provided by a path which is neither dead nor redundant. The basic idea is to explore such paths through the graph traversal with a depth first search (DFS) strategy by taking each node as the root iteratively. A node is added to the current path if the intersection between the current node and the intersection set obtained from the previously visited node is not empty, otherwise, the DFS for the current path will stop (since obtained path is *dead*). However, such straightforward exploration will be infeasible in the case of consonants BBAs where the graph is complete. The unnecessary operations can be avoided for the redundant paths due to the inclusion relationship. In order to improve the efficiency of the graph traversal, the authors proposed several strategies that avoid *redundant* paths, such as:

1. ***Root suppression***. When the focal element associated to the current node v_i is included in the focal element associated to a previous node v_k in the decreasing cardinality ranking (i.e., $k < i$) any path starting from this current node is redundant as the intersection between the focal elements associated to v_i and v_k gives the focal element associated with v_i. Therefore, one can remove v_i from the root candidates and reduce the computation time.

2. ***Early stopping***. Given a root node v_r and the current node v_i, if v_i is included in another root node v_h, such that $h < r$, any path including v_i is redundant (adding v_h to this path leads to a *superpath*), so that exploration can be stopped.

3. ***Graph simplification***. When a node v_i is included in multiple nodes $\{v_i^h\}_{h=1\ldots m}$, the graph can be simplified by keeping only one of the edges among $\{v_i^h\}_{h=1\ldots m} \rightarrow v_i$ as redundant inclusion relationships will lead to redundant path. It is straightforward to select the edges from the highest indexed node $v_i^m \rightarrow v_i$ because it allows us to not shorten the paths and thus to preserve the maximal length paths This simplification can make the exploration more efficient in case of of multiples inclusions.

5.2.3 Extraction of disjoint sets

Canonical decomposition is a technique which represents a complex BBA as the combination of Simple BBAs or Generalized Simple BBAs. It allows us to introduce Denoeux's cautious rule for the combination process. However, it is not feasible to compute the weight value required for each compound hypothesis, especially in the case of 2D discernment frame. A flexible solution is to transform the 2D BBA representation in a compact 1D equivalent where the Fast Mobius Transform can be applied, as suggested by the authors [121].

In order to allows for a full scalability with the cardinality of the discernment frame as the extraction and transformation are related to the number of focal elements, the authors in [121] propose to decompose the 2D discernment frame into a set of disjoint sets included in the union of all the available focal elements, as illustrated in Figure 5.2. It can be constructed from the graph representation. Each non-redundant path provides an intersection set which can be considered as the candidate for the extraction of disjoint sets. When a new candidate is considered, one can check if it has a non-empty intersection with any of the already stored disjoint sets. If not, the candidate is stored as a disjoint set. Otherwise, the the non-empty intersection is removed (*difference* operator) from the candidate, so that it no longer intersects any already stored disjoint sets and can be added to the list of disjoint sets. Then, from the list of the disjoint sets, a 1D discernment frame is defined such that its singleton elements are the disjoint set. Then, based on the mapping from the set of disjoint sets and the set of original focal elements , the 2D BBAs are transformed in 1D BBAs.

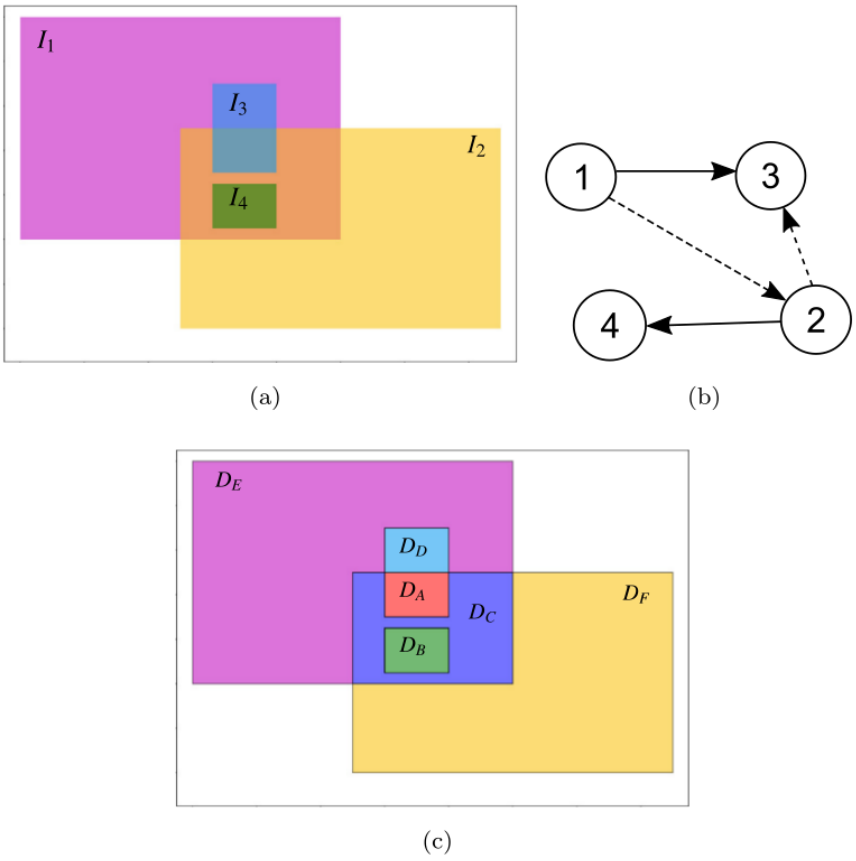


Figure 5.2: Illustration of the extraction of disjoint sets [118]. (a) the original 2D BBA representation with a set of 2D focal elements $\{I_1, I_2, I_3, I_4\}$. (b) The intersection-inclusion graph representation for the focal elements of BBA. (c) The extraction of disjoint sets $\{D_A, D_B, D_C, D_E, D_F\}$.

5.2.4 Maximal intersections based decision making

The general decision making is taken on the singleton hypotheses by maximizing the function that is defined on Ω after BBA transformation, for example the pignistic probability defined in Definition 5.1.19. Due to the additive property for such probability-like function, the maximum value can be only achieved on the elements of the discernment frame with the maximal intersections, which is defined as follows:

Definition 5.2.4. *Given a set of focal elements* $\mathrm{A} = \{\mathrm{A}_1, \mathrm{A}_2, ..., \mathrm{A}_i, ..., \mathrm{A}_n\}$*, the* ***maximal intersection*** I_m *is derived from the subset of* A*, such that any different focal element added to the subset would lead to an empty intersection.*

Definition 5.2.5. *Given the intersection-inclusion graph* G*, the* ***maximal intersection*** I_m *is represented by a non-redundant path which is not a subpath of any other non-redundant path.*

The set of maximal intersections I can either be computed directly by DAG exploration based on efficient graph traversal using the mentioned tricks, or from the 1D representation mentioned in Section 5.2.3. The decision can be then made by maximizing the BetP of the elements inside the set I:

$$\mathrm{X}^* = \underset{\mathrm{I}_m \in \mathrm{I}, \mathrm{I} \subset \mathrm{D}}{\operatorname{argmax}} \frac{\mathrm{B}et\mathrm{P}(\mathrm{I}_m)}{|\mathrm{I}_m|}, \tag{5.8}$$

where the BetP for the compound hypotheses I_m is defined as follows:

$$\mathrm{B}et\mathrm{P}(\mathrm{I}_m) = \frac{1}{1 - m(\emptyset)} \sum_{\mathrm{B} \cap \mathrm{I}_m \neq \emptyset, \mathrm{B} \in 2^{\Omega}} \frac{|\mathrm{B} \cap \mathrm{I}_m|}{|\mathrm{B}|} m(\mathrm{B}). \tag{5.9}$$

Chapter 6

Epipole localization based on 2D belief fusion framework

Contents

6.1 Introduction

In Chapter 4, we have proposed to localize epipole based on uncertainty estimation using the multi-modal sampling strategy. Instead of relying on the best consensual solution which may be unreliable due to the presence of amounts of outliers for the keypoint matches, a number of sub-best solutions are considered as well and summed up to the final prediction map for the epipole location. Although the proposed method with multiple sampling helps to improve the reliability of epipole, the direct accumulation with sum operation suffers from several limitations:

1. The precision is decreased due to the enlargement of the uncertainty region when the number of solutions increases;

2. The reliability may be reduced when the outlier sources achieves the major votes.

In order to overcome these limitations, it is crucial to deal with the outlier solutions when the number of sources is high. A straightforward method is to associate each source with a weight which measures the quality of the source and then perform the weighted accumulation over all available sources. However, it requires prior knowledge about the weights which are usually unavailable in practice. The number of inliers may provide potential cues about the weight information but it is not very useful as the outlier solutions may also achieve high supports.

Among the sub-best solutions, some of them exhibit consistency in terms of epipole location and others not, so that we assume they are more likely to be outliers. Furthermore, there may be different groups of consistent solutions. If we can identify these groups in terms of consistency, some outlier solutions may be discarded, while the candidate regions for epipole location can be restricted to consistent groups. A direct accumulation of all the solutions does not allow us to capture the desired consistency. It requires using a more complex and flexible framework which allows us to measure the consistency between solutions and draw out the consistent groups.

In the following, we focus on the Belief Function Theory (BFT) framework designed to handle the combination of uncertain sources. This formalism was made popular by various real-world applications [38] for which it provides an efficient modelling of imprecise information, allowing for fairer and more consistent decisions. However, for some applications, scalability remains a challenge, either in terms of the size of the discernment frame or in terms of the number of sources to be combined.

Firstly, regarding the size of the Discernment Frame (DF), the issue is that belief functions (mass, plausibility, etc.) are defined on the DF powerset, so that for a DF denoted Ω whose cardinality is $|\Omega|$, there are potentially $2^{|\Omega|}$ hypotheses to consider and to enumerate. Such an issue arises typically in localization applications in which DF corresponds to possible positions of the considered target, i.e typically $|\Omega| = 10^6$ if we require a spatial resolution equal to 10×10 cm^2 within an area of 100×100 m^2. First solutions, e.g. [8], use some tricks (e.g. conditioning) to consider only a subpart of DF at once. Then, [121, 132, 172] propose a solution where 2^{Ω} elements (singleton and compound hypotheses) are no longer labelled (as for enumeration) while each element of the focal set (that is usually a small subset of 2^{Ω}) is handled through its own description. Specifically, in [132, 172], the handled focal elements are described as sets of rectangles (tiles) similarly to the representation used in Interval Analysis [71], whereas [121] provides a more general representation of any 2D shapes using polygons. In both cases, belief function operators based on set relationships (intersection, union etc.) have to be redefined in an efficient way.

Secondly, regarding the number of sources, one difficulty is related to the used combination rule. In particular, using the very popular conjunctive rule proposed by Smets [141], the mass on the empty set ($m(\emptyset)$), usually called conflict, is an increasing function with respect to the number N of combined BF: $m(\emptyset) \to 1$ when $N \to +\infty$. Note that this rule was proposed to avoid evidence modelling issues (e.g. as in the case of the Zadeh example) hidden by mass normalization as performed in the original Dempster's rule or orthogonal sum [140]. Considering alternative rules would not solve the issue. Indeed, some hybrid rules (e.g., those proposed by Yager [165] or Dubois and Prade [42]) performing a dispatching of the conflict are not associative, which in turn may raise additional issues about the combination ordering of the sources. Instead of searching alternatives to the conjunctive combination rule, some authors proposed to discount the Basic Belief Assignments (BBAs) to combine so that the conflict remains under control [123, 178].

Besides the choice of a tailored combination rule, a large number of sources raises the issue of the presence of unreliable ones. Indeed, the higher the number of sources, the more likely is the fact that one or some of them are unreliable. Such sources are called 'outliers' for the combination since they are incompatible/inconsistent with the remainder of the sources. Some authors have proposed algorithms to handle sets of sources including outliers, either by extending the q-relaxation [41] proposed for the Interval Analysis to BFT [123], or by extending RANSAC [48] to BF [172]. In the first case, the combination rule is modified to be robust to the presence of outliers, making it however intractable in the case of a large number of outliers (the q parameter being usually in the range of a few units). In the second case, having explicitly estimated a set of in-

liers (conversely to q-relaxation), the conjunctive rule is used however with a number of sources ranging in the tens.

In [179], the authors propose to handle a large number of sources by clustering BBAs and firstly combining the BBAs that belong to the same cluster. In their work, using the canonical decomposition, clusters are simply defined as sets of Simple Support Functions (SSF) having the same focal elements so that their combination is straightforward and also produces a SSF. Then, intra-cluster resulting SSFs are discounted with respect to the number of initial SSFs in the cluster. However, such an approach assumes that the canonical decomposition of initial BBAs involves a small set of SSFs, so that each element may appear several times when considering the different initial BBAs, which is not the case when considering a large 2D discernment frame. Thus, even if we keep the general idea of BBA clustering that was already proposed by [139], both the clustering criterion and the use of clustering results are different.

In this chapter, our purpose is to provide as accurately and precisely as possible an estimation of the epipole location, while not underestimating residual uncertainty. Specifically, we deal with the combination of a large number of imperfect sources with the presence of outlier solutions within the 2D Belief framework using the 2CoBel library introduced in Chapter 5. To overcome the challenges of efficiency related to large number of sources, we propose to first cluster all considered solutions into different groups in terms of consistency and then explore the fusion strategy within each group. The combination of a large number of sources boils down to several combinations of a smaller number of sources. To respect the consistency inside the group and to highlight the inconsistency between different groups, the core idea is to keep individual solutions provided by each group instead of imposing a final fusion between different groups.

6.2 Problem formulation

Let us recall the pipeline for the proposed epipole localization based on uncertainty estimation in Chapter 4. Given the set of putative point matches $\mathscr{I}$, it follows first the principle of RANSAC to subsample iteratively a set of random 8-tuple of point matches from $\mathscr{I}$. Each sampled 8-tuple of point matches corresponds to a solution for the epipole location and its covariance matrix which are derived from the fundamental matrix based on the constraint $\mathbf{F}\mathbf{e} = 0$ and singular value decomposition (SVD) [115]. Let us consider the set of the n solutions for epipole and its covariance matrix:

$$\mathrm{S}_{\mathbf{e}}^{n} = \left\{(\mathbf{e_1}, \Sigma_{\mathbf{e_1}}), (\mathbf{e_2}, \Sigma_{\mathbf{e_2}}), \ldots, (\mathbf{e_i}, \Sigma_{\mathbf{e_i}}), \ldots, (\mathbf{e_n}, \Sigma_{\mathbf{e_n}})\right\},$$

ranked in decreasing order according to their consensus value corresponding to the size of their inlier set $\mathscr{I}_i$. The considered solutions are the p first elements in $\mathrm{S}_{\mathbf{e}}$ with p derived with respect to threshold $\theta \in (0,1)$ such that

$$\frac{|\mathscr{I}_p|}{|\mathscr{I}_1|} \geq \theta > \frac{|\mathscr{I}_{p+1}|}{|\mathscr{I}_1|}. \tag{6.1}$$

Geometrically, each solution $(\mathbf{e_i}, \Sigma_{\mathbf{e_i}})$ is represented by the shape of an ellipse whose axes are defined by eigenvectors and eigenvalues such that

$$(\mathbf{x}-\mathbf{e})^{\mathrm{T}} \Sigma^{-1} (\mathbf{x}-\mathbf{e}) = \kappa^2,$$

with κ defined by the considered confidence level.

By taking the first p solutions, $\mathrm{S}_{\mathbf{e}}^{p}$ and their associated 2D ellipses as the input, the problem we address here is how to combine them within the 2D Belief framework in order to derive a consistent solution for the epipole location. The basic idea of our approach is then to introduce a mutual validation test for any potential solution, based on the consistency among several solutions obtained independently. Note that this idea is the very core of ensemble approaches that aim to increase estimation robustness and accuracy by combining different algorithm outputs. From the set of the p putative imprecise solutions for the epipole location, we aim to derive a

more accurate localization. However, due to the presence of erroneous keypoint matches, some of these ellipses do not include the true epipole location. Filtering them is all the more complicated that they nevertheless correspond to rather consensual solutions (among the p most consensual ones). However, we hope (and assume in the following) that there exists a subset of them including the true epipole location, and that this subset may be detected based on adequate criteria.

This problem can be related to the general issue raised by the combination of a large number of 2D belief function assignments for localization in 2D domain. Supposing the first p solutions $S_{\mathbf{e}}^{p}$ are represented by a set of BBAs, denoted by $\mathscr{M} = \{m_i\}_{1\leq i\leq p}$, where m_i corresponds to the belief function for the solution $(\mathbf{e_i}, \Sigma_{\mathbf{e_i}})$, a direct combination with the conjunctive rule does not allow to explore the consistency among different groups of BBAs. The basic idea is to follow the method based on BBA clustering proposed in [179] which is initially developed for 1D BBAs. With the clustering, the set of BBAs is divided into several groups gathering smaller numbers of BBAs, which leads to the reduce of computation cost in [179]. However, when it comes to 2D BBAs, the clustering criterion in [179] based on the same elements of the canonical decomposition is not feasible anymore due to the size of discernment frame. It requires to explore the clustering of 2D BBAs in terms of more flexible consistency measures.

Focused on the 2D Belief framework using 2CoBel library, we represent the BBA for pieces of evidence for epipole location with the 2D polygones (as approximations of ellipses). The considered discernment frame (Ω) is the set of the location solutions for the epipole, that is to say the plane containing the static camera image plane (which can be bounded to the image frame or not depending on the camera setting). Using the 2CoBel scalability, sub-pixel coordinates are handled. Then, for each solution, a consonant BBA is derived whose nested FEs are 2D polygons approximating the ellipses corresponding to different values of confidence level κ in Equation 4.28. Specifically, denoting by n_{FE}^{0} the number of FEs, in our experiments, the default setting is $n_{\mathrm{FE}}^{0} = 2$, with uncertainty levels corresponding to 50% and 95% ($\kappa^2 = 1.386$ and $\kappa^2 = 5.991$, respectively) by referring to [83, 128], with equally distributed mass. These dogmatic BBAs ($m(\Omega) = 0$) leads to our preference for discarding poor solutions (i.e., with the true epipole outside of the biggest focal element) rather than involving them in the fusion process. Indeed, these poor solutions will not bring relevant information for actual epipole location so that we propose to filter them in order to focus on smaller but relevant sets of sources based on clustering.

In terms of clustering, we first explore in the following part the available clustering algorithms as well as the consistency measure between 2D BBAs related to ellipses. Then, we adapt them to the proposed belief clustering method to replace the direct accumulation of multiple sampled solutions for epipole location. We also explore the combination with the additional sensor sources which are useful to guide furthermore the localization of epipole.

6.3 Clustering algorithms

The grouping of 2D BBAs involves the choice of a clustering algorithm. In general, clustering is regarded as a machine learning method to classify a set of data points into different groups or clusters such that data points in the same group share similar features and differ from data points in different groups in terms of specific metrics. To extend the clustering from data points to 2D BBAs associated with the uncertain pieces of evidence for the epipole localization, we need to choose a clustering algorithm along with a similarity or dissimilarity metrics between BBAs. In this section, we first introduce the available clustering algorithms briefly and then discuss how to choose the appropriate one for our problem. The dissimilarity metrics will be discussed furthermore in Section 6.3.3.

6.3.1 Choice of clustering algorithm

Many works have been devoted to the problem of clustering. An overview of such clustering algorithms can be found in several articles [47, 99]. Depending on the process, the clustering algo-

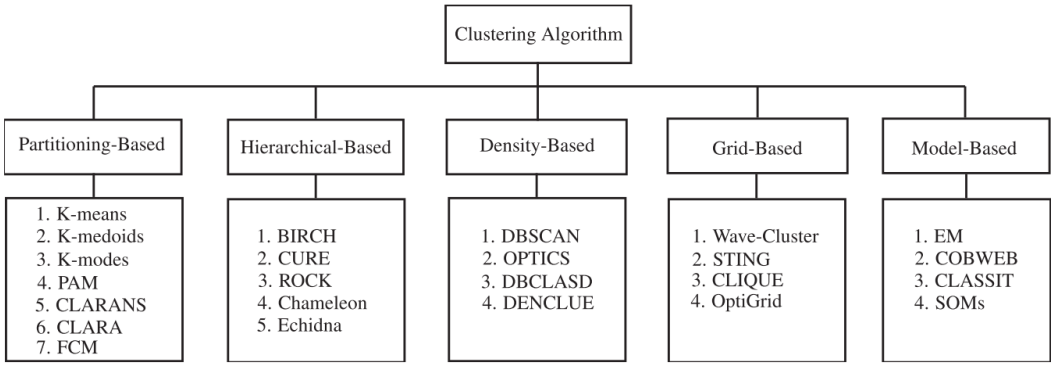


Figure 6.1: Overview of clustering algorithm [47].

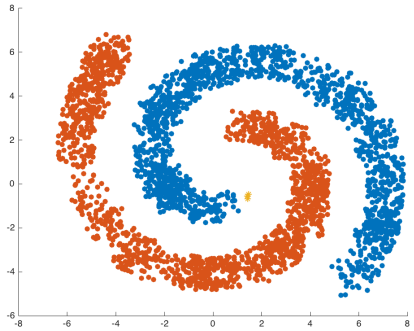
rithms can be roughly divided into five classes including partitional clustering, model-based clustering, density-based clustering, grid-based clustering, and hierarchical clustering, summarized in Figure 6.1.

The partitional clustering algorithms like the k-means algorithm [93] and k-medoids algorithm [130], are based on iteratively reallocating the groups of data points until to the convergence of group assignment. Specifically, given the initial group centers, it computes the distance of data points to each center and assigns the cluster label for each point by minimizing the distance between the point and the group center. The new group centers are recomputed according to the choice of method. The process is repeated until the convergence of group centers. Despite the advantage of fast computation by determining all clusters at once, the requirement of initialization for the number of clusters as well as the centers makes this kind of algorithms not suitable for our problem in which the number of clusters is impossible to guess a priori. In addition, the clustering results may be not robust to the initialization of cluster centers.

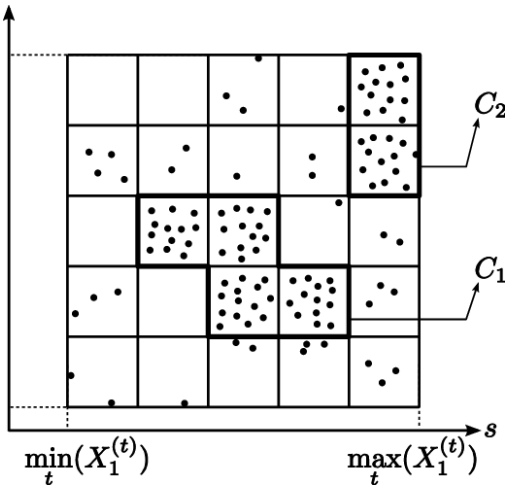
The model-based clustering is based on the data fitting for some mathematical models. The data is generated from a mixture of probability distributions which are usually Gaussian Mixture Models (GMMs). A component in the gaussian mixture models is represented by a mean vector and a covariance matrix. The parameters of models are estimated by maximizing the likelihood for the data being generated from such assumed models. The model-based clustering is more flexible in terms of the shape of cluster compared to the K-means which is restricted to circular clusters. However, it requires an appropriate model assumption for the object, which may be not available for a complex objects, such as the BBAs.

The density-based methods are designed to achieve the grouping of arbitrary-shape clusters affected by the noise. Based on the density of points, DBSCAN is one of the typical density-based methods. The density of data points is defined as the number of data points whose distance from the considered point is less than a defined threshold ϵ. The first step of DBSCAN is to identify the core points which has a high density larger than a threshold m. The rest of the points are then assigned to the cluster represented by each core point if the distance from the considered point and the core point is less then the threshold ϵ. An isolated point which is not included in any cluster related to a core point is identified as an outlier point. Different clusters will be merged if their core points are close enough which means the distance between them is less than ϵ. An example of DBSCAN clustering is illustrated in Figure 6.2a. Like partitional clustering algorithms, DBSCAN also has the advantage of fast computation. In addition, it does not require knowing the number of clusters and can be robust to noises. However, the objects in the same cluster are connected with the density relationship which is not the desired characteristic in our problem. To handle the combination of different BBAs in the same clusters, it requires the BBAs in the same clusters to be consistent with all other objects, not only parts of them.

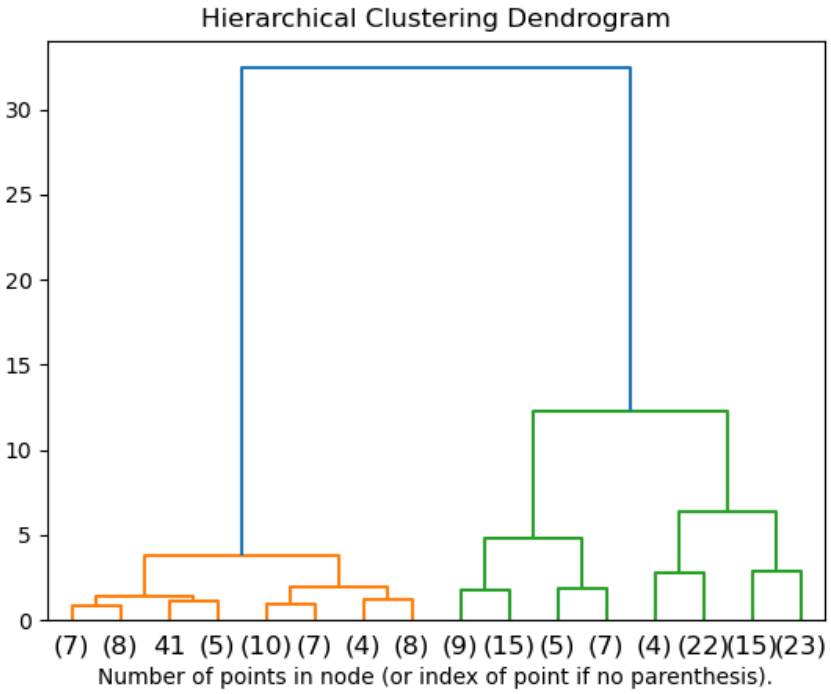
The grid-based methods are based on the assignment of data points to grid cells constructed from the division of the space of the data objects, as illustrated in Figure 6.2b. The clusters are



(a) Example of DBSCAN clustering.



(b) Example of Grid-based clustering.



(c) Example of hierarchical clustering.

Figure 6.2: Illustration of different clustering methods

generated by accumulating the adjacent cells whose density is larger than a defined threshold. As the clustering is performed at the grid level, the computation time depends only on the size of grid instead of the dataset cardinality. Thus it can be an efficient tool to deal with large datasets. However, it is not feasible to construct the grid cell for objects in high dimension, or complex objects like ellipses or mass functions.

Finally, we come to hierarchical clustering which provides an appropriate clustering tool for our problem due to several advantages. Firstly, it does not require the knowledge of the number of clusters like partitional algorithms, or an explicit model assumption like model-based methods. Secondly, unlike the grid-based methods, it can deal with complex objects by feeding the distance matrix. In addition, the desired consistency among all objects in the same cluster can also be achieved with the complete linkage between objects. More details about hierarchical clustering can be found in the following.

6.3.2 Hierarchical clustering

The hierarchical clustering is a family of clustering algorithms representing the object relationships by a tree or a dendrogram and finding clusters in an iterative way. According to the processing order, the hierarchical clustering algorithms can be divided between agglomerative (bottom-up) or divisive (top-down) ones. The divisive algorithms are based on the successive split of a set of data points into smaller clusters. Inversely, the agglomerative approaches start from individual data points as the leaf nodes and successively merge them to form clusters. Cluster split and merge ar usually based on a distance measure. Then, hierarchical clustering usually refers to the agglomerative algorithms.

Given a set of objects X = $\{x_1, x_2, ..., x_i, ..., x_n\}$, the agglomerative algorithm steps are described as follows:

1. Create a set of clusters C = $\{c_1, c_2, ..., c_i, ..., c_n\}$, where each cluster includes a single object, denoted by $c_i = \{x_i\}$;
2. For any pairs of clusters inside C, compute the distance between two clusters according to the selected distance measure and linkage criterion, and then find the pair of clusters (c_i, c_j) with the minimal distance. Merge the selected pair of clusters (c_i, c_j) into one cluster and add the merged cluster to C after removing c_i and c_j;
3. Repeat the Step 2 until there is only one cluster inside C, representing the root of the tree.

The critical part is the used distance measure and the linkage criterion to decide which clusters should be merged in Step 2. The distance measure deals with a pair of objects which will be discussed in Section 6.3.3. The linkage criterion determines the distance between clusters based on previous object pair distance function. Given two clusters c_i and c_j and the distance measure between objects $d(\cdot,\cdot)$, the choices for the linkage criterion are:

- The **Single Linkage** computes the distance between clusters based on the distance between the closest objects in clusters:

$$\mathrm{D}(c_i, c_j) = \min_{x_p \in c_i, x_q \in c_j} d(x_p, x_q). \tag{6.2}$$

  Single linkage is thus also called the nearest neighbor clustering. It performs a loose linkage between clusters since apart from the two closest elements that determine cluster distance, the others elements may be far from each other.

- The **Complete Linkage** computes the distance between clusters based on the distance between the furthest objects in clusters:

$$\mathrm{D}(c_i, c_j) = \max_{x_p \in c_i, x_q \in c_j} d(x_p, x_q), \tag{6.3}$$

Complete linkage is thus also called the maximal linkage. It performs a tight linkage between clusters since by controlling the maximum distance between pair of elements, it allows for clusters in which the objects are closed to each other.

- The **Average Linkage** computes the distance between clusters based on the average distance of all pair of objects in clusters:

$$D(c_i, c_j) = \frac{\sum_{x_p \in c_i, x_q \in c_j} d(x_p, x_q)}{|c_i||c_j|}. \tag{6.4}$$

- The **Centroid Linkage** computes the distance between clusters based on the distance between the cluster centroids:

$$D(c_i, c_j) = d\left(\frac{\sum_{x_p \in c_i} x_p}{|c_i|}, \frac{\sum_{x_q \in c_j} x_p}{|c_j|}\right). \tag{6.5}$$

- The **Ward's Linkage** minimizes the increase of variances after the merge of two clusters:

$$D(c_i, c_j) = \sum_{x \in c_i \cup c_j} d^2(x, \mu_{c_i \cup c_j}) - \sum_{x \in c_i} d^2(x, \mu_{c_i}) - \sum_{x \in c_j} d^2(x, \mu_{c_j}), \tag{6.6}$$

where μ_{c_i} is the centroid of c_i , μ_{c_j} the centroid of c_j and $\mu_{c_i \cup c_j}$ for the merged cluster $c_i \cup c_j$.

Among these linkages, in the perspective of conjunctive combination of BBAs, we consider the complete linkage as it provides the most compact clusters which allows us to satisfy the requirement that all BBAs in the same cluster should be consistent with each other.

6.3.3 Distance measure

We have discussed different clustering algorithms and conclude that the agglomerative hierarchical clustering with the complete linkage is the best suited for our problem. In [98], a Hierarchical Ascendant Clustering (HAC) was proposed for the clustering of objects having imprecise and uncertain features. Then the authors define BBAs representing the belief that two objects belong to a same or a different cluster. Such a problem is thus rather different from ours whose objective is to derive some subsets of compatible sources (non conflictual epipole solutions represented in terms of BBAs) for data fusion. We are leveraging the fact that BBAs are already defined for our location problem and cluster them in the perspective of their fusion, e.g., controlling the conflict degree in each cluster of BBAs. In our case, we aim to achieve this based on a well-chosen BBA distance.

To choose the distance measure between objects, let us recall that the considered object in our problem is the epipole location and its covariance matrix, denoted as $(\mathbf{e_i}, \Sigma_{\mathbf{e_i}})$. The uncertainty of epipole location can be represented as a 2D Gaussian distribution in probability framework, or a 2D ellipse for the confidence region in geometric framework, or a BBA (basic belief assignment or a mass function) in belief function theory. Depending on the framework, the distance between $(\mathbf{e_i}, \Sigma_{\mathbf{e_i}})$ and $(\mathbf{e_j}, \Sigma_{\mathbf{e_j}})$ can be measured in different ways including probability-based distance such as KL-divergence or the Wasserstein distance, the geometric distance between two ellipses and the BBA-based distances. To be coherent with the Belief function framework, we focus on BBA-based distances, more specifically Jousselme's distance (see 5.1.14).

6.4 Proposed belief clustering

Up to this point, we have presented the choice of clustering algorithm for 2D BBAs object as well as the involved distance metric. In the following, we present the steps to perform the proposed belief clustering and fusion as illustrated in Figure 6.3 .

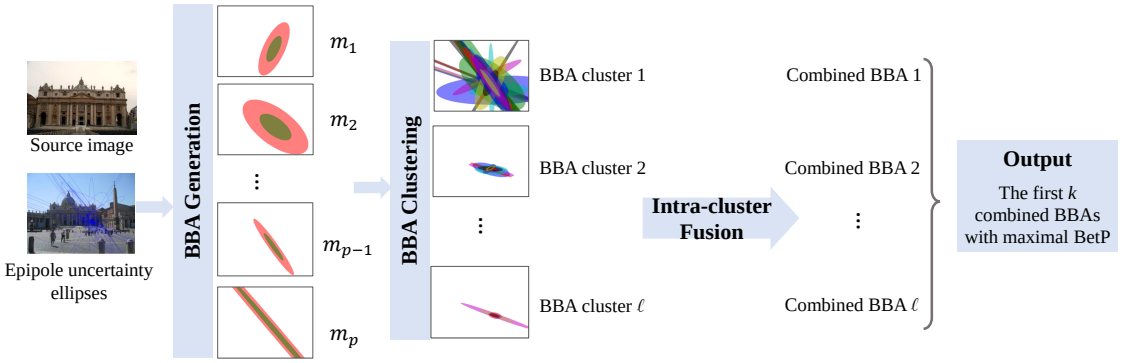


Figure 6.3: Overview of the proposed method. First, the p sources for epipole uncertainty estimation are generated based on the observations with the multi-modal sampling strategy. Each solution is represented by a 2D BBA with two consonant focal elements associated with respectively 95% and 50% confidence level. These BBAs are then clustered by aggregation in l groups. Each group provides a solution as a combined BBA in a set which is ranked according to the pignistic probability BetP. In our application, we select the top k solutions.

6.4.1 BBA clustering

Given a set of BBAs $\mathscr{M} = \{m_1, m_2, ..., m_p\}$ derived from a set of solutions associated to epipole and its covariance matrix, we first cluster them using the agglomerative hierarchical clustering with the complete linkage and the distance matrix $\mathrm{D}_{p\times p}$, where $\mathrm{D}_{ij} = d_\mathrm{J}(m_i, m_j)$ pre-computed using Jousselme's distance, as mentioned in Section 6.3. We then compute the theoretical threshold to apply on Jousselme's distance so that we guaranty that two BBAs have at least one pair of focal elements intersecting (avoiding total conflict). For two BBAs m_1 and m_2 having two nested focal elements of respective mass values a and $(1-a)$ and area ratio equal to $\kappa_{50}^2/\kappa_{95}^2$ (ellipses at 50% and 95% confidence levels), Jouselme's distance between them in the case of conflict degree equal to 1 is

$$d_{th} = \sqrt{a^2 + (1-a)^2 + 2a(1-a)\frac{\kappa_{50}^2}{\kappa_{95}^2}}. \tag{6.7}$$

Proof. If m_1 and m_2 are totally conflictual, $\sum_{\mathrm{A}\in 2^\Omega} \sum_{\substack{\mathrm{B}\in 2^\Omega, \\ \mathrm{A}\cap\mathrm{B}=\emptyset}} m_1(\mathrm{A})\, m_2(\mathrm{B}) = 1$.

Under this hypothesis, it follows that

$$\forall(\mathrm{A},\mathrm{B}) \in 2^\Omega \times 2^\Omega, m_1(\mathrm{A})\, m_2(\mathrm{B}) > 0 \Rightarrow \mathrm{A}\cap\mathrm{B} = \emptyset.$$

Therefore, $\langle m_1, m_2\rangle = 0$.
Besides, since m_1 has only two nested focal elements,

$$\forall(\mathrm{A},\mathrm{B}) \in 2^\Omega \times 2^\Omega, m_1(\mathrm{A})\, m_1(\mathrm{B}) > 0 \Rightarrow \frac{|\mathrm{A}\cap\mathrm{B}|}{|\mathrm{A}\cup\mathrm{B}|} = \begin{cases} 1 & \text{if } \mathrm{A} = \mathrm{B}, \\ \frac{k_{50}^2}{k_{95}^2} & \text{otherwise.} \end{cases}$$

Then, $\langle m_1, m_1\rangle = a^2 + 2\frac{\kappa_{50}^2}{\kappa_{95}^2}a(1-a) + (1-a)^2$ and it is the same for $\langle m_2, m_2\rangle$. Therefore,

$$\begin{aligned}\frac{1}{2}\left(\langle m_1, m_1\rangle + \langle m_2, m_2\rangle - 2\langle m_1, m_2\rangle\right) &= \langle m_1, m_1\rangle \\ &= a^2 + 2\frac{\kappa_{50}^2}{\kappa_{95}^2}a(1-a) + (1-a)^2\end{aligned}$$

□

In our experiments, we set the maximum distance for clustering slightly below the theoretical value (namely, $d_{th} - 0.05$) in order to increase the consistency between BBAs in the same cluster.

In the perspective of conjunctive combination of all the BBAs belonging to the same given cluster, the complete linkage which uses the *max* operator for computing the distance between two clusters from the distance values between samples allows us to bound the distance between any pair of BBAs we will combine during the intra-cluster combination step. However, even a complete linkage cannot guarantee that there is a common intersection for all BBAs in the same cluster (since only pairs of BBAs are considered). Hence, we set as a supplementary constraint that the intersection between all the largest focal elements of the BBAs within a given cluster is not empty. Indeed, for any set of j BBAs $\{m_i\}_{1\leq i\leq j}$, $\textcircled{\cap}_{m_i}(\emptyset) < 0 \iff \exists(\mathrm{A}_1,\ldots,\mathrm{A}_j) \in \mathscr{F}_{m_1}\times\cdots\times\mathscr{F}_{m_j}$ such that $\bigcap_{1\leq i\leq j}\mathrm{A}_i \neq \emptyset$. If the BBAs are consonant, the m_i FEs are nested, and a non empty intersection with a FE implying a non-empty intersection with any FE including it, we have only to check the existence of at least one non-empty intersection between the largest FEs.

The clustering criterion boils down to the minimisation of both the cluster number and the intra-cluster distance under two constraints, namely the intra-cluster distance being lower than $d_{th} - 0.05$, and the non empty intersection between $\bigcup_{m_i} = \bigcup_{\mathrm{A}\in\mathscr{F}(m)}\mathrm{A}$ for all m_i in the cluster. It allows us to derive consistent BBA clusters with respect to the conjunctive combination. In the following, let l denote the number of obtained BBA clusters and $\{\mathscr{M}_1,...,\mathscr{M}_i,...,\mathscr{M}_l\}$ the set of clusters with (i) $\mathscr{M}_i \cap \mathscr{M}_j = \emptyset, \forall\left(i,j\right) \in \{1,\ldots,l\}^2, i \neq j$ and (ii) $\cup_{i\in\{1,\ldots,l\}}\mathscr{M}_i = \mathscr{M}$.

6.4.2 Intra-cluster fusion

From the partition $\{\mathscr{M}_1,...,\mathscr{M}_i,...,\mathscr{M}_l\}$ of the set of BBAs $\mathscr{M}$, the BBAs within each cluster shall be combined using the conjunctive rule [141]. This rule assumes cognitive independence between BBAs which here comes from the independence between RANSAC solutions (corresponding to different 8-tuples, i.e., different linear systems). Note that cognitive independence does not prevent BBAs to be similar, which is all the more expected for BBAs representing the ground truth epipole.

However, for clusters including more than a few tens of BBAs, a step of BBA approximation has to be implemented (cf. Section 5.1.3) to control computational complexity. Specifically, we perform BBA approximation each time the number of focal elements is larger than n_{FE}^{max} and decrease it to n_{FE}^{sum}, with typically $n_{\mathrm{FE}}^{max} = 20$ and $n_{\mathrm{FE}}^{sum} = 10$. The used BBA approximation process is the same as in [8].

Now, one issue introduced by the BBA approximation is the loss of the associativity of the combination which induces a dependency of the result with respect to the combination order. Different strategies have thus been explored. On the one hand, it may appear as more natural to gather first the closest BBAs (still according to Jousselme's distance) so that the combination ordering follows the distance one. On the other hand, one may remark that, since BBA approximation decreases BBA commitment, the BBAs combined in the end may have a greater impact in the final BBA. One can also wonder whether it is worth recomputing the distance with updated cluster BBA after each combination or if a preordering can be defined from the initial BBA distances.

In this study, we have experimented those different strategies and the two more efficient are presented in Section 6.4.3. They correspond to updated minimal (respectively maximal) distance ordering. Let us define the intersection between two BBAs by

$$\mathscr{F}(m_i) \cap \mathscr{F}(m_j) = \{\mathrm{A}\cap\mathrm{B}\}_{(\mathrm{A},\mathrm{B})\in\mathscr{F}(m_i)\times\mathscr{F}(m_j)}.$$

Then, fusion is performed as follows. The two first BBAs to combine are:

$$\left(i^*, j^*\right) = \underset{\substack{(i,j)\in\mathscr{F}(m_i)\times\mathscr{F}(m_j)\\ \mathscr{F}(m_i)\cap\mathscr{F}(m_j)\neq\emptyset}}{\arg\min} d_{\mathrm{J}}(m_i, m_j); \tag{6.8}$$

whereas the next BBA to combine to the current BBA combination result $\tilde{m}$ is

$$j^* = \underset{\substack{j\in\mathscr{F}(m_j)\\ \mathscr{F}(\tilde{m})\cap\mathscr{F}(m_j)\neq\emptyset}}{\arg\min} d_{\mathrm{J}}(\tilde{m}, m_j); \tag{6.9}$$

Equations 6.8 and 6.9 correspond to *min* ordering. The *max* ordering is obtained replacing argmin by argmax in them.

After this intra-cluster fusion step, the l cluster BBAs $\{\tilde{m}_i\}_{i\in\{1,\dots,l\}}$ are ranked according to their maximum $BetP_i$ value or Pl_i value: $max_{A\in 2^\Omega} f_i(A)$ with $f \in \{BetP, Pl\}$ since 2CoBel [121] handles decisions on compound hypotheses. Recalling that our aim is to provide a set (as small as possible) of solutions (possibly under the form of BBAs) including the ground truth, we consider the k first cluster BBAs as the *proposed solution set*. The proposed approach is summarized in Algorithm 2.

Algorithm 2 : 2D belief clustering for epipole localization

Input: I_1 and I_2: A pair of images from two different views; m, θ: Multiple Model sampling parameters; n_{FE}^0: initial FE number; n_{FE}^{max}, n_{FE}^{sum}: summarization parameters; k: *proposed solution set* cardinality

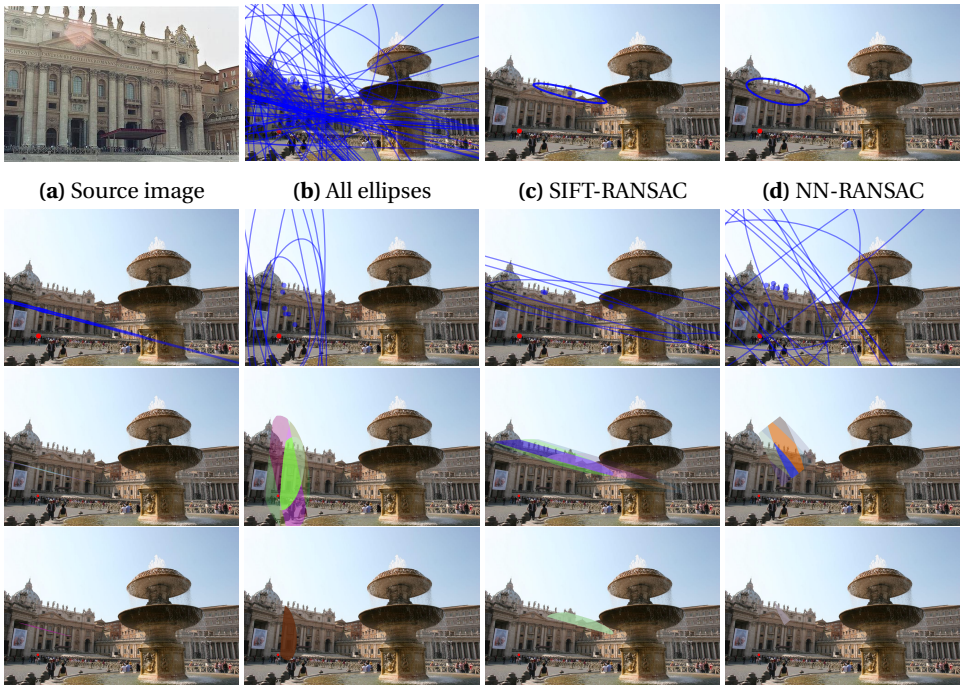
Output: The *proposed solution set* $\tilde{\mathscr{M}} = \{\tilde{m}_1, \dots, \tilde{m}_k\}$ of BBAs representing imprecise solutions for epipole location

1: Extract putative correspondence set $\mathscr{P} = \{p_1, \dots, p_n\}$ between I_1 and I_2
2: Run RANSAC algorithm and, during it, select the m models with the largest inlier support, and rank them in $\mathscr{S}_{\mathbf{F}}^0 = \{\tilde{\mathbf{F}}_1, \dots, \tilde{\mathbf{F}}_m\}$
3: $p = \text{argmax}_{l\in\{1,\dots,m\}} \, l \times \mathbb{1}_{|\mathscr{I}_l|\geq\theta\times|\mathscr{I}_1|}$ ($\mathbb{1}_Z$ is the indicator function of set Z)
4: $\mathscr{S}_{\mathbf{F}} \leftarrow$ the p first elements of $\mathscr{S}_{\mathbf{F}}^0$
5: Initialize $\mathscr{M} = \emptyset$;
6: **for** $\tilde{\mathbf{F}}_\mathbf{i}$ in $\mathscr{S}_{\mathbf{F}}$ **do**
7: Estimate epipole $\mathbf{e_i}$ (using SVD) and covariance matrix $\Sigma_{\mathbf{e_i}}$
8: Build the 2D BBA m_i with n_{FE}^0 nested FEs corresponding to preset confidence levels κ in Equation 4.28
9: $\mathscr{M} \leftarrow \mathscr{M} \cup \{m_i\}$
10: **end for**
11: Compute d_{th} from Equation 6.7
12: Set of clusters $\{\mathscr{M}_1, \dots, \mathscr{M}_l\} \leftarrow$ output of HAC (complete linkage, Jousselme's distance, $d_{th} - 0.05$ threshold, input $\mathscr{M}$)
13: Initialize $\tilde{\mathscr{M}} = \emptyset$
14: **for** $i = 1$ to l **do**
15: $\tilde{m}_i = \textcircled{\cap}_{m_j\in\mathscr{M}_i} m_j$
16: $\tilde{\mathscr{M}} \leftarrow \tilde{\mathscr{M}} \cup \{\tilde{m}_i\}$
17: **end for**
18: Rank BBas in $\tilde{\mathscr{M}}$ according to max value of chosen decision criterion ($BetP$ or pl) and only keep in $\tilde{\mathscr{M}}$ the k first top-ranked BBAs

6.4.3 First experiments and results

To evaluate the performance of the proposed method, we selected 128 pairs of images with various pose variations from the dataset *Saint peters square* (cf. Section 2.1.3). The number of inliers between each pair of image is at least larger than 15 and above 20% of putative matches. We compare the proposed method with the standard RANSAC method based on traditional features (**SIFT-RANSAC**) and based on the learned features (**NN-RANSAC** [167]). From their results, the epipole uncertainty is derived as in [26] ("Least squares SIFT" and "Least squares NN").

For each pair of images, we derive multiple pieces of evidence (regarding the epipole location along with its uncertainty ellipse) by considering various solutions provided by RANSAC instead of retaining only the most consensual hypothesis. The number of iterations for sampling point matches during RANSAC is set to $n = 10^5$. The number of considered sources p is set to be at most 100 (as long as their inlier support satisfies the condition depending on θ). Under this bound, p exact value is determined by θ ($p = f(\theta)$). In our experiments, we study the result sensitivity to θ and k and infer some guidelines on setting them. For hierarchical clustering, we use the **Agglom-**



(a) Source image **(b)** All ellipses **(c)** SIFT-RANSAC **(d)** NN-RANSAC

(e) BetP = 0.630 (rank 1) **(f)** BetP = 0.626 (rank 2) **(g)** BetP = 0.503 (low rank) **(h)** BetP = 0.299 (low rank)

Figure 6.4: Qualitative illustration of our method. Upper row: the source image (a), set of epipole uncertainty ellipses (b) and the result of existing methods (c)-(d) (the ground truth is highlighted in red); Next three rows: for a given cluster, the corresponding original ellipses (first raw), the final BBA (second raw) and the maximum BetP element; cases of (e): the top ranked cluster, (f): the second ranked cluster, (g)-(h): two clusters with a low rank/BetP due to the sources being less consistent.

Table 6.1: Number of image pairs on which the respective method (left column) contains the ground truth epipole. For the proposed fusion, we present results with consensus threshold values $\theta = 0.9$ and $\theta = 0.5$.

Method	#Image pairs including the ground truth					
	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$
SIFT-RANSAC	39	-	-	-	-	-
NN-RANSAC	63	-	-	-	-	-
Fusion-min ($\theta = 0.9$)	31	44	55	58	63	65
Fusion-max ($\theta = 0.9$)	30	44	55	58	64	65
Fusion-min ($\theta = 0.5$)	23	40	47	49	56	60
Fusion-max ($\theta = 0.5$)	23	39	48	50	56	60

(a) Source image **(b)** All ellipses **(c)** SIFT-RANSAC **(d)** NN-RANSAC

(e) BetP = 0.791 (rank 1) **(f)** BetP = 0.724 (rank 2) **(g)** BetP = 0.567 (low rank) **(h)** BetP = 0.432 (low rank)

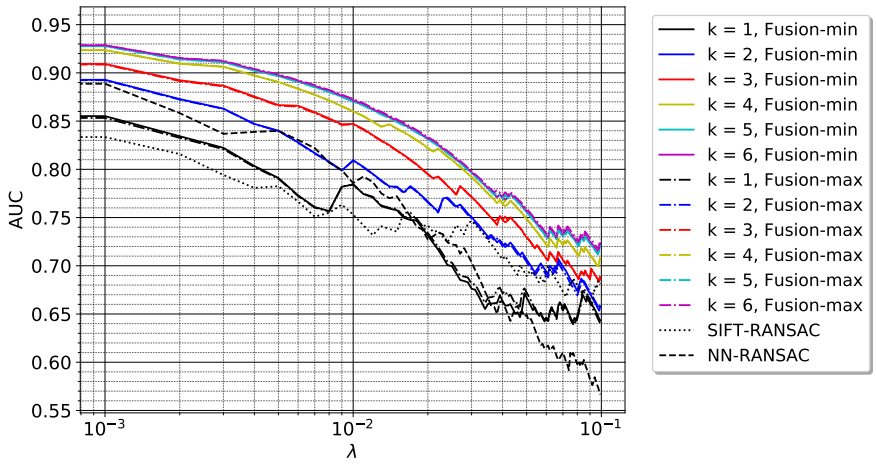
Figure 6.5: Qualitative illustration of our method. Upper row: the source image (a), set of epipole uncertainty ellipses (b) and the result of existing methods (c)-(d) (the ground truth is highlighted in red); Next three rows: for a given cluster, the corresponding original ellipses (first raw), the final BBA (second raw) and the maximum BetP element; cases of (e): the top ranked cluster, (f): the second ranked cluster, (g)-(h): two clusters with a low rank/BetP due to the sources being less consistent.

erativeClustering function of the module scikit-learn [116] with the complete linkage. Now, since this function does not allow for applying an additional binary constraint, we introduce the "non-empty intersection" constraint (between disjunctions of respective focal element sets) a posteriori during the fusion step based on *min/max* distance ordering, as mentioned in Section 6.4.2 (**Fusion-min/max**).

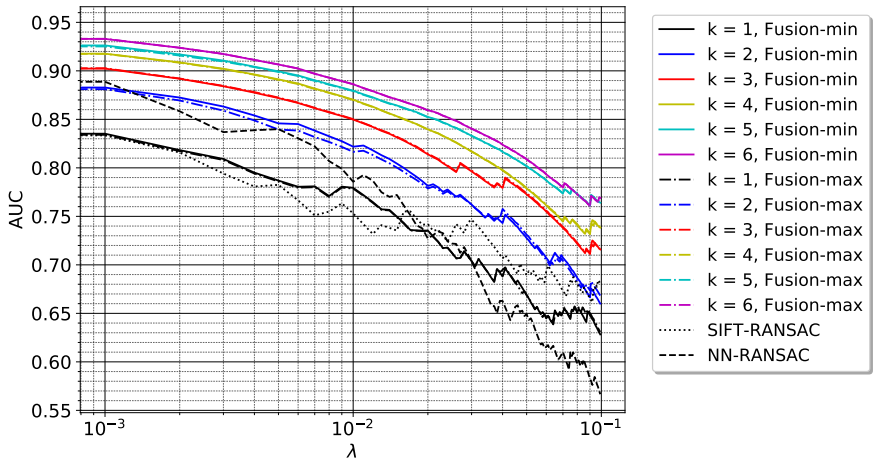
Figures 6.4 and 6.5 illustrate some qualitative localization results, provided by existing methods, by top-ranked clusters and by low-ranked clusters respectively. It illustrates that in difficult settings, the existing methods tend to be overconfident. Top ranked clusters exhibit a higher consistency among the BBAs which results in a strong ellipse alignment, and even for challenging poses the true solution is present at the top. However, we notice that the first-ranked BBA may fail in providing the right epipole location conversely to the second-ranked one, underlying the benefit of keeping several cluster BBAs for fusion with addition sources. We also see that the low rank clusters consist in uncertainty areas exhibiting less consistency that can thus be harmlessly discarded. Finally, these examples also show the importance, for further fusion, to keep the whole BBA and not only the BetP solution since a too early decision may miss the actual epipole location (not included in the BetP selection in Figure 6.4).

Since our output is in the form of a BBA (or an ordered set of BBAs) whereas RANSAC outputs are in the form of uncertainty ellipses (associated to 2D Gaussian distributions), for a fairer quantitative evaluation, we convert these latter in a BBA. Specifically, we derive consonant BBAs having five equi-weighted focal elements, corresponding to the ellipses associated with respectively 95%, 75%, 50%, 25%, 10% confidence levels. We consider two different metrics in our experiments in terms of performance related to accuracy and precision.

The first one counts the number of image pairs in which the ground truth epipole is included



(a) Consensus threshold θ = 0.9



(b) Consensus threshold θ = 0.5

Figure 6.6: Curve of AUC for cumulative curve, versus $\epsilon(\lambda)$.

in at least one focal element of the considered BBA. Specifically, for the proposed method, if the ground truth epipole is included in at least one focal element of one of the k first pieces of evidence, we consider it as positive. The result is summarized in Table 6.1 for different values of k varying from 1 to 6 which ranks the variable number of clusters obtained for each image pair. According to this table, choosing higher values for the θ parameter allows for slightly better results. We interpret such result as supporting the assertion that RANSAC filtering is beneficial, even if we relax the assumption that it is optimal. Secondly, we note the very low sensitivity of the results to the criterion min or max in the fusion. Thirdly, concerning the k parameter, the increasing and asymptotic behaviour of the number of image pairs including the ground truth is clearly visible. Note also that due to the difficulty of the geometry on some image pairs, the upper bound for performance is equal to 81 (i.e. we check that among the 128 image pairs, in 47 of them less than 2 ellipses among the p estimated from $\mathrm{F}_i, i \in \{1,\ldots,p\}$ solutions include the ground truth). Finally, we specify that the results obtained using NN-RANSAC are biased by the fact that this latter has been trained on the same dataset and that much less performant results have been obtained considering other datasets (as shown in Section 6.5.4).

The second quantitative evaluation considers the modified metric [8]

$$\epsilon(\lambda) = \sum_{\mathrm{A}\in 2^{\Omega}} d(\mathbf{e_{gd}}, \mathrm{A})\, m(\mathrm{A}) + \lambda \sum_{\mathrm{A}\in 2^{\Omega}} |\mathrm{A}|\, m(\mathrm{A}), \tag{6.10}$$

where $\lambda \in \mathbb{R}_{\geq 0}$ is a weighting parameter between terms, $\mathbf{e_{gd}}$ is the ground truth epipole location

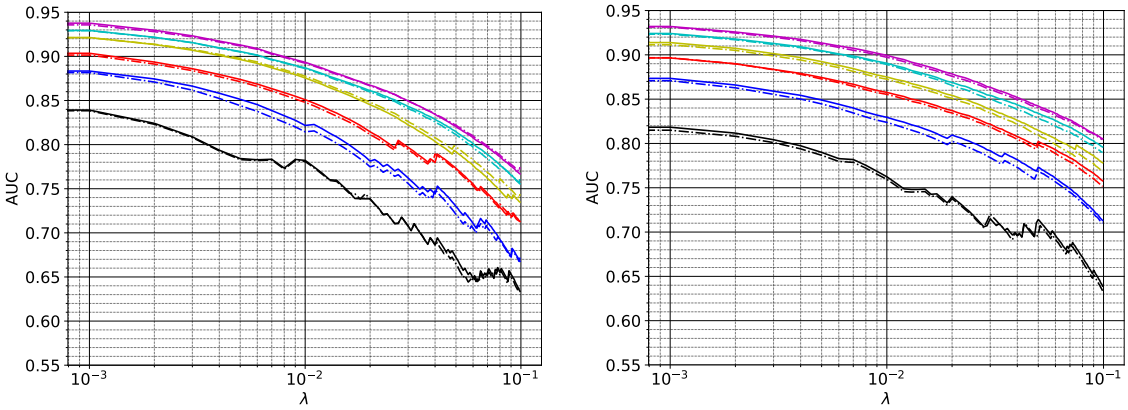
and $d(\mathbf{e_{gd}}, \mathrm{A})$ is defined as

$$d(\mathbf{e_{gd}}, \mathrm{A}) = \begin{cases} 0 & \text{if } \mathbf{e_{gd}} \text{ is included in A,} \\ \min\limits_{\mathbf{p}\in \mathrm{A}} \| \mathbf{e_{gd}} - \mathbf{p} \|_2 & \text{otherwise,} \end{cases} \tag{6.11}$$

where $\| . \|_2$ is the Euclidean distance. This measure allows one to control the compromise between the guarantee for the ground truth epipole belonging to the set of focal elements in the considered solution, and the imprecision related to the area of focal elements. BBAs with large focal elements (thus spatially imprecise) including $\mathbf{e_{gd}}$ exhibit low error values for λ close to 0, but higher error values when λ increases. Conversely, committed BBAs with small focal elements close to $\mathbf{e_{gd}}$ but not necessary including it are all the more badly scored that λ is low and better evaluated for λ having large positive values. For each considered algorithm, we compute the error $\epsilon(\lambda)$ for each pair of images and then derive the empirical cdf (cumulative density function) of $\epsilon(\lambda)$. Since, as previously stated, the λ value strongly impacts the performance and then the ordering of the evaluated algorithms, we plot performance versus λ. For this, the whole cdf is summarized through its Area Under the Curve (AUC) value. The higher this value, the more efficient an approach is.

Figure 6.6 shows, for the different algorithms we consider, the AUC versus λ. Specifically for the proposed method, we consider the solution with the smallest value of $\epsilon(\lambda)$ among the proposed k solutions. It corresponds to an optimistic assumption that an additional source (as explored in next section) will allow for "good" cluster selection. Nevertheless, we found interesting to evaluate, albeit in a preliminary way, the proposed approach on this public dataset that offers very various poses and scenes. The results underline that, as the value of k increases, the performance of the proposed BBA clustering improves as expected, and it outperforms other methods. In most cases, the desired estimation is within the 4 or 5 first-ranked clusters. The sub-figures (a) and (b) in Figure 6.6 allow for comparison between results achieved with $\theta = 0.9$ and $\theta = 0.5$. We notice that results are slightly better with $\theta = 0.5$ which is different from the conclusion derived from Table 6.1 and, here, underline that the consensus measure used by RANSAC is not optimal. We also note that results appear robust with respect to the fusion order introduced by different fusion strategies (*min* and *max* ordering discussed in Section 6.4.2).

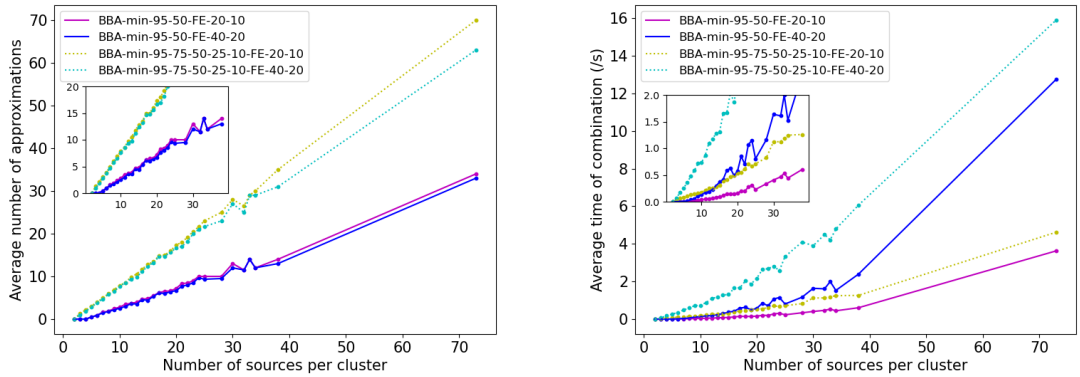
Finally, let us evaluate the robustness of the approach with respect to the FE number parameters, either n_{FE}^0 during the allocation, or n_{FE}^{max} and n_{FE}^{sum} during BBA approximation. Figure 6.7 evaluates the impact of the approximation parameters in two cases of initial BBA allocations: two nested FEs ($n_{\mathrm{FE}}^0 = 2$) corresponding to uncertainty levels 95% and 50% and five nested FEs ($n_{\mathrm{FE}}^0 = 5$) corresponding to uncertainty levels 95%, 75%, 50%, 25% and 10%. The curves corresponding to different approximation parametrizations are represented with different line styles, so that we can check the very low impact of these parameters on AUC. Comparing the two subfigures, we also note the low impact of the number of initial FEs, even if more pronounced for $k = 1$. Figure 6.8 shows, versus the number of sources per cluster (i.e., the number of BBAs to combine), the average number of approximation and the average computation time in seconds, still distinguishing the two BBA allocation cases $n_{\mathrm{FE}}^0 \in \{2, 5\}$. We notice that the approximation number curve mainly depends on n_{FE}^0 and, as expected, on the ratio $\frac{n_{\mathrm{FE}}^{max}}{n_{\mathrm{FE}}^{sum}}$ rather than on the absolute values of approximation parameters. Meanwhile, the average running time for cluster-BBA computation depends mainly on approximation parameters, in particular n_{FE}^{sum} which controls the complexity of the combination rule. Finally, noticing that the gain in AUC is either negligible when increasing $(n_{\mathrm{FE}}^{max}, n_{\mathrm{FE}}^{sum})$ or very low when increasing the initial number of FEs per BBAs, while the running time increases in a significant way, leads us to set default parameters to $n_{\mathrm{FE}}^0 = 2$, $n_{\mathrm{FE}}^{max} = 20$, and $n_{\mathrm{FE}}^{sum} = 10$.



(a) BBA allocation with $n_{\text{FE}}^0 = 2$

(b) BBA allocation with $n_{\text{FE}}^0 = 5$

Figure 6.7: Impact of the approximation parameters on the AUC curves: different colors correspond to different numbers of kept clusters (from $k = 1$ in black to $k = 6$ in magenta), plain and dashed lines correspond to $(n_{\text{FE}}^{max}, n_{\text{FE}}^{sum})$ equal to $(20, 10)$ and $(40, 20)$, respectively.



(a) Average number of BBA approximations

(b) Running time (in second) of cluster-BBA computation

Figure 6.8: Impact of BBA parameters n_{FE}^0, n_{FE}^{max}, and n_{FE}^{sum} ($n_{\text{FE}}^0 \in \{2, 5\}$ is called '95-50' or '95-75-50-25-10', respectively; $(n_{\text{FE}}^{max}, n_{\text{FE}}^{sum}) \in \{(20, 10), (40, 20)\}$ is called 'FE-20-10' or 'FE-40-20', respectively); subplots inside each subfigure are a zoom on $[0, 40]$ x-values.

6.5 Multi-source camera localization

With the proposed BBA clustering method, we have grouped a set of BBAs to different groups and selected some of them as the candidates regions based on a criterion with the k maximal pignistic probability (BetP) related to the focal elements with maximal intersections. The output of the belief clustering is the *proposed solution set* containing at most k BBAs representing the k most likely locations for the second camera. It enhances the consistency between BBAs inside the same clusters and narrows down the search region of epipole. However, it still remains some ambiguities among the final selected k clusters.

In order to reduce such ambiguities, we can use the additional sources to guide furthermore the selection and increase the reliability. For example, when the epipole corresponds to the camera wearer, the search range for epipole can be limited to the area occupied by pedestrians presented in the view of static camera. The pedestrian detections can then be regarded as the additional source for the localization problem. In addition, despite of the limit precision, the sensor data from GNSS can serve as additional source to help handling the ambiguity between the target and other pedestrians presents in the scene at the same time and be easily integrated to the belief function framework with other sources for epipole localization in the view of static camera.

In the following, we investigate two different examples of such sources whose availability depends on the considered system. In both cases, we will define BBAs modeling the new evidence brought by each of these additional sources. These BBAs are defined on the same discernment frame as previously, namely Ω, the image plane of the static camera.

6.5.1 BBA from a pedestrian detector

When considering a wearable camera, the device is necessarily co-localized with the person carrying it. Thus, a computer vision pedestrian detector appears as a relevant source for localization in the view of the static camera. Among the many algorithms proposed to detect and localize pedestrians, Convolutional Neural Networks (CNN) have proven to be highly effective, e.g. [88, 149, 181]. Their output is a set of Bounding Boxes (BB) around each detected pedestrian. Now, in the perspective of BBA definition, we underline three main BB features: (i) as the camera is often held near the head or the shoulder, it is more likely located in the BB upper part than in the lower part; (ii) as a BB can imperfectly enclose the pedestrian (in particular in case of occlusion), the mobile camera may be outside BB although very close; (iii) in case of multiple pedestrians detected it is impossible to assume which one is more likely to wear the camera.

Based on the these BB features, we consider a set of BBs denoted by $\mathscr{B} = \{\mathrm{B}_1, \ldots, \mathrm{B}_i, \ldots, \mathrm{B}_m\}$, followed with the dilation operation with disk structuring element of radius ρ denoted by δ_ρ, and the upper half of any box B_i denoted by B_i^{up}. We define a BBA associated to $\mathscr{B}$ with four focal elements: $\mathrm{B} = \bigcup_{\mathrm{B}_i \in \mathscr{B}} \mathrm{B}_i$, $\mathrm{B}^{\delta} = \bigcup_{\mathrm{B}_i \in \mathscr{B}} \delta_\rho(\mathrm{B}_i)$, $\mathrm{B}^{up} = \bigcup_{\mathrm{B}_i \in \mathscr{B}} \mathrm{B}_i^{up}$ and $\mathrm{B}^{\delta,up} = \bigcup_{\mathrm{B}_i \in \mathscr{B}} \delta_\rho(\mathrm{B}_i)^{up}$, illustrated in Figure 6.9. Among these four focal elements, the mass is about equally distributed even if we may refine the precise value during experiments. Note that as previously the BBA is dogmatic, but for a different reason. Here, we assume the pedestrian detector reliable enough to not miss the camera carrier. More specifically, we assume its ambiguities (between the different pedestrians present in the scene) are complementary to the ambiguities among BBA clusters. Then, we choose the BBA to be dogmatic in order to have a chance to detect, based on the degree of conflict, when it provides a completely erroneous solution.

6.5.2 BBA from GNSS data

In this setting, we assume that GNSS data are provided by a low-cost and light sensor embedded by the camera carrier. Thus the measured localization is rather imprecise and has to be combined with our other pieces of evidence for localization in order to increase spatial precision. The conversion of the rough 3D GNSS location into one or several focal elements in the image space requires the knowledge of the static camera pose parameters along with the theoretical 3D preci-

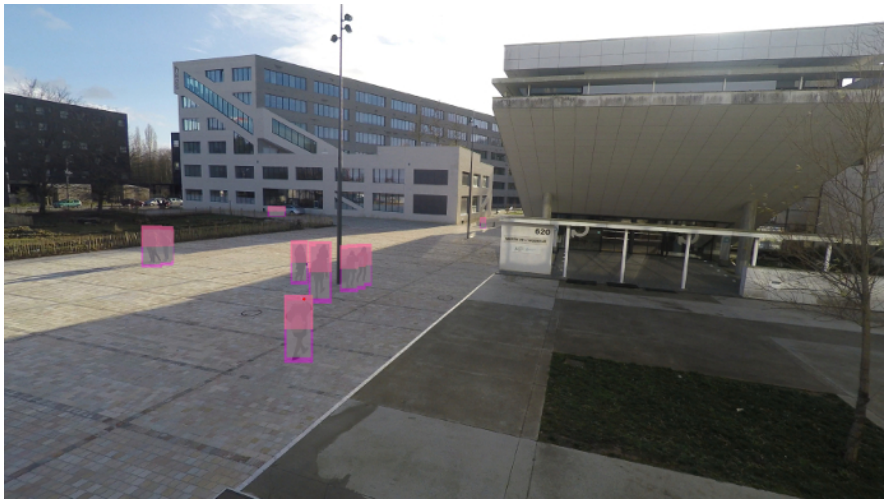
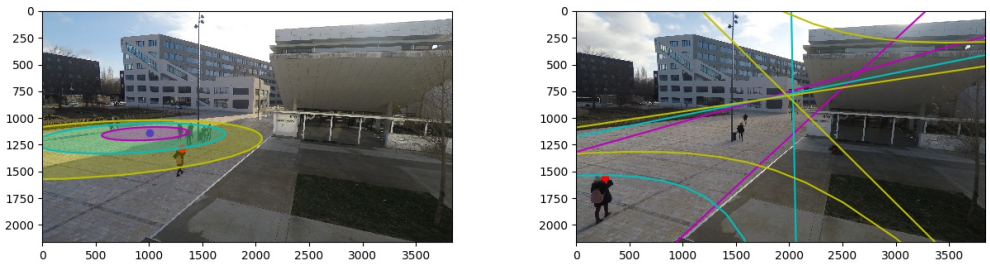


Figure 6.9: Focal elements of the BBA associated to the bounding boxes provided by the pedestrian detector.



(a) **(b)**

Figure 6.10: Illustration of the focal elements derived from GNSS localization. The three conics represent the projected uncertainty areas on image plane associated with GNSS noise level equal to σ, 2σ and 3σ, respectively. (a) presents the case of the projected conics being ellipses and (b) presents the case of the projected conics being hyperboles due to due to the presence of GNSS uncertainty areas behind the reference camera.

sion of GNSS. Besides, since the altitude (z coordinate) provided by the GNSS is far more imprecise than the horizontal plane location [172], we rather consider a prior (even if approximate) on the height h of the mobile camera (due to the fact it is held by a pedestrian).

Let P_h denote the plane parallel to plane $z = 0$ and having elevation equal to h and let (x, y) be the 2D GNSS coordinates representing its location on the Earth surface. Considering the GNSS intrinsic imprecision, we represent imprecise location areas as 2D nested disks in P_h centered on (x, y). Now, when the static camera elevation is large with respect to the interval (length) of the possible heights for mobile camera, the imprecision assuming a given height h is small with respect to GNSS intrinsic imprecision. Thus, in the following, we ignore the vertical uncertainty (along h) with respect to the horizontal one.

Given the position of GNSS (x, y) with a certain noisy level σ, there are two possibilities to derive the focal elements of the BBA associated to the GNSS data. One direct way is to first project (x, y) to the pixel coordinate (u, v) on image plane and follow the standard propagation pipeline to obtain the covariance matrix associated with (u, v). Then the focal elements of BBA would be nested ellipses centered in (u, v). However, this approach can only provide the approximate estimation for the uncertainty area with the first order propagation. It does not consider the fact that the center of ellipse on image plane may not be the projection of the center of circle on the plane

P_h after the projective transformation neither that the perspective projection of circle may not be ellipse but a hyperbole.

Thus, we adopt an alternative way which considers the general case for the transformation of conic under the perspective projection from P_h to the image plane. We simply project the 2D nested disks in the plane P_h to the image plane of the static camera, which can be performed via a homography transformation for circles. Under the pin-hole camera model and with the hypothes is of fixed height h for the ego-camera, the transformation from the plane 3D P_h with height h to the image plane can be represented by a homography matrix H described in the following equation:

$$\lambda\mathbf{p} = \lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathrm{KR}(\mathbf{q}-\mathbf{c}) = \mathrm{KR} \begin{bmatrix} x-c_x \\ y-c_y \\ h-c_z \end{bmatrix} = \mathrm{KR} \begin{bmatrix} 1 & 0 & -c_x \\ 0 & 1 & -c_y \\ 0 & 0 & h-c_z \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathrm{H}\mathbf{x},$$

where K is the intrinsic matrix of camera, (c_x, c_y, c_z) is the 3D coordinates of camera center, and R the rotation matrix for camera's orientation and the homography matrix

$$\mathrm{H} = \mathrm{KR} \begin{bmatrix} 1 & 0 & -c_x \\ 0 & 1 & -c_y \\ 0 & 0 & h-c_z \end{bmatrix}. \tag{6.12}$$

For a circle centered in $\mathbf{x} = (x, y, 1)$ with the radius r on P_h, we have $\mathbf{x}^{\mathrm{T}}\mathrm{C}_r\mathbf{x} = 0$, where C_r is the conic for circle with the following expression:

$$\mathrm{C}_r = \begin{bmatrix} 1 & 0 & -x \\ 0 & 1 & -y \\ -x & -y & x^2+y^2-r^2 \end{bmatrix}.$$

We then perform the projection of these circles on the image plane by conic transformation. Let C_e denote the transformed conic on the image plane. For the pixel $\mathbf{p}$, we have $\mathbf{p}^{\mathrm{T}}\mathrm{C}_e\mathbf{p} = 0$. By replacing $\mathbf{x}$ by $\lambda\mathrm{H}^{-1}\mathbf{p}$, we have $\mathbf{p}^{\mathrm{T}}\mathrm{H}^{-\mathrm{T}}\mathrm{C}_r\mathrm{H}^{-1}\mathbf{p} = 0$. So the transformed conic C_e in the image plane becomes

$$\mathrm{C}_e = \mathrm{H}^{-\mathrm{T}}\mathrm{C}_r\mathrm{H}^{-1}. \tag{6.13}$$

In most cases, C_e is an ellipse as illustrated in Figure 6.10a, under rares circumstances, the projective projection can result in a hyperbole as shown in Figure 6.10b. In the application, the BBA representing the GNSS localisation information has three nested focal elements that are the inside areas of conics obtained by perspective projection of three P_m circles centered on (x, y), having radius equal to σ, 2σ and 3σ, respectively, with σ the theoretical uncertainty or error bar on (x, y). Using 2CoBel, these focal elements are approximated by 2D polygons, denoted by A_σ, $\mathrm{A}_{2\sigma}$ and $\mathrm{A}_{3\sigma}$. As previously, the mass is about equally distributed between these three focal elements even if we may fine tune the precise values during experiments and the BBA is dogmatic, which means we assume it reliable enough to not miss the epipole, and otherwise the fusion process will be invalidated thanks to conflict detection.

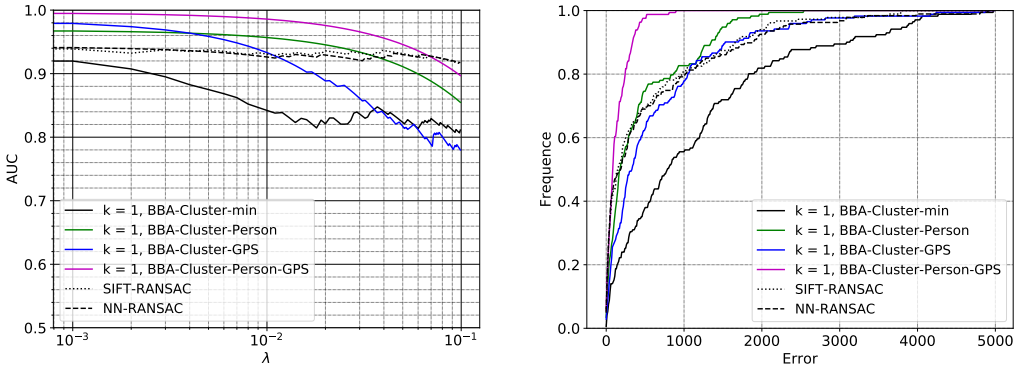
6.5.3 Global fusion algorithm

From Section 6.4, we derive a set of l cluster BBAs $\tilde{\mathscr{M}} = \{\tilde{m}_i\}_{i\in\{1,\dots,l\}}$ and from Section 6.5 another set of BBAs brought by additional sources $\mathscr{M}^a = \{m^a{}_j\}_{j\in\{1,\dots,q\}}$. Now, if all $\mathscr{M}^a$ BBAs can be assumed reliable (i.e., there is at least one FE containing $\mathbf{e}$), we know that some BBAs of $\tilde{\mathscr{M}}$ correspond to outliers. Furthermore, by construction of the clusters, two different cluster BBAs are highly

Algorithm 3 BBA selection based on additional sources

Input: The set of cluster BBAs $\tilde{\mathscr{M}} = \{\tilde{m}_i\}_{i\in\{1,\dots,l\}}$; The set of additional BBAs $\mathscr{M}^a = \left\{m_j^{add}\right\}_{j\in\{1,\dots,q\}}$;
Output: BBA m_{out}

$m^a_{\textcircled{\cap}} = \textcircled{\cap}_{m_i^a\in\mathscr{M}^a} m_i^a$;
$\bigcup m^a_{\textcircled{\cap}} = \bigcup_{\mathrm{B}\subseteq\Omega, m^a_{\textcircled{\cap}}(\mathrm{B})>0} \mathrm{B}$;
$\mathscr{M}_c = \{\}$
for $i = 1$ to l **do**
  if $\bigcup m^a_{\textcircled{\cap}} \cap \bigcup \tilde{m}_i \neq \emptyset$ **then**
    $m^{i,a} = \tilde{m}_i \textcircled{\cap} m^a_{\textcircled{\cap}}$
    Add $m^{i,a}$ to $\mathscr{M}_c$;
  end if
end for
$m_{out} = \underset{m^{i,a}\in\mathscr{M}_c}{\arg\max}\{max_{\mathrm{A}\subset\Omega} \mathrm{Pl}^{i,a}(\mathrm{A})\}$



(a) AUC versus λ

(b) Cumulative density function for λ = 0.01

Figure 6.11: Results in terms of AUC and CDF of the error ϵ(λ) achieved by the multi-source localization of mobile camera.

incompatible so that, for conjunctive fusion process, we can only retain one among the l cluster BBAs. Our fusion strategy is then to use the $\mathscr{M}^a$ BBAs to adjudicate.

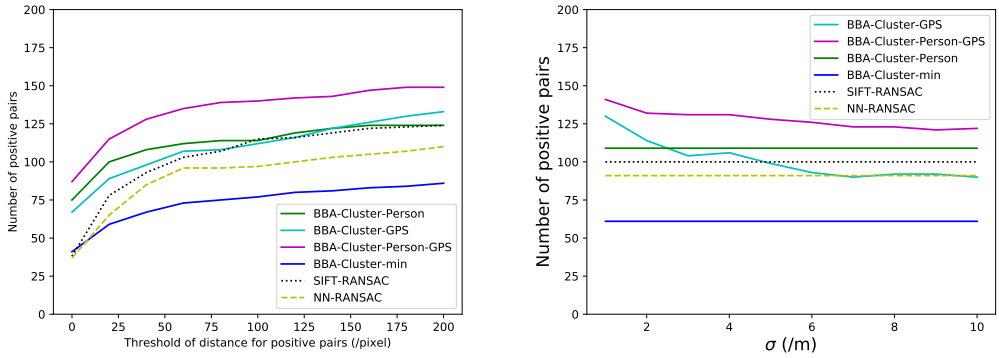
To save some computational resources we use two tricks in Algorithm 3. Firstly, benefiting from the associativity property of conjunctive rule (see Definition 5.1.15), $\mathscr{M}^a$ BBAs are combined only once before $\tilde{\mathscr{M}}$ BBAs inspection: $m^a_{\textcircled{\cap}} = \textcircled{\cap}_{m_i^a\in\mathscr{M}^a} m_i^a$, along with the disjunction of $m^a_{\textcircled{\cap}}$ focal elements, denoted by $\bigcup m^a_{\textcircled{\cap}}$. More generally, for a BBA m, the disjunction of all its focal element is denote $\bigcup m$. Secondly, $\tilde{\mathscr{M}}$ BBAs are filtered by testing the intersection between $\bigcup m^a_{\textcircled{\cap}}$ and the focal elements of $\tilde{m}_i \in \tilde{\mathscr{M}}$: if the intersection is empty, $\tilde{m}_i$ is withdrawn.

Then, having evaluated to consistency of each cluster-BBA with the additional source BBA through their conjunctive combination, the output BBA is the one that will maximize the decision (epipole location) criterion (even if the algorithm itself does not provide such a crisp decision).

6.5.4 Experiments and results using additional sources

We evaluate the combination of the proposed clustering method with the additional sources on 196 synchronized image pairs extracted from the dataset *Building entrance* (see Section 2.1.2). In this dataset, the epipole corresponds to the camera wearer moving on the ground level so that the pedestrian detection can be used as the additional source for epipole localization.

For BBA clustering, accordingly to previous section, we set θ = 0.5 and use *min* ordering for intracluster BBA combination. For the pedestrian detector, we focus on the widely used object detector **YOLO** [131]. As the GNSS source is not available in this dataset, we generate the sim-



(a) Number of positive pairs versus distance threshold; GNSS noise $\sigma = 3m$.

(b) Number of positive pairs versus GNSS noise level; distance thresold = 50px.

Figure 6.12: Number of positive pairs for different sources used for mobile camera localization.

ulation of GNSS position by adding a random normally distributed noise to the ground truth of epipole location. For each pair of images, we randomly sample a realization from the distribution $\mathcal{N}(\mathbf{e_{gd}}, \Sigma_{\mathbf{e_{gd}}})$ with $\Sigma_{\mathbf{e_{gd}}} = \sigma^2 \times \mathbf{I_{2\times 2}}$, where σ is the defined noise level.

Since we investigate the benefit of multi-source fusion for our application of localization, the presented results contain a gradually increasing number of sources, from only considering **SIFT-RANSAC** or **NN-RANSAC** results or BBA clustering results with $k = 1$((**BBA-cluster-min**)), to also considering either one additional source (the pedestrian detector (**BBA-cluster-min-Person**) or the GNSS simulation (**BBA-cluster-min-GPS**)) or the two additional sources (**BBA-cluster-min-Person-GPS**).

Figure 6.11a shows the AUC curves versus the λ parameter whereas Figure 6.11b shows the cdf for $\lambda = 0.01$ (knowing that the first term of $\epsilon(\lambda)$ ranges in $[0, 5.10^3]$ and the second term in $[0, 10^5]$). We notice the very high performance achieved considering the three sources (BBA clustering on RANSAC solutions, pedestrian detector and GNSS data). Specifically, we note that the single use of BBA clustering method with $k = 1$ is less competitive than the standard RANSAC approaches. This is mainly due to the fact that the standard approaches provide a larger ellipse than the BBA clustering, that nevertheless was defined to provide several rather committed solutions for further fusion purposes. Note also that, if on the previous dataset NN-RANSAC provided better results than SIFT-RANSAC, its results on this dataset are rather disappointing. Now, combining the output of BBA clustering with one additional source (either pedestrian detector or GNSS data) allows us to overcome the limitations of these traditional approaches, and using all sources results in a rather significant leap in performance. Such results both highlight the effectiveness of the filtering of the set of initial solutions based on BBA clustering and the ability of an additional source to select the correct BBA cluster.

We also provide an alternative evaluation of our results by looking at the evolution of the number of positive image pairs with respect to the distance threshold to the ground truth epipole. The distance on pixel between the BBA provided by each method and the ground truth epipole for each pair of images, is defined as

$$d(e_{gd}, m) = \begin{cases} 0 & \text{if } \exists A \in \mathcal{F}(m),\ e_{gd} \text{ is included in A;} \\ \min\limits_{x \in \cup \mathcal{F}(m)} \| e_{gd} - x \| & \text{otherwise.} \end{cases} \tag{6.14}$$

Note that previous equation is completely consistent with Equation 6.11 since it boils down using it with the categorical BBA having as unique focal element the disjunction of all focal elements of BBA m.

To count the positive pairs including the ground truth, we relax the requirement of "including" to "close", as we are more interested in the position of mobile camera wearer instead of the

exact position of mobile camera center. A pair of image is considered as positive for an estimation method if the distance $d(e_{gd}, m)$ is smaller than a defined threshold.

Figure 6.12a provides the number of such positive pairs versus the used distance threshold. Note that, in our application, we consider it is still meaningful to localize the target camera wearer even when the distance threshold to the ground truth epipole is increased to 100 pixels compared to the large resolution of the reference image (4K). Figure 6.12a confirms the conclusions previously drawn from Figure 6.11a.

Finally, for the distance threshold equal to 50 pixels, we look at the number of positive pairs versus GNSS noise. We note that, as expected, as the GNSS noise increases, the number of positive pairs obtained by fusion involving GNSS data decreases. Nevertheless, when used in addition to the BBA clustering output and pedestrian detector, the GNSS data appear useful even with rather high noise levels (up to 10 m) since performance overcomes the results obtained without it.

6.6 Multi-temporal epipole localization

In this application, we consider no longer a mobile camera within the field of view of a static camera, but a pair of static cameras. Assuming that these cameras capture synchronized video streams of a dynamic scene, we aim to exploit the temporal sequence for epipole localization. As the cameras are fixed and the scene is dynamic, each pair of frames can provide a new estimation for the fixed epipole location using a standard RANSAC process applied to image pairs. Note that, depending on the speed of the moving objects with respect to the frame acquisition frequency, we may have to sub-sample the sequence in order to ensure sufficient changes in the video content and cognitive independence of epipole estimations. In our case, the sequence frequency is 2 frames per second and we observe that, due to the fact that most matches occur on moving objects, the ratio of consecutive point matches (between each image pair) is about 12%, so that further sequence sub-sampling is not necessary.

Then, each instantaneous epipole solution is associated to a consonant BBA having $n^0_{\mathrm{FE}} = 2$ nested focal elements corresponding to ellipses at 50% and 90% confidence levels using uncertainty propagation. This BBA construction is similar to the one used in Section 6.4, except the derivation of the epipole solution performed here using standard RANSAC. Then, if the whole sequence contains N frames, we derive a set of N BBAs. Among them, many may be irrelevant (*outlier* BBAs) due to erroneous matches between keypoints.

Now, the basic idea of fusion is still to remove outliers based on cross-validation performed between multiple sources. Having a single temporal sequence, we create multiple sources by splitting it into T non-overlapping subparts. To each subpart $t \in \{1, \ldots, \mathrm{T}\}$ we associate a set $\mathscr{M}_t$ of $\lfloor \frac{\mathrm{N}}{\mathrm{T}} \rfloor$ BBAs. $\mathscr{M}_t$ is then processed according to the proposed belief clustering to derive *proposed solution sets* containing the k top-ranked cluster BBAs: $\tilde{\mathscr{M}}_t = \{\tilde{m}_{t,1}, \ldots, \tilde{m}_{t,k}\}$. Theoretically, we have to evaluate every T-tuples of BBAs formed by selecting one BBA in each set $\tilde{\mathscr{M}}_t$. However, as described in Algorithm 4, to save computational resources, we perform processing sequentially by detecting the BBAs with conflict degree equal to 1 as early as possible to avoid their combination. Then, like in Algorithm 3, we also base ranking of the combined BBAs on decision criterion. Now, the main difference is that we do not select the top BBA but the v first-ranked ones, in an ad-hoc and conservative spirit. These v first-ranked BBAs are then gathered in a single BBA using disjunctive combination since they are incompatible by construction (selection of incompatible clusters in at least one subpart of the sequence).

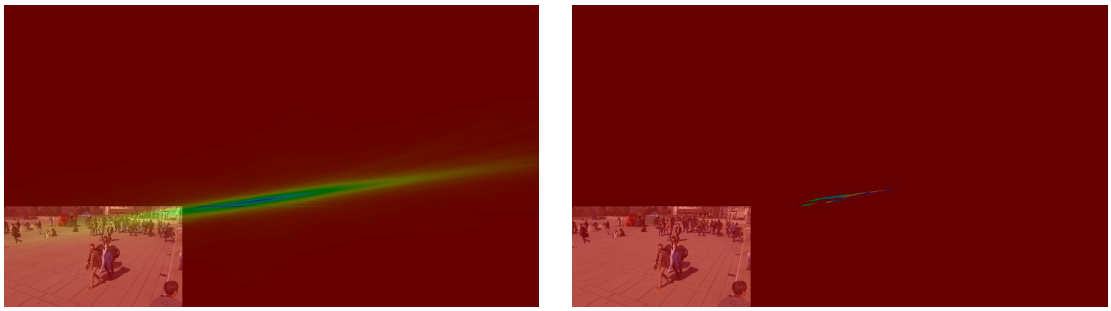
6.6.1 Experiments and results using temporal sequence

We evaluate the epipole localization using a temporal sequence captured by a pair of static cameras from the dataset 2.1.4. The localization algorithm is the one described in Section 6.6, Algorithm 4 with T = 4 and v = 10. Since the whole sequence contains 400 frames, each of the 4 image subsets contains 100 consecutive frames, from which BBA clustering allows us to retain the

Algorithm 4 Epipole estimation from video sequence

Input: The T sets of cluster BBAs $\tilde{\mathscr{M}}_t$, $t \in \{1,\ldots,\mathrm{T}\}$;
Output: BBA m^{out};

```
M_cur = M̃_1;
M_new = ∅;
for i = 2 to T do
  for all m ∈ M_cur do
    for all m̃_{i,j} ∈ M̃_i such that ⋃m ∩ ⋃m̃_{i,j} ≠ ∅ do
      Add m ⓝ m̃_{i,j} to M_new;
    end for
  end for
  M_cur = M_new;
end for
Rank (according to chosen criterion, BetP or Pl) BBAs in M_new and store the first-ranked v BBAs
in M_out
m^out = ⓤ_{m_s ∈ M_out} m_s;
```



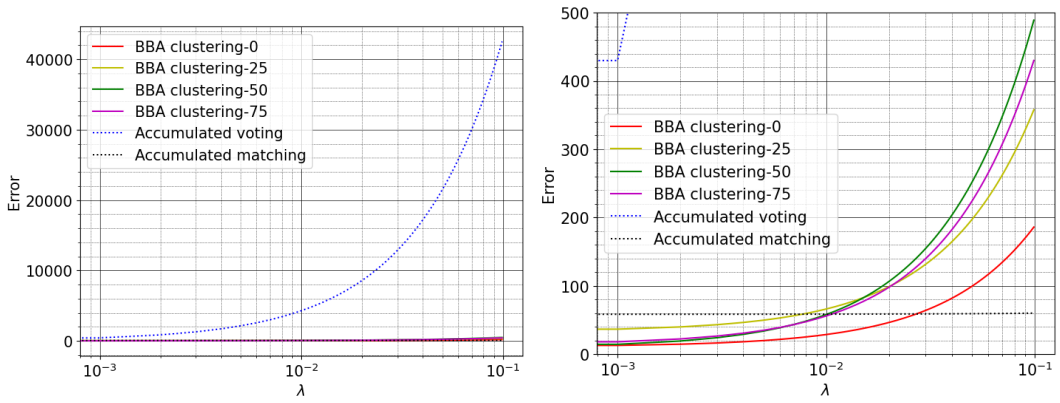
(a) Accumulated Vote **(b)** BBA clustering

Figure 6.13: Qualitative comparison between accumulated voting and the proposed multi-temporal BBA fusion.

5 first-ranked cluster BBAs ($\forall t \in \{1,\ldots,4\}$, each $\tilde{\mathscr{M}}_t$ contains 5 BBAs). For comparison, we also consider the accumulated voting proposed in Chapter 4 which has proved to be more conservative/cautious than RANSAC in case of difficult settings.

Figure 6.13 illustrates qualitatively the results of the accumulated voting and of the proposed Algorithm 4. It clearly appears that our result is much more precise than the voting strategy proposed earlier [26]. Now, SIFT-RANSAC considering the whole sequence of 400 images provides a very confident result (ellipse with axes of a few pixels; not shown). Even if RANSAC result may appear rather good since it is actually close to the actual epipole location (58 pixels), it completely fails in estimating the actual uncertainty of its solution. Indeed, with so many data (keypoint matches accumulated through the temporal sequence), RANSAC algorithm (whatever the considered variant) will be overconfident in the obtained result missing its actual reliability. The proposed method appears then as a good compromise between perhaps too cautious results obtained using accumulated voting and too committed ones obtained using RANSAC sampling.

Quantitative evaluation is presented on Figure 6.14 which shows equation 6.10 with respect to λ values. In contrast to the previous experiments, in this setting we have only one result sample (obtained considering the whole sequence) so that we cannot plot error statistics (pdf, AUC). In order to check the sensitivity of our approach to the considered subsets in the sequence, we introduce different offsets (called as '-0', '-25', '-50' and '-75') in the sequence split. From Figure 6.14 left, we see that, due to the very large size of the obtained uncertainties, the accumulated voting completely fails in providing an interesting result. Then, zooming on low error values (Figure 6.14 right), we notice that, for low λ values, our approach slightly outperforms RANSAC result. Indeed



(a) Whole error dynamic range **(b)** Zoom with bounded error dynamic range $[0,500]$

Figure 6.14: $\epsilon(\lambda)$ (Equation (equation (6.10))) versus λ; the four curves 'BBA-clustering-XX' refer to different offset values in the proposed approach, 'accumulated voting' refers to the method proposed in [26], and 'Accumulated matching' to SIFT-RANSAC considering the keypoint matches accumulated during the entire sequence.

the distance term in equation 6.10 is about $10px$ against about $60px$. However, since uncertainty is much higher, once $\lambda > 10^{-2}$ the RANSAC solution error becomes lower. Nevertheless, as stated previously, the evidential approach result seems a good compromise between result precision and actual uncertainty. Finally, let us underline that the subset offset does not seems to impact results. More specifically, the epipole location is influenced very marginally but the focal element size does vary a little bit more (which explains the fact that error curves separate when λ increases).

6.7 Conclusion

In this chapter, we deal with the epipole localization within the 2D Belief function framework. We propose a fusion strategy based on the belief clustering to overcom the challenges raised by a large number of sources, including a significant ratio of outliers, need to be combined. The adopted approach for mitigating the impact of the presence of outliers is to perform a preliminary clustering process, which organizes the sources in coherent groups. This step allows for intra-cluster fusion to be performed without increasing the mass on the empty set or requiring the user to dispatch it. The resulting BBA across the source clusters may be used afterwards for fusion with additional sources of information. Our strategy exploits the fact that our algorithm is less committed than the standard vision-based solutions and thus more favorable to the use of additional sources. The closest applications to our work are related to pedestrian or vehicular transportation, but the underlying strategy of intra-cluster fusion may be helpful in a wider range of problems which benefit from large amounts of conflicting data sources.

Up to this point, we have address the agent localization based on epipole estimation. In the following part, we extend the target of localization from the camera wearer to other pedestrians.

Part IV

Pedestrian Localization

Chapter 7

Use of scene geometry for across-view data association

Contents

7.1 Introduction

7.1.1 Motivation

The pedestrian localization is a paramount task in computer vision, which enables us to identify, analyze and predict human interactions and activities in different contexts. It is a crucial part for various applications such as driving guidance systems, emergency intervention and the military search as well as the intelligent surveillance system in our case. Different pedestrian localization systems have been conceived, including GPS-based, vision based or radio based [5] systems,

Among them, the vision-based systems provide the most feasible and least costly solution. Relying on detection and localizing pedestrians in 3D, the vision-based systems can be divided into monocular systems based on a single camera, stereo systems based on two cameras, or multiple camera networks. When it comes to our case for the geo-localization of dangerous perpetrators in the body-worn camera equipped by the law enforcement agent (LEA), the process involves a collaborative system between the egocentric camera moving on the ground level with the support of security camera footage fixed at the top level at a certain height from the ground.

Given the detections of pedestrians from the views of the mobile camera on the ground level and the static camera at the top level, the geo-localization of the target pedestrians consists in localizing the target pedestrian in the 3D local world coordinate frame and then transforming it from the local world coordinate frame to the geo-reference frame in order to provide GPS coordinates in the real world. The pedestrian localization in the 3D local world coordinate frame can be performed in several ways depending on the knowledge of camera calibrations as well as the scene environment. If the pedestrian is detected by one camera, a projection ray in 3D connecting the camera center and a contact point on the foot of the pedestrian can then be derived by perspective constrains using the detected bounding box on image plane as well as the camera parameters. The pedestrian position in 3D can then be determined with the intersection of the corresponding projection ray and the plane on the ground which is assumed to be flat. An alternative way is to perform the triangulation based on the geometric constraints for two cameras. This method requires to search the same pedestrian in the views of both cameras, which can be referred to the problem of data association (DA).

Egocentric (first-view) visual data provides rich information for safety and security applications, ranging from public photos in urban settings to wearable cameras used by law enforcement agents (LEA). Compared to the single use of egocentric view, the joint use with an overview camera has the potential to support a more accurate pedestrian localization, as well as a finer analysis of interactions occurring among the visible participants. However, it also raises more challenges for across-view data association due to strong visual appearance variations and scale changes, which make the egocentric DA is more difficult than standard across-view DA for relating directly visual information between the views.

An alternative way for across-view data association is to rely on the geometric priors provided by the relative pose estimation between the cameras may assist the association significantly by restricting the matching candidates to a small subset of detections in the other view. Although it requires a spatial calibration of cameras, the 3D geometric priors relating two views help overcome visual ambiguity due to the change of views for the same person. However, the geometry constraint are not meant to solve entirely the DA, but rather to complement an appearance based constraint, as the potential match subset may still contain multiple detections especially in crowded areas, depending on the alignment of the pedestrians with respect to the cameras. In addition, the bounding box associated with the detection of pedestrians may be not fitted perfectly the real person which make geometric matching not accurate due to the error introduced in the process of triangulation. Furthermore, the relative pose estimation is built up on identifying invariant low-level visual cues, which suffers occasionally from the same limitations as the appearance source in previous paragraph. This implies that the relevance of the geometrical constraints depends on the spatial configuration of the cameras and of the dynamic objects in the scene.

To overcome the limits for the single use of appearance side or the geometric side, our work explores different approaches to use the geometric priors for DA in the context of overlapping fields of view between two cameras and studies how the geometric consistency can help improve the standard appearance cues. The contribution of this study is twofold. First, we derive a reliable multi-views DA by estimating spatial relationships among the observed pedestrians with the geometric constraints. Second, we develop different strategies for the joint use of geometric priors and appearance cues using both an explicit combination rule and a learned strategy, depending on the availability and reliability of annotated training data.

7.1.2 Related works

The data association problem consists in matching the detections of same object, which is a fundamental step in the multi-object tracking system. A large body of literature is devoted to data association (see for example [13] for a comprehensive survey), since this term encompasses multiple procedures. Depending on the context, they can be divided into different categories. For a single-camera system, the data association is performed with a temporal sequence of observations [12, 79, 85, 117], where handling the occlusions are very challenging along with false detections. When it comes to the multiple cameras networks, the data association can be either coped with the batches methods by solving the global optimization problem in a graph-based representation for the sequential observations [86, 122, 162, 175], or with the online methods using Hungarian algorithm [80] for the observations across the instantaneous multiple views, referred as across-view data association. Considering the scenario occurring in realistic situations for our project related to surveillance task, we restrict our work to deal with the across-view data association for pedestrians detections in the overlapping fields of views from two calibrated cameras. Note that this problem is different from the person re-identification problem in non-overlapping configurations.

The similarity or dissimilarity measure between objects is an important and challenging part for any data association algorithm. Most of the works are related to appearance-based criteria which rely on the similarity or distance between the feature vectors extracted from the image patches. Various feature descriptors have been proposed for pedestrian detections, including the hand-crafted features based on the color histogram [17], the Histograms of Oriented Gradients (HOG) [34] or the SIFT-like features, and some learned representations using the deep neural networks. These descriptors can be used individually or in an ensemble manner [9, 66, 152] for the similarity measure. Despite the improvement of feature descriptors, the reliability of the appearance-based criteria is still limited by the visual ambiguities due to strong visual appearance variations.

There are also some works relying on the available geometric constraints for object matching. When the object moves on a plane, the homography-based constrains [15, 46, 78] are used to build the homography mapping between the 2D object observations on the ground from different views. The similarity or dissimilarity measure can then be derived from transferred error associated to the considered pair of objects under the mapping of homography matrix. A more general constraint may be based on the epipolar geometry which limits the search region of an object to the region around the corresponding epipolar line. Compared to the point-to-point mapping with the homography matrix, the epipolar constraint provides the point-to-line correspondence which may lead to ambiguities for the objects lying on the same line. Besides the 2D constraints based on the homography or the epipolar geometry, it is also possible to employ the 3D geometric constraints. In [77], the authors proposed to match the objects according to the 3D position measurements provided by additional sensors which may be not available in the practical case for the urban surveillance task. Our work relies on a more feasible solution based on the 3D geometric consistency related to the 2D measurements for a same object using the multiple view geometry and the scene information. Despite being robust to the appearance change of objects, the geometric-based methods for data association may be influenced by the errors introduced in the 2D measurements as well as the camera calibration process. Instead of relying on a single appearance aspect or geometric aspect, our work explores furthermore the combination of them.

When it comes to combine the geometric consistency and the appearance based similarity for the data association problem, there are multiple avenues to perform it. In the single camera context, structural constraints defined among detections can cope not only with the relative motion of the targets, but also with the jitter introduced by the camera motion [168]. For multiple views, one may typically extend an across-time global optimization [23, 158], or cast the DA as a global energy minimization [120]. In all cases, the underlying idea is to combine into a single objective function two penalty terms, one derived from appearance and one from geometry, in addition to the usual association constraint (e.g., one-to-one assignment). However, the reliability of the two

terms usually suffers from different sources of errors. A fundamental challenge is then represented by how to combine two different terms in the presence of variable reliability and quality.

There exist a large number of combination rules devoted to combining two imprecise terms, ranging from the explicit rules defined as simple functions (linear or not) of real numbers such as sum, weighted sum, product or min/max, or more sophisticated rules supported by uncertain reasoning formalism, such as belief function theory [40] or possibility theory [43], to the data-driven based rule [85, 90, 163] which trains a single classifier for different similarity scores. By learning a fusion rule, the data-driven method based on the deep neural network provides the flexibility of dynamically exploiting multimodal and compact representations of the detections we intend to associate. This requires in turn a labelled training set of a size to adapt the complexity of the fusion model. Due to the simplicity and the power of approximation of feedforward neural networks, their use has been explored in some multimodal fusion tasks, typically involving visual and non-visual information [36, 51, 91, 107, 108, 156]. In our work, we follow these deep fusion fashion to explore the suitable combination for appearance features and geometric consistency in the data association problem.

7.2 Data association

We focus on the problem of across-view data association, which aims to build the correspondence between 2D detections of pedestrians from two synchronized views. Specifically, two cameras are positioned with the overlapping filed of views. We assumed that the cameras are calibrated and their poses with respect to the real world are also known. The detections of pedestrians are represented with a set of bounding boxes provided by the pre-trained pedestrian detectors. In the following, we first formulate the studied problem and then present the association algorithm. Finally, we discuss the performance and some potential ways to improve it.

7.2.1 Assignment problem and Hungarian algorithm

The across-view association between two frames can be derived from the classical assignment problem which is initially designed to find the optimal work assignments among workers given the one-to-one performance between the workers and the jobs. Assuming a set of works, denoted by $\mathscr{D} = \{d_1, d_2, ..., d_i, ..., d_N\}$ and a set of workers with the same cardinality denoted by $\mathscr{E} = \{e_1, e_2, ..., e_i, ..., e_N\}$, the assignment is performed by solving the optimization problem

$$\begin{aligned} &\min \sum_{i=1}^{N}\sum_{j=1}^{N} c_{ij}x_{ij}, \\ &\text{subject to } \sum_{i=1}^{N} x_{ij} = 1, \forall j \quad \text{and} \quad \sum_{j=1}^{N} x_{ij} = 1, \forall i, \end{aligned} \tag{7.1}$$

where c_{ij} represents the cost to assign the work d_i to the worker e_j, and x_{ij} is a binary variable to represent the assignment. If the work d_i is assigned to the worker e_j, then $x_{ij} = 1$, otherwise $x_{ij} = 0$.

The most popular algorithm to such problem is proposed by two Hungarian mathematicians Kuhn and Munkres so that it is also known as Hungarian algorithm [18, 73, 106]. This algorithm deals with the optimal assignment by minimizing the sum of cost related to assignment in polynomial time which is much more efficient compared to a complete combinatorial approach. Given a cost matrix, the basic idea of the Hungarian algorithm is to use some permutations provided by the row subtract and the column subtract with special values in order to obtain a set of independent zeros elements with the number of rows which guarantees that the sum of the elements in the original cost matrix in the position of these independent elements are minimal due to two propositions in [80]:

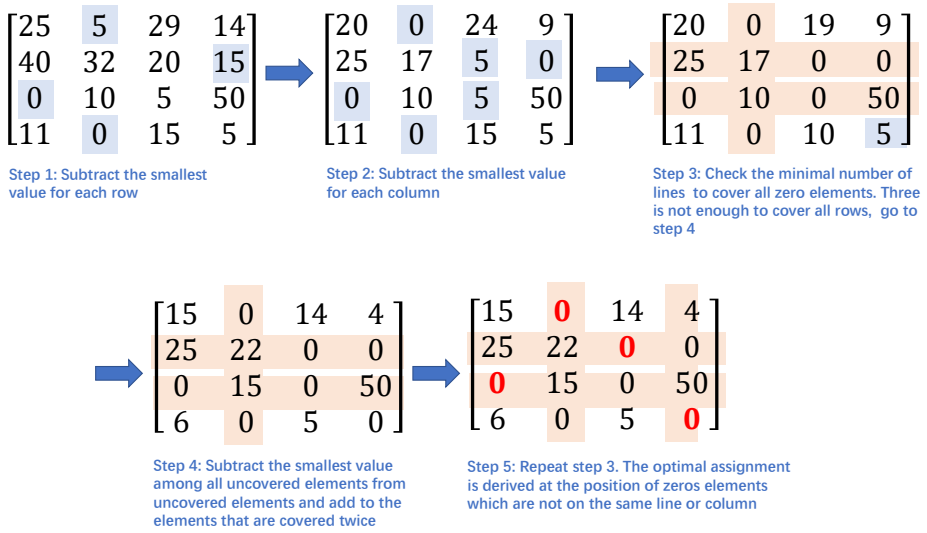


Figure 7.1: Illustration of Hungarian algorithm with an example.

Proposition 7.2.1. *If m is the maximum number of independent zero elements of a matrix* A*, then there are m lines which contain all the zero elements of A.*

Proposition 7.2.2. *The solution of assignment problem is not changed if the elements in the cost matrix are subtracted with arbitrary constants for each row and column.*

The full pipeline of the Hungarian algorithm is then performed as follows:

1. For each row of the given matrix, find the smallest value and subtract it from each element in the row. The new matrix is called the reduced matrix;
2. For each column of the reduced matrix, find the smallest value and subtract it from each element in the column;
3. Cover all zero elements with the minimal number of lines. If the number of lines equals the total number rows of cost matrix, the optimal assignment is then derived from the elements in the position of the zero elements which are neither on the same row nor the same column. Otherwise continue with the following steps;
4. Find the smallest value ϵ among all uncovered elements and subtract it from uncovered elements. Add ϵ to the elements in the position of the intersection of two covered lines.
5. Repeat Steps 3 and 4 until an optimal solution is achieved.

An example is also illustrated in Figure 7.1.

7.2.2 Association cost

To apply the Hungarian algorithm for the problem of data association, the core part of the Hungarian algorithm is to build the cost matrix. Given two sets of detections from two views, denoted by $\mathrm{P}_1 = \{p_i\}_{1 \leqslant i \leqslant \mathrm{M}}$ and $\mathrm{P}_2 = \{p_j\}_{1 \leqslant j \leqslant \mathrm{N}}$, the association cost c_{ij} can be derived from a dissimilarity measure between the i-th detection in the first view and j-th detection in the second view. As some detections may be present only in one view, the decision of non association should be possible. To quantify the benefit of a non-association with respect to a costly association, a cost of non-association is defined so that whenever an association cost is higher than it, the optimization process is allowed to reject it.

We note that the Hungarian algorithm presented in Section 7.2.1 is designed for the one-to-one balanced assignment with squared cost matrix. In our case, the number of detections in two views may be different and the association is not exactly 'one-to-one' due to the possibility of non-association. Consequently, to adapt the Hungarian algorithm, we extend the dimension of the association matrix to $(M+N)\times(M+N)$ and incorporate the cost of non-association as follows:

$$A_{ij} = \begin{cases} c_{ij} & \text{if } 1 \le i \le M,\ 1 \le j \le N, \\ \gamma & \text{otherwise,} \end{cases} \tag{7.2}$$

where γ is a parameter representing the non association cost, which is constant whatever the considered detection in this simplified model.

Although the Hungarian algorithm provides an optimal global solution for the data association problem, the accuracy of association relies mainly on the reliability of the cost c_{ij} in Equation (7.2). The basic approach that relies on single cues such as the 2D appearance features may be not robust due to multiple factors, such as the strong variations related to human pose and environmental illumination between multiple views. Specifically, the scale variations and the strong occlusions frequently occurring in the first-view camera increase the difficulty of the task for association between egocentric and an overview camera, which requires that DA be flexible in terms of not associating a potentially high number of targets which are present only in the overview camera. In the following, we first introduce the cost derived from the appearance cues. Different geometric priors are then explored and proposed to be integrated with the appearance cues into the cost to improve the association accuracy.

7.3 Appearance-based cost

The association costs based on appearance rely on the visual feature descriptors associated with the detections. The cost values are derived from the distances between feature descriptors. These descriptors can be distinguished between the traditional hand-crafted descriptors based on color, texture or shape information, and the learned ones extracted by a deep neural network. In this work, we consider a basic hand-crafted descriptor as the baseline representation, as well as an advanced feature extractor based on neural network in order to study how the geometric priors can improve the appearance features with different qualities.

7.3.1 Baseline representation

We consider the descriptor based on the color histogram as the baseline representation. This descriptor is built by concatenating the color channels into a 1D histogram and couple with the distance measure based on χ^2 Bin Ratio-Based Distance [67], which is robust to partial occlusion. Let $\mathbf{h^i} \in \mathbb{R}^n$ denote a L_2 normalized histogram with n bins used as a descriptor of the detections i, and $\mathbf{h^j} \in \mathbb{R}^n$ describing an object j. Then, the χ^2 Bin Ratio-Based Distance between $\mathbf{h^i}$ and $\mathbf{h^j}$ is defined by [67]

$$d_{hist} = d_{\chi^2} - \frac{1}{2}\left\|\mathbf{h^i} + \mathbf{h^j}\right\|_2^2 \sum_{k=1}^{n} \frac{(h_k^i - h_k^j)^2 h_k^i h_k^j}{(h_k^i + h_k^j)^3} \tag{7.3}$$

where $d_{\chi^2} = \frac{1}{2}\sum_{k=1}^{n} \frac{(h_k^i - h_k^j)^2}{h_k^j + h_k^j}$.

The main advantage of this hand-crafted representation is that it does not require any learning task nor annotated data for its use, although as we will show, some labelled observations may be useful for parameter tuning.

7.3.2 Learned representations

During the last decade, learned features have proved to be very efficient to perform tasks such as object recognition and image classification, as well as detection and association. In this work, we consider a deep convolutional neural network based pedestrian re-identification (re-ID) algorithm. Specifically, similarly to face recognition algorithms as [63], a general approach which is widely used for this task is to train a network using an ID loss [146] or a triplet loss [65] on pedestrian datasets. Then, the last layer is extracted and used as a compact, invariant pedestrian representation. For a given sample, this feature vector may be used for ranking available candidates by the wildly used cosine similarity in the range of $[-1, 1]$.

A re-ID algorithm will always benefit from fine-tuning with scene-specific data, due to the inevitable changes with respect to the training set which will create a slight domain shift. However, available pre-trained models provide already a strong baseline performance [96] for re-ID tasks, which is entirely adequate for our fusion objective.

7.4 Geometric costs

As applying a geometric cost requires that the pedestrian views be synchronized and provided by two cameras with overlapping fields of view, which is not the standard scenario in re-identification, the use of a geometric cost that is less popular for the task of data association. There are however two contexts in which geometry supports data association: a) *handshaking*, i.e., transferring a pedestrian at the shared border of the fields of view of two cameras in order to be able to keep tracking with the same ID, and b) *multi-view analysis* when pedestrians need to be identified simultaneously in different, overlapping views for various aims (tracking, localization, activity recognition). In this study, we consider two different ways to take into account the geometric priors for the data association problem, which are derived from the geometric distance and a basic geometric constraint.

7.4.1 Geometric distance

It is possible to discriminate between true and false elementary associations based on the geometric distance between rays which pass through the camera center and the object. Figure 7.2 illustrates that the rays from two different views for the same object intersect in the 3D location of object. Assuming that the camera parameters are available, namely, the 3D rotation matrix $\mathrm{R} \in \mathbb{R}^{3\times3}$, the camera center $\mathbf{c} \in \mathbb{R}^3$ and the intrinsic matrix $\mathrm{K} \in \mathbb{R}^{3\times3}$, we have the following relationship between image pixel (u, v) and 3D point $\mathbf{q} \in \mathbb{R}^3$

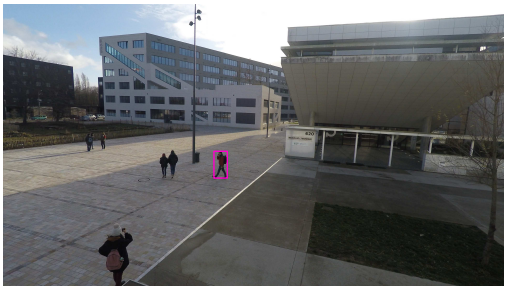
$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathrm{KR}(\mathbf{q}-\mathbf{c}) = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \mathrm{R}(\mathbf{q}-\mathbf{c}), \tag{7.4}$$

with $\lambda \in \mathbb{R}$. Then, the ray $\mathbf{r} \in \mathbb{R}^3$ which passes through the camera center $\mathbf{c}$ and object $\mathbf{q}$ can be expressed as

$$\mathbf{r} = \mathbf{c} + t(\mathbf{q}-\mathbf{c}) = \mathbf{c} + \lambda t \mathrm{R}^{\mathrm{T}} \begin{bmatrix} \frac{u-c_x}{f_x} \\ \frac{v-c_y}{f_y} \\ 1 \end{bmatrix} = \mathbf{c} + s\mathbf{n}, \tag{7.5}$$

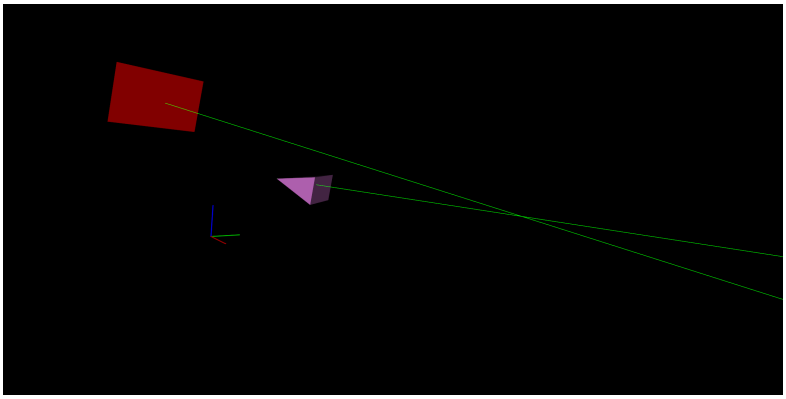
where $s = \lambda t$ is the constant for scale, and $\mathbf{n}$ is the direction vector of $\mathbf{r}$.

Let $\mathbf{p_i} = (u_i, v_i)$ and $\mathbf{p_j} = (u_j, v_j)$ denote the pixels on the image planes of two cameras. The distance between their corresponding projection rays $\mathbf{r_i}$ and $\mathbf{r_j}$ is



(a) Top view

(b) Egocentric view



(c) 3D visualization of projection rays.

Figure 7.2: Illustration of geometric distance prior. (a) and (b) show a synchronised pair of frames from the static overview camera and the egocentric camera. (c) visualizes the cameras (two polygons) and the projection rays (green lines) which intersect in 3D for a correct association across two views, marked by red rectangle on (a) and (b).

$$d_{geo}(\mathbf{r_i}, \mathbf{r_j}) = \frac{\left| \overrightarrow{\mathbf{c_1 c_2}} \cdot (\mathbf{n}^i \times \mathbf{n}^j) \right|}{\left\| \mathbf{n}^i \times \mathbf{n}^j \right\|}, \tag{7.6}$$

where $\overrightarrow{\mathbf{c_1 c_2}}$ represents the baseline connecting the two camera centers, $\mathbf{n}^i$ and $\mathbf{n}^j$ are the direction vectors in Equation (7.5). If $\mathbf{p_i}$ and $\mathbf{p_j}$ correspond to a same point in 3D, then $d_{geo}(\mathbf{r_i}, \mathbf{r_j})$ will be equal to zero as $\mathbf{r_i}$ and $\mathbf{r_j}$ intersect. In the case where $\mathbf{p_i}$ and $\mathbf{p_j}$ do not correspond to a same point in 3D, $d_{geo}(\mathbf{r_i}, \mathbf{r_j})$ is then considered as the cost to punish the false association between $\mathbf{p_i}$ and $\mathbf{p_j}$.

7.4.2 Cheirality constraint

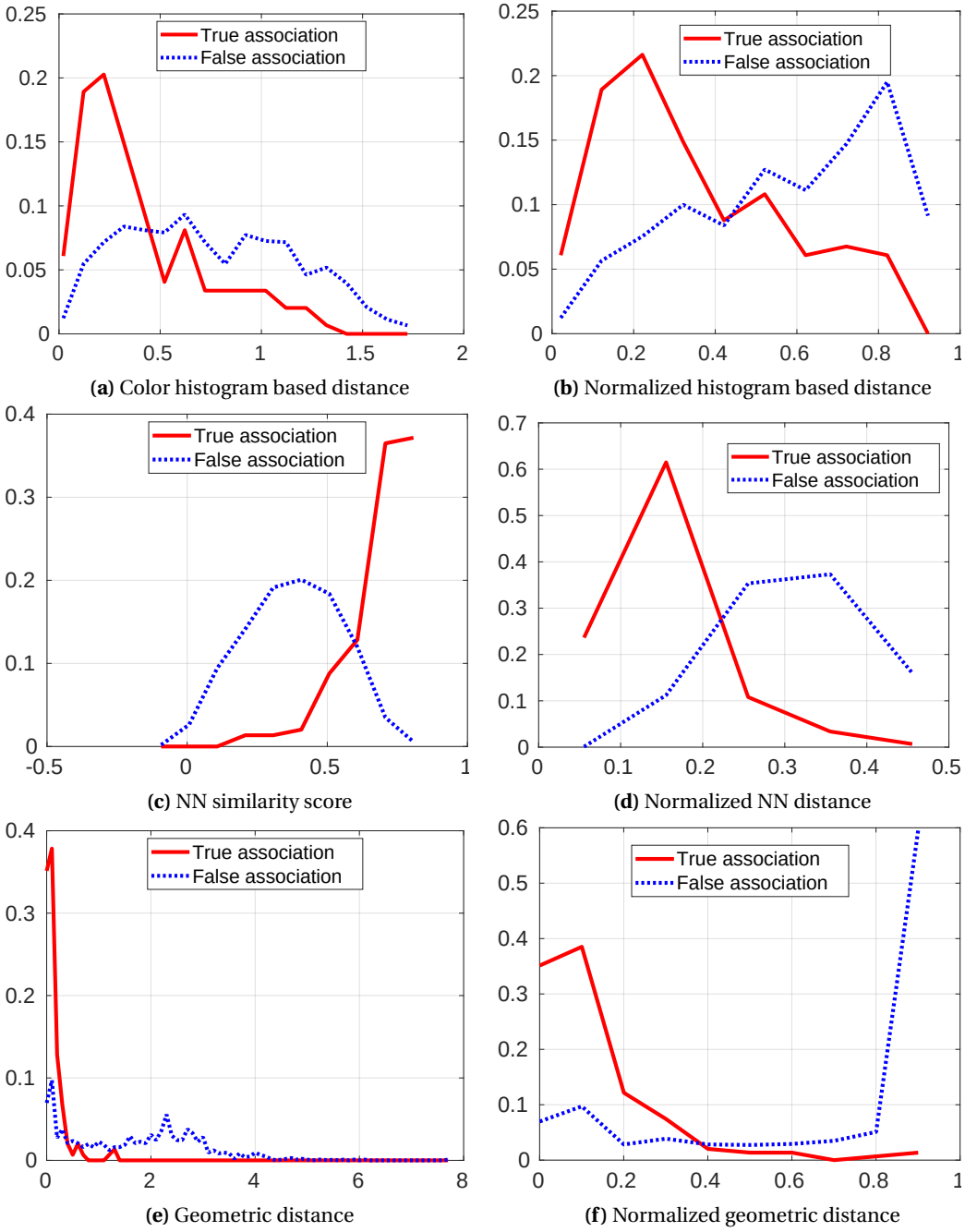
A basic geometric constraint is that the object corresponding to correctly associated detections should be located in front of both cameras. This property is denoted as *cheirality* [61] in the computer vision community. We note that the cheirality constraint is an additional piece of information, different from the distance between the 3D rays, as the latter does not provide any cues about the relative location of the pedestrian with respect to the cameras. It is also independent on the metric scale factor (used to convert distances in metric units), which may be unavailable when the relative pose between the cameras is computed only up to the scale. Given the 2D measurements on pixels for two cameras $\mathbf{p_i}$ and $\mathbf{p_j}$, the 3D point $\mathbf{q}$ can be obtained by triangulation [60]. The constraint is then expressed as $z(\mathbf{q}, \mathbf{c_1}) > 0$ and $z(\mathbf{q}, \mathbf{c_2}) > 0$, where $z(\mathbf{q}, \mathbf{c})$ denotes the depth value of the 3D point $\mathbf{q}$ in the reference system of the camera $\mathbf{c}$. We tighten the camera front constraint as follows, in order to avoid considering the first-view camera wearer as a potential association due to geometry estimations errors:

$$z(\mathbf{q}, \mathbf{c_1}) \geqslant 0.5 \quad \text{and} \quad z(\mathbf{q}, \mathbf{c_2}) \geqslant 0.5. \tag{7.7}$$

This crisp constraint can be easily taken into account by setting the corresponding cost in the association matrix to a very high value when the cheirality constraint does not hold.

7.5 Rule-based combination

7.5.1 Combination rule



(a) Color histogram based distance

(b) Normalized histogram based distance

(c) NN similarity score

(d) Normalized NN distance

(e) Geometric distance

(f) Normalized geometric distance

Figure 7.3: Distribution of true associations and false associations for different costs before normalization.

For the rule-based data fusion, it is not straightforward to identify an effective combination rule without prior knowledge about the characteristics of individual sources. In order to explore an efficient combination of association costs, we investigate first the statistical distribution of individual costs for the true and false associations separately.

As the distribution of true associations and false associations for the appearance based costs and the geometric costs are distributed on different value scales as shown in Figures 7.3a, 7.3c and 7.3e, it is necessary to normalize different costs before combining them. Depending on the considered costs, different normalization techniques are used.

1. **Distance measure.** For the distance metrics ranging in $[0, +\infty)$, such as the color histogram

distance and the geometric distance, we transform them in the range of $[0,1]$ with a tanh function, which allows us to preserve the influence of small values and reduce that of large values. The normalized distance is thus computed as $d'_{hist} = \tanh(d_{hist})$ for the color histogram distance and $d'_{geo} = \tanh(d_{geo})$ for the geometric distance.

2. **Cosine similarity.** The appearance based cost using the learning approach is derived from the cosine similarity, called s_{nn}, which ranges in $[-1,1]$, Min-max normalization is thus applied as suggest in [69]. By subtracting the normalized similarity from 1, we obtain the normalized distance for the appearance based cost using learning approach, denoted by $d'_{nn} = \frac{1}{2}(1 - s_{nn})$.

The distribution of costs after normalization is shown in Figures 7.3b, 7.3d and 7.3f. We note that the appearance based costs follow a bell-shaped distribution, while the distribution of the normalized geometric cost exhibits different behavior. For the combination rule, we can apply either a weighted sum rule (or weighted arithmetic mean when weight sum is equal to 1)

$$d_{+}^{geo,app} = w_{geo} d'_{geo} + w_{app} d'_{app}, \tag{7.8}$$

where the potential variation of the appearance based distributions requires that the weights w_{geo} and w_{app} be tuned, along with the non-association cost γ, or the product rule

$$d_{*}^{geo,app} = d'_{geo} d'_{app}, \tag{7.9}$$

which may be particularly adapted to the profile of the geometric distances. Note that this rule is a particular case of weighted geometric mean where weighted are equal to 1. However, conversely to the arithmetic mean case, for product, we do not fit weights.

The considered combination rule is intended to be applied to an arbitrary pair of synchronized images, as for example a photo of an event taken by a participant at ground level, along with the corresponding photo from a static overview camera. One may wonder in this case whether the assumptions made to support the cost aggregation strategy in Equation (7.9) would be generally valid. Although the distributions presented in Figure 7.3 are generated from data acquired in a specific location with the same camera pair, we expect that for various scenes the distances for true/false associations will follow the same families of distributions. Moreover, the product rule does not involve any parameters that require tuning depending on the slightly varying parameters of the distance distributions. The only parameter of the method which depends on the experimental conditions is the non-association cost γ (Equation (7.2)). The choice of γ allows also for favoring pedestrian associations with the risk of generating false matches (high value of γ) versus enforcing a stricter matching policy with the risk of missing some harder associations (low value of γ).

7.5.2 Experiment and results for rule-based combination

In our experiments, we evaluate the association performance for the geometric costs, appearance based costs, as well as for different combination rules on the dataset *Building entrance* (see Section 2.1.2). We also study the influence of the cheirality constraint on the association performance. Note that the static camera is registered as presented in Section 2.1.2 . To obtain the extrinsic parameters of the mobile camera in the fixed reference system, we first compute relative rotation and translation between the mobile and static cameras and then combine the relative pose with the extrinsic parameters of the static camera. The scene scale is determined by the constraint of mobile camera height with respect to the ground. The height is set to $1.5m$ by considering the mobile camera is held at eye level by a person whose height is $1.60m$, and that the distance from eyes to the top of head is about $0.1m$.

Given an pair of images, the implementation pipeline consists in applying first the pedestrian detectors to extract the bounding boxes in each image and then generating geometric and different appearances features. The baseline for detectors we consider in this work is the widely popular

object detector **YOLO** [131]. The second detector **Idemia** is related to the representation extraction algorithm introduced in Section 7.3.2. Regarding the appearance features for each detected pedestrian instance, the baseline, denoted as **Hist**, is a color histogram distance computed as follows. For each bounding box, we consider 32 bins in the HSV color space, where we used 8 bins for partitioning the H channel values, and 2 bins for S and V channels respectively, and **NN** is the descriptors extracted from the deep neural network as presented in Section 7.3.2. Regarding the geometric features, we consider two different ways to integrate it into different fusion strategies. We select the top-center point of bounding box for projecting it in 3D and compute explicitly the 3D geometric distance between the projected rays for an pair of bounding boxs. The cost matrix is fed with different distance measures for all detected bounding boxes. Following our analysis of the distribution of correct and incorrect association distances, the non association cost is set to $\gamma = 0.25$ for the sum combination, and to $\gamma = 0.1$ for the product combination, respectively.

To determine if a pair of bounding boxes is correct, we compare this pair of bounding boxes with the ground truth. For each person, the ground truth is given by a pair of annotated bounding boxes in two views. For each view, we check if the predicted bounding box corresponding to the ground truth bounding boxes in the considered view by setting the threshold to 0.3 for Intersection over Union (IoU) between bounding boxes. The evaluation for the performance of association is based on the average accuracy over all dataset. For each image pair in dataset, the association accuracy is:

$$\text{Accuracy} = \frac{\#\text{ true positives} + \#\text{ true negatives}}{\#\text{ of persons}} \tag{7.10}$$

One important consideration is that the final result of the DA algorithm is influenced at the same time by the association step, but also by the detection algorithm. The coupling between the two steps may be quite complex: a high precision/low recall detector causes inevitably missed associations, while a high recall/low precision detector may allow for a good final result only if the non-association cost and a good distance matrix help in rejecting the false positive detections. In order to avoid the influence caused by the difference in performance between the detectors, we first evaluate the association accuracy without considering the performance of detection and then evaluate the global detection and association pipeline:

1. **Association evaluation** We focus first on evaluating the performance of the DA step specifically. To this aim, we consider the default detection thresholds for the two detectors, which allow for a reasonable compromise between precision *Pr* and recall *Re*. For the used dataset, we report that $Pr_{\text{YOLO}} = 92.2\%$, $Re_{\text{YOLO}} = 72.9\%$, $Pr_{\text{Idemia}} = 97.6\%$, $Re_{\text{Idemia}} = 84.3\%$. The number of persons (individual) in Equation (7.10) is considered as the number of correct detections *provided by the detectors* in the DA input. This allows for a maximum DA accuracy of 100% independently of the detector performance.

2. **Global evaluation** In the second part of the experiments, we compare the entire pipeline (detection and association). By considering the number of persons in Equation (7.10) as the number of correct *ground truth* pedestrians, we account for errors in detection and in association. In other words, the accuracy level is bounded by the detector recall in this case. Let us denote **Accuracy*** this stricter accuracy measure. In order to provide a fair comparison, we tune YOLO to have a similar recall as the second detector (i.e. 85%); consequently, YOLO's precision falls down to 49.8% due to the higher rate of false positives.

The performance of association global evaluation following the different cost functions and combination rules is summarized in Table 7.2 and Table 7.3. In the reported data, the costs for different methods being evaluated are listed in Table 7.1. The overall results highlight clearly that geometric priors improve significantly the performance of the association based on appearance in two different ways. The first one is by performing the combination rules for geometric cost and appearance based cost, and the second one is the use of the cheirality constraint. Both priors contribute in independent ways to filtering the false associations, while requiring different preliminary steps for their use (a relative camera pose estimation in the case of the cheirality constraint,

Table 7.1: Association costs for different evaluated methods of rule-based combination.

Name	Association cost
Geometry only	d'_{geo}
Appearance only	appearance based distance, either **Hist** (d'_{hist}) or **NN** (d'_{nn})
Sum combination	weighted sum of geometry and appearance costs ($w_i = 0.5$)
Product combination	product of geometry and appearance costs

Table 7.2: Association accuracy. We compare the performance of the combination of geometric cost with different appearance-based features, including the color histogram (Hist) and the descriptor extracted from neural network (see Section 7.3.2), as well as with different pedestrian detectors (YOLO and Idemia). For different case, we compare the combined cost with the single use of individual appearance-based cost or geometric cost.

Cost Function		**Accuracy**		
Method	Cheirality constraint	YOLO [131] + Hist [67]	Idemia [63] + Hist [67]	Idemia [63] + NN [63]
Geometry only	No	0.610	0.654	0.654
	Yes	0.851	0.849	0.849
Appearance only	No	0.574	0.537	0.665
	Yes	0.628	0.579	0.820
Sum combination	No	0.724	0.782	0.709
	Yes	**0.864**	**0.895**	0.890
Product combination	No	0.614	0.660	0.707
	Yes	0.839	0.831	**0.896**

and an additional PnP+scale estimation for the 3D distance prior). Regarding the combination rules, the performance improvement varies depending on the specific feature extractors. If the appearance based distance provides a good separability, as in the case of the NN descriptor, then the use of the product aggregation is effective and avoids further parameter tuning. In the case of a less discriminative distance, such as the histogram-based distance in our case, the sum rule may provide a better performance while requiring at the same time a more careful tuning of the cost weights and non-association cost. Finally, Table 7.3 presents the global performance of a detection-association pipeline in which the accuracy measure is affected by the two steps in different ways. In this case too, the experiments underline the significantly positive impact in all situations of the additional geometry information.

Table 7.3: Global method accuracy (the two detectors are tuned for the same recall level).

Cost Function		Accuracy*	
Method	Cheirality constraint	YOLO* [131] + Hist [67]	Idemia [63] + NN [63]
Geometry only	No	0.571	0.599
	Yes	0.684	0.742
Appearance only	No	0.360	0.648
	Yes	0.482	0.751
Sum combination	No	0.611	0.709
	Yes	**0.714**	**0.784**
Product combination	No	0.550	0.639
	Yes	0.663	0.779

7.6 Data-driven combination

The rule-based combination of geometric cost and appearance-based cost have achieved better performance than the single use of individual cues on the tested dataset. However, it requires to look into carefully the statistically distribution for the involved costs in order to find an optimal parameter or rule. An alternative way is to learn automatically the combined score from these two different cues. In our work, we consider the association problem as a binary classification problem and explore how to combine the geometric feature and the appearance-based feature with the neural network based on multilayer perceptron (MLP) which is a simple but powerful tool for the regression and classification problems.

7.6.1 Multilayer Perceptrons

As one typical category of the neural networks, the multilayer perceptron (MLP) is a type of classifier built upon the conception of perceptron which was initially conceived by Warren McCulloch and Walter Pitts [101] and then developed by Frank Rosenblatt [112] with the inspiration from the biological neuron. As the unit of MLP, a perceptron, also known as neuron illustrated in Figure 7.4 accepts an one dimensional vector as an input and provides a scalar value as output by passing a weighted sum over the input vector through a non linear function known as activation function, which can be summarized with the following mathematical formula:

$$y = \phi(\sum_{i=1}^{n} w_i x_i + b) = \phi(\mathbf{w}^{\mathrm{T}}\mathbf{x} + b), \tag{7.11}$$

where $\mathbf{x}$ is the input vector, $\mathbf{w}$ is the weight vector applied on input element, b is the bias and ϕ is called the activation function.

The MLP is then formed by the stack of more than one neuron. It has typically one input layer taking the input vector, and successive hidden layers including arbitrary numbers of neurons for each layer followed by the output layer to predict the final decision about the input, where each neuron is connected with every neuron in its previous layer by the relationship described in Equation (7.11).

Like other typical models fitting algorithms, the essential part of MLP is to estimate the unknown parameters which are the weights and bias associated with each neuron. It is usually based

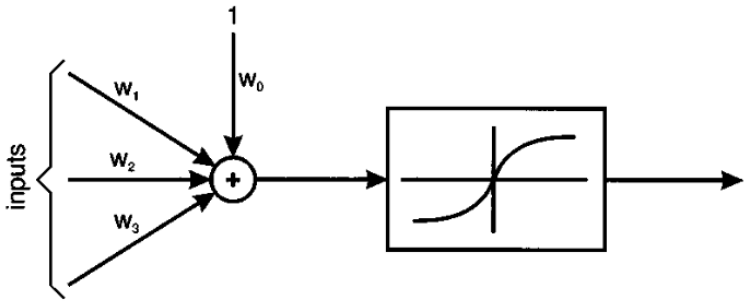


Figure 7.4: Overview of perceptron [7].

on the *backpropagation* of the loss function related to a defined error measure between the predicted outputs and the desired outputs during the training process. By computing the gradient of a loss function $\mathscr{L}$ with respect to a weight w with the chain rule, w is iteratively updated as follows:

$$w^{n+1} = w^n - \eta \bigtriangledown \mathscr{L}, \tag{7.12}$$

where η is called the learning rate.

Compared to other variants of neural networks like the convolutional neural network which are widely used to work on multiple dimensional data like the image for most of computer vision tasks such as image classification and object recognition, the multilayer perceptron architecture is a more appropriate choice when it comes to one dimensional input feature. For the association problem of person detections in our case, the input feature vector is formed by the appearance-based cues or geometric cues, or the combination or them. Although it is also possible to form the input for the appearance-based cues by 2D image patches extracted with the detected bounding box, we use directly the 1D pre-defined or learned visual feature descriptor as the input of a multilayer perceptron network to focus more on the combination process instead of learning additionally the visual feature representation.

7.6.2 Binary classification

The binary classification is the task of assigning each individual element with a class label $y \in \{0,1\}$, where in general 0 means negative and 1 means positive. It is involved in many applications including medical testing, quality control or object recognition. When it comes to our association problem, it refers to predict whether a pair of detections is a true or false match given the input features associated with this pair of detections. The loss function used for training the MLP for this task is by default the binary cross entropy function defined as follows

$$\mathscr{L}(\mathbf{w}, \mathbf{x_n}) = -\frac{1}{\mathrm{N}} \sum_{n=1}^{\mathrm{N}} \left\{ y_n \log \hat{y}_n + (1 - y_n) \log(1 - \hat{y}_n) \right\}, \tag{7.13}$$

where $\hat{y}_n = f(\mathbf{w}, \mathbf{x}_n)$ is the predicted output label for observation $\mathbf{x}_n$ and y_n is the target label. When it encounters the imbalance between the number of positive samples and negative samples, the above loss function can be replaced by

$$\mathscr{L}(\mathbf{w}, \mathbf{x_n}, p_c) = -\frac{1}{\mathrm{N}} \sum_{n=1}^{\mathrm{N}} \left\{ p_c y_n \log \hat{y}_n + (1 - y_n) \log(1 - \hat{y}_n) \right\}, \tag{7.14}$$

where p_c is the weight of positive sample and can be computed as the ratio between the number of negative samples and the number of positive samples in the training dataset.

In our case, we aim at deriving a matching score $s_{ij} \in [0;1]$ from the binary classification task for a pair of detections which are characterized by their visual representations $(\mathbf{v}_i, \mathbf{v}_j)$, as well as

by their geometrical representations $(\mathbf{g}_i, \mathbf{g}_j)$. $\hat{y}_n$ is the predicted matching score for a pair of detections. The target score will be 0 for a false pair of detections and 1 for a true pair. The whole process may be performed in two different manners:

1. **Partial learning**: two independent feedforward neural networks are used to learn a matching visual score $s^v_{ij} = f^v(\mathbf{v}_i, \mathbf{v}_j)$ and a matching geometric score $s^g_{ij} = f^g(\mathbf{g}_i, \mathbf{g}_j)$ respectively, and the global score s_{ij} is obtained by combining s^v and s^g using an explicit combination rule (e.g. weighted sum, product)

2. **Joint learning**: a single network is used to learn the global score $s_{ij} = f(\mathbf{v}_i, \mathbf{v}_j, \mathbf{g}_i, \mathbf{g}_j)$. This approach can fully take into account the co-occurrences present in the visual and geometric sources of information, and may be denoted as an early fusion as opposed to the previous approach which falls into the late fusion category.

It remains to define the geometric representations associated to the pedestrian detections and the architecture of MLP.

7.6.3 Input features

The visual features are represented by a 1D descriptor $\mathbf{v}$ as presented in Section 7.3. For the geometric features, we adopt a ordered set of 3D lines $\mathbf{g} = (\mathbf{l}_1, \mathbf{l}_2, \mathbf{l}_3, \mathbf{l}_4)$ corresponding to the projections rays of four coordinates of detection box. Specifically, the 3D lines are represented in the camera coordinate frame instead of the world coordinate frame so that the variation of lines are constrained by relative camera pose instead of the absolute pose. As for a 3D line, we use the Plücker homogeneous line coordinates [60]. Given a line joining the two 3D points $\mathbf{A}$ and $\mathbf{B}$, the the Plücker homogeneous line coordinates is derived from the 4×4 skew-symmetric homogeneous Plücker matrix L

$$\mathrm{L} = \mathbf{A}\mathbf{B}^{\mathrm{T}} - \mathbf{B}\mathbf{A}^{\mathrm{T}}, \tag{7.15}$$

with the Plücker homogeneous line coordinates $\mathbf{l}$ being a 6-vector

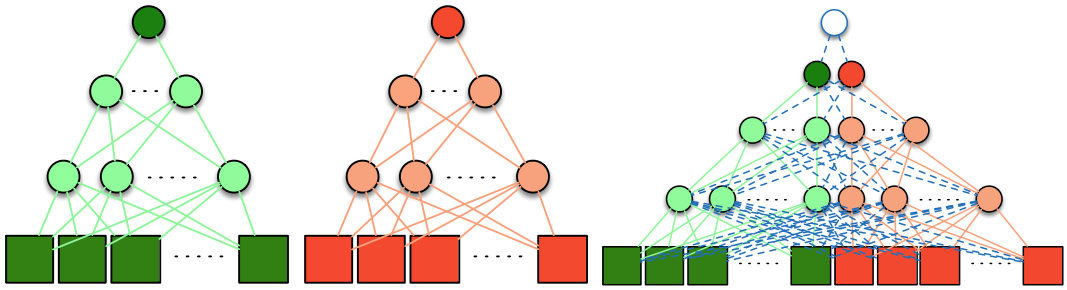
$$\mathbf{l} = \{\mathrm{L}_{12}, \mathrm{L}_{13}, \mathrm{L}_{14}, \mathrm{L}_{23}, \mathrm{L}_{42}, \mathrm{L}_{34}\},$$

which can be normalized as $[-1, 1]^6$ for the numerical stability.

Note that in some cases when the relative yaw angle between cameras is large, the 3D box around the pedestrian is flipped during projection in the images and the order of the lines is switched between the two views. Even if this situation can be detected with an additional check, we did not implement it. Indeed, the geometric score does not need to reflect highly accurately the inter-line distances (as it would be the case in a 3D reconstruction task), but merely to guide the association problem towards the instances with the lowest cost relative to the rest of the candidates.

7.6.4 Architectures

For all the learning tasks, we use MLP architecture with two hidden layers followed by ReLU activations and the last output layer followed by Sigmoid activation. The hidden layer widths depend on the input sizes will be specified in Section 7.6.5. For the joint learning task, all the preliminary tests using a standard training procedure reached a performance which was intermediary with respect to that provided by the independently trained MLP. This is in agreement with reports in the literature underlining the need to guide gradually the training process [50, 107], and we assume that this is due to the very different nature and size of the features, which makes their optimal joint exploitation quite difficult. We adopt a similar approach as [50] in the context of exploiting wind and pressure data for cyclone trajectory prediction, in order to guide the joint learning process as follows (see also Figure 7.5). We build the layers of the final MLP as concatenations of the independently trained networks, including the initial last layers which will thus create a two-neuron layer



(a) Visual MLP **(b)** Geometric MLP **(c)** Joint MLP

Figure 7.5: The three networks used for score learning: (a) visual MLP; (b) geometric MLP; (c) joint learning.

in the new architecture, on top of which we add a new single neuron output. This allows us to fully clone two independently trained models into the new one, and at the same time we set to 0 the weights of all the new connections which correspond to the correlations between the visual and geometric data (the dashed lines in Figure 7.5). Regarding the weights for the final added layer, our tests showed that the most stable initialization is to set to 1 the weight for the best performing descriptor, and to 0 for the other. In this way, we initialize a network which performs as the previous best one, and which has an additional component properly initialized to exploit the additional feature. Then, we train the concatenated model with a small learning rate in two consecutive steps as follows.

Step 1: we freeze all the network parameters, except the final newly introduced two weights. In this case, we only learn the weighting parameters between the visual score and geometric score. This is close to performing the weighted sum combination for the outputs of partial learning. For the loss function, we adopt the binary cross entropy on the global score s_{ij}, denoted by

$$\mathscr{L}_1 = \mathscr{L}_{s_{ij}}. \tag{7.16}$$

Step 2: we unfreeze all the newly introduced weights (the dashed lines in Figure 7.5) to explore the correlation between two sources, and we add at the same time two additional terms to the loss function, in order to maintain the outputs of the two parts of the network (the darkly red and green nodes in Figure 7.5) as valid score outputs:

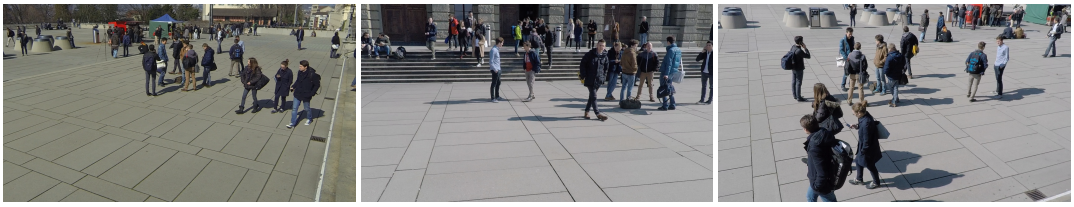
$$\mathscr{L}_2 = \mathscr{L}_{s_{ij}} + \mathscr{L}_{s^{v'}_{ij}} + \mathscr{L}_{s^{g'}_{ij}}, \tag{7.17}$$

where $s^{v'}_{ij}$ and $s^{g'}_{ij}$ present the score outputs for two parts of the network before the global output layer.

7.6.5 Experiments and results for data-driven combination

As the dataset of *Building Entrance* collected by ourself is a small dataset in terms of numbers of image pairs and pedestrians occurrences, the data-driven fusion strategy is thus not suitable for this dataset. In our experiments, we consider a much larger public dataset *WILDTRACK* (see Section 2.1.4) and choose the images captured by three cameras as illustrated in Figure 7.6 to evaluate the data-driven combination method (**Learned**) as well as the rule-based method (**Explicit**) based on the association accuracy.

The implementation of the association algorithm is as the same as before. The only difference is the additional step to learn the matching score for each pair of detections fed into the Hungarian algorithm. We still use object detectors **YOLO** and **Idemia** introduced in Section 7.3.2. The input appearance features based on **Hist** remains a vector containing 32 bins in the HSV color space. For the learning based representations, we first consider the vector representation of 512 float computed by our learning algorithm (Section 7.3.2), denoted as **Des2**. The other one is an embedding



(a) Camera C1 **(b)** Camera C5 **(c)** Camera C7

Figure 7.6: Example of synchronized images for dataset WildTrack in multiple views.

vector of 256 float extracted by the "pre-trained model person-reidentification-retail-0265" [111] in Openvino, denoted as **Des1**.

Regarding the geometric features, we consider two different ways to integrate it into different fusion strategies. For Rule-based fusion, we select the top-center point of bounding box for projecting it in 3D and compute explicitly the 3D geometric distance between the projected rays for an pair of bounding boxs as before. For Data-driven fusion, we deploy a vector of 24 floats which is the concatenation of the coordinates of four projections lines corresponding to four points of bounding box. Each line is represented by six Plucker homogeneous line coordinates with L2 normalization.

Training details for Data-driven fusion. We split a sequence of 400 frames in the dataset of *WILDTRACK* into three parts: the first 200 frames for training, the last 100 frames for validation and the rest for test. To prevent over-fitting, we set a small number of neurons for hidden layers limited by the number of training samples and the size of input features. Specifically, the geometric MLP consists of two hidden layers with 30 and 20 neurons, and the visual MLP with 20 neurons and 10 neurons respectively. The joint MLP is thus composed by 50 neurons for the first hidden layer and 30 neurons for the second layer. For the partial learning, We train first 1000 epochs for the model of visual MLP and geometric MLP separately using ADAM optimizer with a learning rate of $10e^{-4}$ and then 1000 epochs using SGD optimizer with a learning rate of $10e^{-8}$ for refinement. For joint learning, we use SGD optimizer and first train 200 epochs with a learning rate of $10e^{-6}$ for the freezing step and another 200 epochs with a learning rate of $10e^{-7}$ for the unfreezing step. During the training, we apply L2 regularization to avoid over-fitting by setting regularization parameters equal to 0.01 for visual MLP and Joint MLP, and 0.001 for geometric MLP.

The association costs used by the different methods for **Learned** are presented in Table 7.4. For each pair of cameras, we apply three different kinds of combination of detectors (YOLO and Idemia) and visual descriptors (**Hist**, **des1** and **des2**) to test the performance of different fusion methods as well as single source based methods. In our experiments, we have tested several camera pairs which present different geometric configurations and difficulty levels related to association with geometric features. We found that the single use of geometric features can achieve very high performance and the complementary visual features become much less useful when the geometric configuration is very favorable, for example, when two cameras stand closely and have large overlapping areas between their views. The results for such special cases are not reported in our evaluations as the fusion techniques become less necessary in such special cases. We finally present the results for two camera pairs with more difficult configurations and the results are reported in Table 7.5 and 7.6.

Rule-based fusion VS Data-driven fusion. We first compare the data-driven fusion method (column for **Explicit** in Table 7.5 and 7.6) with the rule-based fusion method (column for **Learned** in Table 7.5 and 7.6). Regarding rule-based fusion, we observed that the sum combination achieves the best performance in most of cases. However, we also notice the cases where the combination of geometric sources and visual sources fails to surpass the performance of single modality, for example when using YOLO and visual descriptor **des1** for WILDTRACK C1-C7. Considering the different value scales of geometric cost and visual cost, the effectiveness of rule-based fusion requires a careful parameter tuning for the weight of sum combination and the cost of non association. Compared to the rule-based fusion with explicit distance, the data-driven method achieves

Table 7.4: Association costs of different methods for Rule-based method (**Learned**).

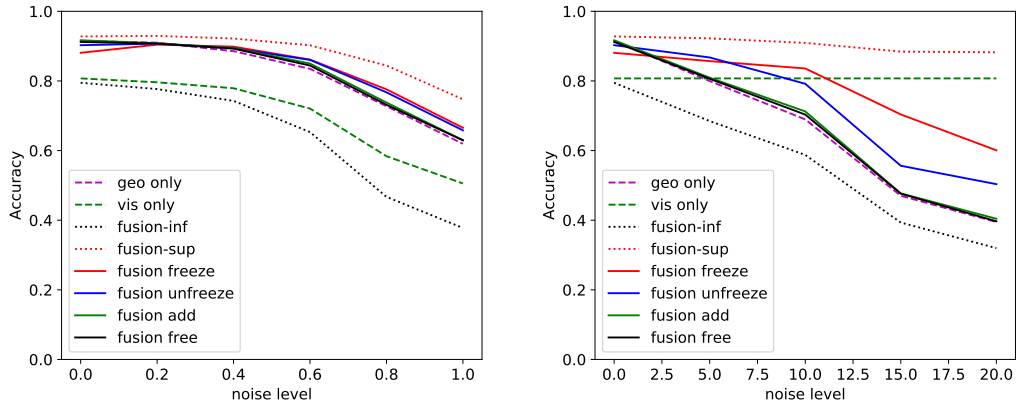
Method name	**Association cost**
Visual Only	Output of Visual MLP, s^v_{ij}.
Geometry Only	Output of Geometric MLP, s^g_{ij}.
Sum	Weighted sum of s^v_{ij} and s^g_{ij}.
Product	Product of s^v_{ij} and s^g_{ij}
Concatenation-free	Output of Joint MLP, s_{ij}. Training without consecutive steps.
Concatenation-freeze	Output of Joint MLP, s_{ij}. Training with consecutive steps, the first step with freezing weights.
Concatenation-unfreeze	Output of Joint MLP, s_{ij}. Training with consecutive steps, the second step with unfreezing weights.

significant improvement in terms of association accuracy for single modality as well as the fusion of multi-sources, with the benefit of supervised signal. For geometric features, by feeding the projected rays of four coordinates of bounding box, the network can explore more geometric relevance between two detections, while the explicit method provides only the geometric distance between a single pair of top-center rays. When it comes to visual features, we observed that sometimes the performance with the learned visual score is below that of explicit cosine distance, such as in the case using YOLO and desc1 for WILDTRACK C1-C7. One reason may be the performance of learned method is limited by the number of samples for training.

Comparison for joint learning and score combination. We discuss the effectiveness of learning based fusion by Joint MLP with different training techniques, denoted as *concatenation-free*/ *freeze*/ *unfreeze*, and score based fusion by rule combination of learned score from partial learning (Visual MLP or Geometric MLP), denoted as *Sum* for the column of **Learned**. From the reported tables, both Joint MLP and score combination achieve close and competitive improvements compared to the single modality in most of cases. The close performance between Joint MLP and score combination shows that in our cases the correlation between geometric features and visual features is weak.

Impact of sources imprecision. We also study the influence of source imprecision on the proposed data-driven fusion method. In our experiments, we degrade the reliability of visual features and geometric features during inference step, by introducing the uniform noise on the position of bounding box as well as on the camera pose including camera rotation and position. The perturbation on bounding box will degrade the quality of geometric feature and visual features at the same time, while the perturbation of camera pose will only degrade the quality of geometric features. The results are shown in Figure 7.7. Without degradation, the individual geometric feature achieves very competitive performance compare to the fusion of multi-sources. However, as the sources imprecision increases, we notice that the fusion method maintains the best performance while the single modality methods drop quickly.

We note that the benefit of the cheirality constraint depends on the configuration of cameras, namely on the relative orientation of their optical axes and the location of the area of interest in the common field of view. Therefore, in some cases, such as the camera pair C5-C7 (see Table 7.5), the cheirality constraint does not help as the triangulation for a false association is also prone to be located in front of both cameras.



(a) Errors on bounding box

(b) Errors on camera pose

Figure 7.7: Influence of source imprecision on association accuracy for data-driven fusion method . The reported results are tested with the detector YOLO and visual descriptor desc1 on camera pairs C5 and C7 for dataset wildTrack. (a) shows the influence of errors on bounding box where the noise level indicates the ratio of bounding box size. (b) shows the influence of camera pose error with different values of rotation error on degree when the position error is fixed at 0.5m.

Table 7.5: Association accuracy (%) for WILDTRACK C5-C7

(Det.;Desc.) / Cost		(YOLO;hist)		(YOLO; desc1)		(Idemia;desc2)	
	Cheirality	Explicit	Learned	Explicit	Learned	Explicit	Learned
Visual only	No	28.26	57.41	72.18	80.72	37.83	70.25
	Yes	40.02	66.68	72.18	83.09	45.31	76.80
Geometry only	No	71.21	91.53	71.21	91.52	71.54	90.42
	Yes	71.21	91.53	71.21	91.52	71.47	90.42
Sum	No	78.06	91.91	80.68	**91.68**	75.51	90.59
	Yes	78.19	91.91	80.68	**91.68**	75.37	90.59
Concatenation-free	No	-	91.52	-	91.20	-	90.49
	Yes	-	91.52	-	91.20	-	90.49
Concatenation-Freeze	No	-	**91.99**	-	91.58	-	**90.92**
	Yes	-	**91.99**	-	91.58	-	**90.92**
Concatenation-unfreeze	No	-	91.82	-	91.05	-	90.83
	Yes	-	91.82	-	91.05	-	90.83
Fusion-sup	-	74.74	92.77	90.11	92.78	77.37	93.03

Table 7.6: Association accuracy (%) for WILDTRACK C1-C7

(Det.;Desc.) / Cost		(YOLO;hist)		(YOLO; desc1)		(Idemia;desc2)	
	Cheirality	Explicit	Learned	Explicit	Learned	Explicit	Learned
Visual only	No	43.08	43.78	84.73	71.22	43.82	58.64
	Yes	43.11	43.98	84.73	71.22	43.81	58.72
Geometry only	No	44.26	87.24	44.26	87.24	45.01	86.06
	Yes	44.86	87.24	44.86	87.24	46.03	86.06
Sum	No	54.01	87.70	66.45	88.49	58.59	86.33
	Yes	54.60	87.70	66.81	**88.49**	59.19	86.33
Concatenation-free	No	-	87.70	-	88.42	-	86.74
	Yes	-	87.70	-	88.42	-	86.74
Concatenation-Freeze	No	-	87.67	-	88.06	-	86.71
	Yes	-	87.67	-	88.06	-	86.71
Concatenation-unfreeze	No	-	**87.74**	-	88.38	-	**86.83**
	Yes	-	**87.74**	-	88.38	-	**86.83**
Fusion-sup	-	64.84	89.55	88.36	90.55	64.84	88.55

7.7 Concluding remarks

This chapter presents an overview of the pedestrian transfer from egocentric view to top view formulated as a joint visual-geometric data association problem. We study this task on multiple datasets, either constructed by ourselves or freely available to the community, and the results show that in all cases the fusion between spatial and appearance cues provides some gain in terms of accuracy. The coupling between the two matching criteria may be done in a traditional manner based on the observed statistics, or by learning a combination, provided that enough training data are available.

Due to the strict regulation on the collection and use of visual data in public areas, it has not been yet feasible to evaluate this approach in a high-density scene. However, in such a setting with an increased level of association ambiguity, it would be expected that pedestrian matching would benefit even more from this type of fusion, and this evaluation is a clear perspective of our work.

Conclusion and future work

Conclusion

In this work we address the image-based and camera-IMU/GPS fusion-based localization of camera wearer and target pedestrian within the network of an egocentric mobile camera and static camera. We work on the geometry of multiple views in computer vision and explore the combination of available sources to enhance the accuracy of localization. Based on the proposed algorithms, a conceptual localization framework is designed for the public surveillance task by the law enforcement agent equipped with body-born camera.

In Part II, we look at the problem of relative camera pose estimation which is an elementary part for both epipole localization and the used geometric priors used in the data association problem. To overcome the limitation of image-based estimation for the challenges of repetitive features raised in the urban environment, we rely additionally on the available information from noisy pose sensors. We first investigate an M-estimator based approach which is flexible for the fusion between pose sensors and image estimation but is sensitive to the prior importance of the visual matches between the two views. To improve the quality of visual keypoint matches, we proposed to rely at the same time on local information provided by visual similarity, and by a global geometrical coherence of the transformation between two views. A conservative weighting scheme is derived for combining the two types of cues. The proposed method achieves significant improvement in terms of the accuracy of relative camera pose on a challenging dataset acquired in an urban scenario.

In Part III, we solve the localization of camera wearer by determining the uncertainty region of epipole which corresponds to the position of one camera in the field of reference view. We first show that the presence of outliers in the standard pipeline for camera relative pose estimation not only prevents the correct estimation of the epipole but also degrades the standard uncertainty propagation for the epipole position. In Chapter 4, we propose a majority voting strategy by sampling multiple models based on the standard RANSAC and uncertainty propagation pipeline to construct a continuous probability map for the location of epipole which achieves a higher reliability on the experimental datasets, in contrast with baseline methods. To improve furthermore the precision of epipole estimation with the direct accumulation of multiple solutions, we then introduce the 2D belief function framework in Chapter 5 and address the localization of epipole with multiple models as a fusion task with a large number of sources including outlier ones. In Chapter 6, we propose a strategy based on the clustering and intra-cluster fusion to overcome the challenge introduced by a large number of sources and outlier sources within the 2D belief function framework. The proposed method enhances the consistency of different solutions inside the the same cluster and highlights the inconsistency between clusters. It exhibits more robustness in terms of accuracy and precision when compared on real data with the standard algorithms which provide a singular solution. In addition, we explore afterwards the combination of the clusters with additional evidences including pedestrian detector, GPS position and temporally-consistent frames which constraint furthermore the candidate region for the location of epipole.

In Part IV, we introduce the problem of across-view data association for the task of pedestrian localization. We propose to use the 3D geometric cues with the classical appearance based cues and integrate them into the objection function of data association algorithm. We consider the

explicit score based combination rule guided by the statistical distribution as well as the feature based combination using the data-driven rule based on a neural network in a supervised manner. The experiments show that both combination rules for the geometric cues and the appearance cues achieve a significant improvement in terms of the association accuracy compare to the single use of individual cues on the evaluated datasets. Specifically, the supervised learned score from the input features outperforms the explicit computed association score due to the benefits of additional information about the ground truth. We suggest to use the learning method for association score when a large dataset is available for training. However, it seems that learning the fusion of the 3D geometric and appearance based features is not very competitive to the explicit combination rule of the individual learned score, which suggests that learning could be more effective in scenes for which a stronger correlation exists between the geometric and visual features.

Future work

In Part III, we deal with the epipole localization for our specific purpose. In more general case, the relative pose including rotation and translation is required. Although the proposed method works well for the uncertainty estimation of a 2D epipole, it requires further investigation on the uncertainty estimation for relative pose with 5DOF (the scale is ignored) based on the multiple sampling strategy, due to the challenge of computational cost raised by the high dimensional space during voting step. The proposed belief clustering does not either allow to deal with the issue of high dimension as the modelling of high dimensional objects within the current belief function framework is much more complex. A potential solution is to approximate the process by split voting in the translation (2D) and in the rotation space (3D), instead of voting jointly in five dimensions at the cost of losing correlation information, which requires more investigation about the influence of independent voting on the accuracy and reliability of pose.

The use of additional sources such as the pedestrian detectors and the position sensors have provided a promising result for epipole localization as shown in Chapter 6. The future research can also explore them to improve the relative pose estimation based on the optimization framework. The proposed multiple sampling strategy in Chapter 4 derives a set of ellipses for epipole uncertainty estimation and each ellipse is associated to a set of minimal number of point matches. The pedestrian detectors or position sensors predict furthermore the candidate research region for the true epipole. If the uncertainty area represented by an ellipse is consistent with region predicted by the additional source, it is more likely that the point matches associated to the ellipse are correct. Thus, the future work can explore this information to predict the weight prior of point matches within the optimization framework to improve relative pose estimation. For example, we can compute the presence frequency of point matches belonging to the ellipses which are consistent with the region predicted by the additional sources. More investigations are needed to measure consistency level between the ellipse and the predicted region.

In terms of data association for pedestrian localization in Part IV, the combination of visual similarity based cost and geometric cost improves the accuracy compared to the single use of individual source. However, it is not wise to set a fixed weighting parameter for all the pairs of detections with varying reliability on different costs. A more feasible way is to adjust the weight based on the uncertainty related to each cost. In addition, the uncertainty information can be also integrated into the association cost as in [173] to improve the robustness of assignment problem. The future work could be possible to develop a full uncertainty propagation pipeline from the initial input errors to the association cost. For the visual cost, one may consider to model feature uncertainty as in [169] and integrate it into the similarity between features. For the geometric cost, there are additional error sources which have been set aside for the moment due to their lower impact but which could be considered as well, including the detected bounding box coordinates, the 2D-3D landmarks used in algorithm PnP for the pose estimation of static camera and the 2D-2D point matches for the relative pose estimation between the mobile camera and the static camera.

Besides, the supervised learning based score exhibits more accuracy than the explicitly com-

puted score for the association problem when a large dataset is available. However, the predicted association costs have to be furthermore fed into the Hungarian algorithm to obtain the final association results. It becomes nowadays feasible to perform the whole association pipeline in an end-to-end manner. To do that, the future work may investigate the deep Hungarian net recently proposed by [164] which allows one to perform the Hungarian algorithm in a differentiable manner so that the whole pipeline for data association is trainable.

Finally, note that, in the same manner that BFT has shown its effectiveness in modelling uncertainty and imprecision for epipole localization, data association could have been handled in this framework. Indeed, numerous works, for instance [10, 39, 57, 103, 132], have already been proposed for such a problem. In this PhD, our first choice was to rely on learning, but further improvements may be made. A comparison or collaboration with uncertain reasoning approaches could be considered in order to understand and overcome the specific learning challenges triggered by limited training data or imprecise assumptions about the problem inputs. Therefore, this joint localization in egocentric and third-person views could be an additional avenue for developing a line of research coupling rich uncertainty theories with powerful statistical learning algorithms.

Appendix

In this part, we present how to transfer the 2D location of a target in the image plane to the 3D location in the global geo-reference, which can is involved in Part III and Part IV. Given the 2D coordinates $\mathbf{p} = (u, v)$ of a pixel, it takes three steps to compute its 3D coordinates in geo-reference: 1) the geo-registration of the camera with respect to the local world coordinate, 2) computing the 3D coordinates of $\mathbf{p}$ in the local world coordinate frame, and 3) the transformation from the local world coordinate to the global geo-reference on Earth.

Camera registration

The registration of the camera consists in computing the rotation and position of camera with respect to the local world coordinate frame, which relies on the 2D-3D point matches using algorithm PnP [49]. It is used to calibrate the static camera when we collect the dataset *Building entrance* (see 2.1.2). In our work, we manually choose landmarks as illustrated in Figure 7.8 to build the 2D-3D point matches. The 3D positions of landmarks are derived based on the distance measurements prvided by a hand-held laser device, and they are expressend in the local world coordinate frame as shown in Figure 7.8.

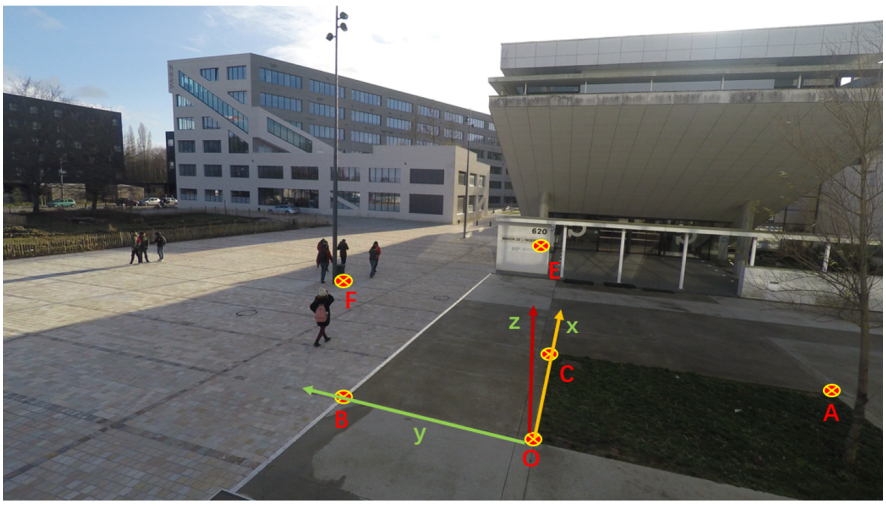


Figure 7.8: Landmarks and local world coordinate frame for camera registration

Computing 3D position of the 2D target

There are two ways to compute the 3D position of the 2D target in the local world coordinate frame depending on the available information. If the poses of both cameras with respect to the local world coordinate frame are known, the 3D position of target can be derived following triangulation using the pair of 2D detections $\mathbf{p} = (u, v)$ and $\mathbf{p}' = (u', v')$ of the target in two camera views. Or, alternatively, if only one of the cameras is registered, we can first compute the projection ray corresponding to the 2D pixel $\mathbf{p} = (u, v)$ under the pinhole camera model and then derive the 3D

position of target by the intersection of the projection ray with the plane of ground (Z = 0 in our case).

Transformation from the local world coordinate frame to the global geo-reference

The rigid body transformation between two coordinate frames can be expressed as

$$\mathbf{X}' = \begin{bmatrix} \mathrm{X}' \\ \mathrm{Y}' \\ \mathrm{Z}' \\ 1 \end{bmatrix} = \begin{bmatrix} \mathrm{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \mathbf{X} = \begin{bmatrix} \mathbf{r_1} & \mathbf{r_2} & \mathbf{r_3} & \mathbf{t} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \\ 1 \end{bmatrix},$$

By choosing the same origin for the local world coordinate frame and the global world coordinate frame (ENU as mentioned in Section 2.2.1), $\mathbf{t} = 0$. For the rotation matrix, we give the transformation for three axes $\mathbf{X_x} = [c, 0, 0]^{\mathrm{T}}$, $\mathbf{X_y} = [0, b, 0]^{\mathrm{T}}$ and $\mathbf{X_z} = [0, 0, h]^{\mathrm{T}}$ and obtain

$$\begin{aligned} \mathbf{X}'(\mathbf{X_x}) &= c\mathbf{r_1}, \\ \mathbf{X}'(\mathbf{X_y}) &= b\mathbf{r_2}, \\ \mathbf{X}'(\mathbf{X_z}) &= h\mathbf{r_3}. \end{aligned} \tag{7.18}$$

Then the three column vector of R can be computed as

$$\begin{aligned} \mathbf{r_1} &= \frac{\mathbf{X}'(\mathbf{X_x})}{c}, \\ \mathbf{r_2} &= \frac{\mathbf{X}'(\mathbf{X_y})}{b}, \\ \mathbf{r_3} &= \frac{\mathbf{X}'(\mathbf{X_z})}{h}. \end{aligned} \tag{7.19}$$

We choose three points respectively located on three axes of the local world coordinate frame, and then measure their GPS coordinates using Google Earth. The transformation between the GPS coordinates and the ENU coordinates is performed with the functions in Matlab (enu2geodetic and geodetic2enu).

Bibliography

[1] (n.d.a). Gyro sensors. https://www5.epsondevice.com/en/information/technical_info/gyro. Online; accessed 27-January-2021. xvii, 18

[2] (n.d.b). Gyroscope. https://learn.sparkfun.com/tutorials/gyroscope/all. Online; accessed 27-January-2021. 18

[3] (n.d.). HallEffect Sensor. https://www.electronics-tutorials.ws/electromagnetism/hall-effect.html. Online; accessed 27-January-2021. xvii, 20

[4] Aamir, H. (2019). Techspot. https://www.techspot.com/news/83061-report-finds-us-has-largest-number-surveillance-cameras.html/. [Online; accessed 27-January-2021]. x, xvii

[5] Ahn, H.-S. and Ko, K. H. (2009). Simple pedestrian localization algorithms based on distributed wireless sensor networks. *IEEE Transactions on Industrial Electronics*, 56(10):4296–4302. 91

[6] Aldea, E., Pollok, T., and Qu, C. (2019). Constraining relative camera pose estimation with pedestrian detector-based correspondence filters. In *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–7. IEEE. 25

[7] Almeida, L. B. (1997). C1. 2 multilayer perceptrons. *Handbook of Neural Computation C*, 1. xix, 104

[8] André, C., Le Hégarat-Mascle, S., and Reynaud, R. (2015). Evidential framework for data fusion in a multi-sensor surveillance system. *Engineering Applications of Artificial Intelligence*, 43:166–180. 58, 60, 66, 74, 78

[9] Avidan, S. (2007). Ensemble tracking. *IEEE transactions on pattern analysis and machine intelligence*, 29(2):261–271. 93

[10] Ayoun, A. and Smets, P. (2001). Data association in multi-target detection using the transferable belief model. *International journal of intelligent systems*, 16(10):1167–1182. 115

[11] Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359. 10

[12] Berclaz, J., Fleuret, F., Turetken, E., and Fua, P. (2011). Multiple object tracking using k-shortest paths optimization. *IEEE transactions on pattern analysis and machine intelligence*, 33(9):1806–1819. 93

[13] Betke, M. and Wu, Z. (2016). Data association for multi-object visual tracking. *Synthesis Lectures on Computer Vision*, 6(2):1–120. 93

[14] Bian, J., Lin, W.-Y., Matsushita, Y., Yeung, S.-K., Nguyen, T.-D., and Cheng, M.-M. (2017). Gms: Grid-based motion statistics for fast, ultra-robust feature correspondence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4181–4190. 25, 33

[15] Black, J. and Ellis, T. (2006). Multi camera image tracking. *Image and Vision Computing*, 24(11):1256–1267. 93

[16] Boufama, B. and Mohr, R. (1995). Epipole and fundamental matrix estimation using virtual parallax. In *Proceedings of IEEE International Conference on Computer Vision*, pages 1030–1036. IEEE. 39

[17] Bouma, H., Borsboom, S., den Hollander, R. J., Landsmeer, S. H., and Worring, M. (2012). Re-identification of persons in multi-camera surveillance under varying viewpoints and illumination. In *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense XI*, volume 8359, page 83590Q. International Society for Optics and Photonics. 93

[18] Bourgeois, F. and Lassalle, J.-C. (1971). An extension of the munkres algorithm for the assignment problem to rectangular matrices. *Communications of the ACM*, 14(12):802–804. 94

[19] Brachmann, E., Krull, A., Nowozin, S., Shotton, J., Michel, F., Gumhold, S., and Rother, C. (2017). Dsac-differentiable ransac for camera localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6684–6692. 11, 24

[20] Brachmann, E. and Rother, C. (2018). Learning less is more-6d camera localization via 3d surface regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4654–4662. 24

[21] Braga, A., Coldren Jr, J. R., Sousa, W., Rodriguez, D., and Alper, O. (2017). The benefits of body-worn cameras: new findings from a randomized controlled trial at the las vegas metropolitan police. *Arlington, VA: CNA*. ix

[22] Brockers, R., Susca, S., Zhu, D., and Matthies, L. (2012). Fully self-contained vision-aided navigation and landing of a micro air vehicle independent from external sensor inputs. In *Unmanned Systems Technology XIV*, volume 8387, page 83870Q. International Society for Optics and Photonics. 24

[23] Cai, Z., Hu, S., Shi, Y., Wang, Q., and Zhang, D. (2017). Multiple human tracking based on distributed collaborative cameras. *Multimedia Tools and Applications*, 76(2):1941–1957. 93

[24] Charco, J. L., Vintimilla, B. X., and Sappa, A. D. (2018). Deep learning based camera pose estimation in multi-view environment. In *2018 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pages 224–228. IEEE. 24

[25] Chavdarova, T., Baqué, P., Bouquet, S., Maksai, A., Jose, C., Bagautdinov, T., Lettry, L., Fua, P., Van Gool, L., and Fleuret, F. (2018). Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5030–5039. 15

[26] Chen, H., Aldea, E., and Le Hégarat-Mascle, S. (2019). Determining epipole location integrity by multimodal sampling. In *Proceedings of the 16th IEEE International Conference on AVSS, The 3th International Workshop on Traffic and Street Surveillance for Safety and Security (IWT4S)*. xix, 75, 87, 88

[27] Chen, Z., Pears, N. E., McDermid, J., and Heseltine, T. (2003). Epipole estimation under pure camera translation. In *DICTA*, volume 3, pages 849–858. 39

[28] Chen, Z., Wu, C., Shen, P., Liu, Y., and Quan, L. (2000). A robust algorithm to estimate the fundamental matrix. *Pattern Recognition Letters*, 21(9):851–861. 39

[29] Choy, C. B., Gwak, J., Savarese, S., and Chandraker, M. (2016). Universal correspondence network. In *Advances in Neural Information Processing Systems*, pages 2414–2422. 24

[30] Chum, O. and Matas, J. (2005). Matching with prosac-progressive sample consensus. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 220–226. IEEE. 11, 24

[31] Cobb, B. R. and Shenoy, P. P. (2006). On the plausibility transformation method for translating belief function models to probability models. *International journal of approximate reasoning*, 41(3):314–330. 60

[32] Csurka, G., Zeller, C., Zhang, Z., and Faugeras, O. D. (1997). Characterizing the uncertainty of the fundamental matrix. *CVIU*, 68(1):18–36. 40

[33] Cuturi, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport. In *NIPS*. 51

[34] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. IEEE. 93

[35] Davidson, I. (2004). An ensemble technique for stable learners with performance bounds. In *AAAI*, volume 2004, pages 330–335. 42

[36] De Vries, H., Strub, F., Mary, J., Larochelle, H., Pietquin, O., and Courville, A. C. (2017). Modulating early visual processing by language. In *Advances in Neural Information Processing Systems*, pages 6594–6604. 94

[37] Denœux, T. (2001). Inner and outer approximation of belief structures using a hierarchical clustering approach. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(04):437–460. 58

[38] Denœux, T. (2016). 40 years of dempster-shafer theory. *International Journal of Approximate Reasoning*, 79:1–6. 66

[39] Denoeux, T., El Zoghby, N., Cherfaoui, V., and Jouglet, A. (2014a). Optimal object association in the dempster–shafer framework. *IEEE transactions on cybernetics*, 44(12):2521–2531. 115

[40] Denoeux, T., El Zoghby, N., Cherfaoui, V., and Jouglet, A. (2014b). Optimal object association in the Dempster-Shafer framework. *IEEE Trans. on Cybernetics*, 44(22):2521–2531. 94

[41] Drevelle, V. and Bonnifait, P. (2011). A set-membership approach for high integrity height-aided satellite positioning. *GPS solutions*, 15(4):357–368. 66

[42] Dubois, D. and Prade, H. (1988). Representation and combination of uncertainty with belief functions and possibility measures. *Computational intelligence*, 4(3):244–264. 66

[43] Dubois, D. and Prade, H. (2000). Possibility theory in information fusion. In *Proceedings of the third international conference on information fusion*, volume 1, pages PS6–P19. IEEE. 94

[44] Efron, B. and Tibshirani, R. J. (1994). *An introduction to the bootstrap*. CRC press. 42

[45] En, S., Lechervy, A., and Jurie, F. (2018). Rpnet: An end-to-end network for relative camera pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0. 25

[46] Eshel, R. and Moses, Y. (2008). Homography based multiple camera detection and tracking of people in a dense crowd. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE. 93

[47] Fahad, A., Alshatri, N., Tari, Z., Alamri, A., Khalil, I., Zomaya, A. Y., Foufou, S., and Bouras, A. (2014). A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *IEEE transactions on emerging topics in computing*, 2(3):267–279. xviii, 68, 69

[48] Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395. 10, 66

[49] Gao, X.-S., Hou, X.-R., Tang, J., and Cheng, H.-F. (2003). Complete solution classification for the perspective-three-point problem. *TPAMI*, 25(8):930–943. 15, 117

[50] Giffard-Roisin, S., Yang, M., Charpiat, G., Kumler Bonfanti, C., Kégl, B., and Monteleoni, C. (2020). Tropical cyclone track forecasting using fused deep learning from aligned reanalysis data. *Frontiers in Big Data*, 3:1. 105

[51] Gogate, M., Adeel, A., and Hussain, A. (2017). Deep learning driven multimodal fusion for automated deception detection. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–6. IEEE. 94

[52] Goldman, Y., Rivlin, E., and Shimshoni, I. (2017). Robust epipolar geometry estimation using noisy pose priors. *Image and Vision Computing*, 67:16–28. xvii, 13, 14, 16, 24, 25, 26, 27, 28, 31, 32

[53] government, U. (2021). Gps space segment. https://www.gps.gov/systems/gps/space/. [Online; accessed 27-January-2021]. 17

[54] Grant, J. M. and Flynn, P. J. (2017). Crowd scene understanding from video: a survey. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 13(2):1–23. ix

[55] Gretton, A., Fukumizu, K., Harchaoui, Z., and Sriperumbudur, B. K. (2009). A fast, consistent kernel two-sample test. In *NIPS*, volume 23, pages 673–681. 42

[56] Gujarati, P. (2013). What is Accelerometer and how does it work on smartphones. https://www.techulator.com/resources/8930-How-does-smart-phone-accelerometer-work.aspx. Online; accessed 27-January-2021. xvii, 19

[57] Hachour, S., Delmotte, F., Mercier, D., and Lefèvre, E. (2014). Object tracking and credal classification with kinematic data in a multi-target context. *Information Fusion*, 20:174–188. 115

[58] Hall, E. H. et al. (1879). On a new action of the magnet on electric currents. *American Journal of Mathematics*, 2(3):287–292. 19

[59] Harris, C. G., Stephens, M., et al. (1988). A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Citeseer. 8

[60] Hartley, R. and Zisserman, A. (2003). *Multiple view geometry in computer vision*. Cambridge university press. 3, 31, 45, 47, 98, 105

[61] Hartley, R. I. (1993). Cheirality invariants. In *Proc. DARPA Image Understanding Workshop*, pages 745–753. 98

[62] Hartley, R. I. (1997). In defense of the eight-point algorithm. *IEEE Transactions on pattern analysis and machine intelligence*, 19(6):580–593. 7

[63] Hasnat, A., Bohne, J., Milgram, J., Gentric, S., and Chen, L. (2017). Deepvisage: Making face recognition simple yet with powerful generalization skills. In *ICCV Workshops*. 97, 102, 103

[64] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778. 28

[65] Hermans, A., Beyer, L., and Leibe, B. (2017). In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*. 97

[66] Hirzer, M., Roth, P. M., Köstinger, M., and Bischof, H. (2012). Relaxed pairwise learned metric for person re-identification. In *European conference on computer vision*, pages 780–793. Springer. 93

[67] Hu, W., Xie, N., Hu, R., Ling, H., Chen, Q., Yan, S., and Maybank, S. (2014). Bin ratio-based histogram distances and their application to image classification. *TPAMI*, 36(12):2338–2352. 96, 102, 103

[68] Ilg, E., Cicek, O., Galesso, S., Klein, A., Makansi, O., Hutter, F., and Brox, T. (2018). Uncertainty estimates and multi-hypotheses networks for optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 652–667. 42

[69] Jain, A., Nandakumar, K., and Ross, A. (2005). Score normalization in multimodal biometric systems. *Pattern recognition*, 38(12):2270–2285. 100

[70] Jansen, A. M., Giebels, E., van Rompay, T. J., and Junger, M. (2018). The influence of the presentation of camera surveillance on cheating and pro-social behavior. *Frontiers in psychology*, 9:1937. ix

[71] Jaulin, L., Kieffer, M., Didrit, O., and Walter, E. (2001). Interval analysis. In *Applied interval analysis*, pages 11–43. Springer. 66

[72] Johnson, A. (2014). Clipper-an open source freeware library for clipping and offsetting lines and polygons. *Retrieved September*. 60

[73] Jonker, R. and Volgenant, A. (1987). A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38(4):325–340. 94

[74] Jousselme, A.-L. and Maupin, P. (2012). Distances in evidence theory: Comprehensive survey and generalizations. *International Journal of Approximate Reasoning*, 53(2):118–145. 57

[75] Junior, J. C. S. J., Musse, S. R., and Jung, C. R. (2010). Crowd analysis using computer vision techniques. *IEEE Signal Processing Magazine*, 27(5):66–77. ix

[76] Kasten, Y. and Werman, M. (2018). Two view constraints on the epipoles from few correspondences. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 888–892. IEEE. 39

[77] Kelly, P. H., Katkere, A., Kuramura, D. Y., Moezzi, S., and Chatterjee, S. (1995). An architecture for multiple perspective interactive video. In *Proceedings of the third ACM international conference on Multimedia*, pages 201–212. 93

[78] Khan, S. M. and Shah, M. (2006). A multiview approach to tracking people in crowded scenes using a planar homography constraint. In *European Conference on Computer Vision*, pages 133–146. Springer. 93

[79] Kim, C., Li, F., Ciptadi, A., and Rehg, J. M. (2015). Multiple hypothesis tracking revisited. In *Proceedings of the IEEE international conference on computer vision*, pages 4696–4704. 93

[80] Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97. 93, 94

[81] Kybic, J. (2009). Bootstrap resampling for image registration uncertainty estimation without ground truth. *IEEE Transactions on Image Processing*, 19(1):64–73. 42

[82] Kybic, J. and Nieuwenhuis, C. (2011). Bootstrap optical flow confidence and uncertainty measure. *Computer Vision and Image Understanding*, 115(10):1449–1462. 42

[83] Lawn, J. and Cipolla, R. (1996). Reliable extraction of the camera motion using constraints on the epipole. In *European Conference on Computer Vision*, pages 161–173. Springer. 68

[84] Lawn, J. M. and Cipolla, R. (1994). Robust egomotion estimation from affine motion parallax. In *European Conference on Computer Vision*, pages 205–210. Springer. 40

[85] Leal-Taixé, L., Canton-Ferrer, C., and Schindler, K. (2016). Learning by tracking: Siamese cnn for robust target association. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 33–40. 93, 94

[86] Leal-Taixe, L., Pons-Moll, G., and Rosenhahn, B. (2012). Branch-and-price global optimization for multi-view multi-target tracking. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1987–1994. IEEE. 93

[87] Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R., and Furgale, P. (2015). Keyframe-based visual–inertial odometry using nonlinear optimization. *The International Journal of Robotics Research*, 34(3):314–334. 25

[88] Li, C., Wang, X., and Liu, W. (2017). Neural features for pedestrian detection. *Neurocomputing*, 238:420–432. 81

[89] Li, J., Xiang, X., Dai, T., and Xia, S.-T. (2019). Making large ensemble of convolutional neural networks via bootstrap re-sampling. In *2019 IEEE Visual Communications and Image Processing (VCIP)*, pages 1–4. IEEE. 42

[90] Li, Y., Huang, C., and Nevatia, R. (2009). Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2953–2960. IEEE. 94

[91] Liang, M., Yang, B., Wang, S., and Urtasun, R. (2018). Deep continuous fusion for multi-sensor 3d object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 641–656. 94

[92] Lim, J. and Barnes, N. (2010). Estimation of the epipole using optical flow at antipodal points. *Computer Vision and Image Understanding*, 114(2):245–253. 39

[93] Lloyd, S. (1982). Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137. 69

[94] Longuet-Higgins, H. C. and Prazdny, K. (1980). The interpretation of a moving retinal image. *Proc. of the Royal Soc. of London. Series B. Biological Sciences*, 208(1173):385–397. 40

[95] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110. xvii, 8, 9, 28, 50

[96] Luo, H., Gu, Y., Liao, X., Lai, S., and Jiang, W. (2019). Bag of tricks and a strong baseline for deep person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0. 97

[97] Luong, Q.-T. and Faugeras, O. D. (1998). On the determination of epipoles using cross-ratios. *CVIU*, 71:1–18. 39

[98] Maalel, W., Zhou, K., Martin, A., and Elouedi, Z. (2014). Belief hierarchical clustering. In *International Conference on Belief Functions*, pages 68–76. Springer. 72

[99] Madhulatha, T. S. (2012). An overview on clustering methods. *arXiv preprint arXiv:1205.1117*. 68

[100] Maronna, R., Martin, D., and Yohai, V. (2006). *Robust Statistics: Theory and Methods*. Wiley Series in Probability and Statistics. Wiley. 42

[101] McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133. 103

[102] Melekhov, I., Ylioinas, J., Kannala, J., and Rahtu, E. (2017). Relative camera pose estimation using convolutional neural networks. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 675–687. Springer. 24

[103] Mercier, D., Lefevre, É., and Jolly, D. (2011). Object association with belief functions, an application with vehicles. *Information Sciences*, 181(24):5485–5500. 115

[104] Moo Yi, K., Trulls, E., Ono, Y., Lepetit, V., Salzmann, M., and Fua, P. (2018). Learning to find good correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2666–2674. xvii, 14, 24, 25, 28, 29, 30, 31, 32, 33

[105] Moulon, P., Monasse, P., and Marlet, R. (2013). Global fusion of relative motions for robust, accurate and scalable structure from motion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3248–3255. 14

[106] Munkres, J. (1957). Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38. 94

[107] Neverova, N., Wolf, C., Taylor, G., and Nebout, F. (2015). Moddrop: adaptive multi-modal gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(8):1692–1706. 94, 105

[108] Nguyen, D.-K. and Okatani, T. (2018). Improved fusion of visual and language representations by dense symmetric co-attention for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6087–6096. 94

[109] Nistér, D. (2004). An efficient solution to the five-point relative pose problem. *IEEE transactions on pattern analysis and machine intelligence*, 26(6):756–770. 7

[110] Nixon, J., Lakshminarayanan, B., and Tran, D. (2020). Why are bootstrapped deep ensembles not better? In *"I Can't Believe It's Not Better!"NeurIPS 2020 workshop*. 42

[111] OpenVINO (n.d.). person-reidentification-retail-0265. https://docs.openvinotoolkit.org/2020.4/omz_models_intel_person_reidentification_retail_0265_description_person_reidentification_retail_0265.html . Online; accessed 19-April-2021. 107

[112] Orbach, J. (1962). Principles of neurodynamics. perceptrons and the theory of brain mechanisms. *Archives of General Psychiatry*, 7(3):218–219. 103

[113] Osband, I., Aslanides, J., and Cassirer, A. (2018). Randomized prior functions for deep reinforcement learning. *arXiv preprint arXiv:1806.03335*. 42

[114] Osband, I., Blundell, C., Pritzel, A., and Van Roy, B. (2016). Deep exploration via bootstrapped dqn. *arXiv preprint arXiv:1602.04621*. 42

[115] Papadopoulo, T. and Lourakis, M. I. (2000). Estimating the jacobian of the singular value decomposition: Theory and applications. In *ECCV*, pages 554–570. Springer. 38, 40, 43, 45, 47, 67

[116] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830. 77

[117] Pellegrini, S., Ess, A., Schindler, K., and Van Gool, L. (2009). You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision*, pages 261–268. IEEE. 93

[118] Pellicanò, N., Aldea, E., and Hégarat-Mascle, S. L. (2016a). Robust wide baseline pose estimation from video. In *ICPR, Cancún, Mexico, December 4-8, 2016*, pages 3820–3825. xviii, 63

[119] Pellicanò, N., Aldea, E., and Le Hégarat-Mascle, S. (2016b). Robust wide baseline pose estimation from video. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 3820–3825. IEEE. 25

[120] Pellicanò, N., Aldea, E., and Le Hégarat-Mascle, S. (2017). Geometry-based multiple camera head detection in dense crowds. In *BMVC - 5th Activity Monitoring by Multiple Distributed Sensing Workshop*. 93

[121] Pellicanò, N., Le Hégarat-Mascle, S., and Aldea, E. (2018a). 2cobel: A scalable belief function representation for 2d discernment frames. *International Journal of Approximate Reasoning*, 103:320–342. xviii, 60, 61, 62, 66, 75

[122] Pellicanò, N., Le Hégarat-Mascle, S., and Aldea, E. (2018b). 2cobel: A scalable belief function representation for 2d discernment frames. *International Journal of Approximate Reasoning*, 103:320 – 342. 93

[123] Pichon, F., Destercke, S., and Burger, T. (2014). A consistency-specificity trade-off to select source behavior in information fusion. *IEEE transactions on cybernetics*, 45(4):598–609. 66

[124] Pollefeys, M., Nistér, D., Frahm, J.-M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S.-J., Merrell, P., et al. (2008). Detailed real-time urban 3d reconstruction from video. *International Journal of Computer Vision*, 78(2-3):143–167. 24

[125] Pollok, T. and Monari, E. (2016). A visual slam-based approach for calibration of distributed camera networks. In *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 429–437. IEEE. 24

[126] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press. 42, 47

[127] R, A. (2020). Resampling Methods-A Simple Introduction to The Bootstrap Method. https://arifromadhan19.medium.com/resampling-methods-a-simple-introduction-to-the-bootstrap-method-3a36d076852f. xviii, 43

[128] Raguram, R., Frahm, J.-M., and Pollefeys, M. (2009). Exploiting uncertainty in random sample consensus. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2074–2081. IEEE. 68

[129] Ranftl, R. and Koltun, V. (2018). Deep fundamental matrix estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 284–299. 25

[130] RDUSSEEUN, L. and KAUFMAN, P. (1987). Clustering by means of medoids. 69

[131] Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*. 84, 101, 102, 103

[132] Rekik, W., Le Hégarat-Mascle, S., Reynaud, R., Kallel, A., and Hamida, A. B. (2016). Dynamic object construction using belief function theory. *Information Sciences*, 345:129–142. 60, 66, 115

[133] Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee. 10

[134] Rupnik, E., Daakir, M., and Deseilligny, M. P. (2017). Micmac–a free, open-source solution for photogrammetry. *Open Geospatial Data, Software and Standards*, 2(1):1–9. 14

[135] Saxena, S., Brémond, F., Thonnat, M., and Ma, R. (2008). Crowd behavior recognition for video surveillance. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 970–981. Springer. ix

[136] Schmandt, M. (n.d.). Gis commons. `https://giscommons.org/chapter-2-input/`. [Online; accessed 27-January-2021]. 17

[137] Schonberger, J. L. and Frahm, J.-M. (2016). Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4104–4113. xvii, 14

[138] Schönberger, J. L. and Frahm, J.-M. (2016). Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 14

[139] Schubert, J. (2008). Clustering decomposed belief functions using generalized weights of conflict. *International Journal of Approximate Reasonning*, 48(2):466–480. 67

[140] Shafer, G. (1976). *A mathematical theory of evidence*, volume 42. Princeton university press. 58, 66

[141] Smets, P. (1990). The combination of evidence in the transferable belief model. *IEEE Transactions on pattern analysis and machine intelligence*, 12(5):447–458. 58, 66, 74

[142] Smets, P. (1993). Belief functions: the disjunctive rule of combination and the generalized bayesian theorem. *International Journal of approximate reasoning*, 9(1):1–35. 58

[143] Smets, P. and Kennes, R. (1994). The transferable belief model. *Artificial intelligence*, 66(2):191–234. 59

[144] Steck, H. and Jaakkola, T. S. (2003). Bias-corrected bootstrap and model uncertainty. In *NIPS*, pages 521–528. Citeseer. 42

[145] Stine, R. (1989). An introduction to bootstrap methods: Examples and ideas. *Sociological Methods & Research*, 18(2-3):243–291. 42

[146] Sun, Y., Zheng, L., Deng, W., and Wang, S. (2017). Svdnet for pedestrian retrieval. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3800–3808. 97

[147] Sundlie, P. O., Taylor, C. N., and Fernando, J. A. (2015). Sources of uncertainty in feature-based image registration algorithms. In *Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR VI*, volume 9464, page 94640Z. Int. Soc. for Optics and Photonics. 38

[148] Sur, F., Noury, N., and Berger, M.-O. (2008). Computing the uncertainty of the 8 point algorithm for fundamental matrix estimation. In *BMVC 2008*, page 10. 40, 44

[149] Tomè, D., Monti, F., Baroffio, L., Bondi, L., Tagliasacchi, M., and Tubaro, S. (2016). Deep convolutional neural networks for pedestrian detection. *Signal processing: image communication*, 47:482–489. 81

[150] Torr, P. H. and Zisserman, A. (2000). Mlesac: A new robust estimator with application to estimating image geometry. *Computer vision and image understanding*, 78(1):138–156. 11, 24

[151] Tzutalin (2015). LabelImg. https://github.com/tzutalin/labelImg. 15

[152] Vandoni, J. (2019). *Ensemble Methods for Pedestrian Detection in Dense Crowds*. PhD thesis. 93

[153] Vatti, B. R. (1992). A generic solution to polygon clipping. *Communications of the ACM*, 35(7):56–63. 60

[154] Vetterling, W. T., Teukolsky, S. A., Press, W. H., and Flannery, B. P. (1989). *Numerical recipes*. University Press. 40

[155] Villani, C. (2008). *Optimal transport: old and new*, volume 338. Springer Science & Business Media. 51

[156] Wang, S., Zhang, J., and Zong, C. (2018). Learning multimodal word representation via dynamic fusion methods. In *Thirty-Second AAAI Conference on Artificial Intelligence*. 94

[157] Welsh, B. C. and Farrington, D. P. (2009). Public area cctv and crime prevention: an updated systematic review and meta-analysis. *Justice Quarterly*, 26(4):716–745. ix

[158] Wen, L., Lei, Z., Chang, M.-C., Qi, H., and Lyu, S. (2017). Multi-camera multi-target tracking with space-time-view hyper-graph. *International Journal of Computer Vision*, 122(2):313–333. 93

[159] Wikipedia (2021). Gyroscope. https://en.wikipedia.org/wiki/Gyroscope. Online; accessed 27-January-2021. 17, 18

[160] Wong, D., Hayes, M., and Bainbridge-Smith, A. (2010). Imu-aided surf feature matching for relative pose estimation. In *2010 25th International Conference of Image and Vision Computing New Zealand*, pages 1–6. IEEE. 24

[161] Wu, C. (2013). Towards linear-time incremental structure from motion. In *2013 International Conference on 3D Vision-3DV 2013*, pages 127–134. IEEE. 14

[162] Wu, Z., Thangali, A., Sclaroff, S., and Betke, M. (2012). Coupling detection and data association for multiple object tracking. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1948–1955. IEEE. 93

[163] Xiang, Y., Alahi, A., and Savarese, S. (2015). Learning to track: Online multi-object tracking by decision making. In *Proceedings of the IEEE international conference on computer vision*, pages 4705–4713. 94

[164] Xu, Y., Osep, A., Ban, Y., Horaud, R., Leal-Taixé, L., and Alameda-Pineda, X. (2020). How to train your deep multi-object tracker. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6787–6796. 115

[165] Yager, R. R. (1987). On the dempster-shafer framework and new combination rules. *Information sciences*, 41(2):93–137. 66

[166] Yi, K. M., Trulls, E., Lepetit, V., and Fua, P. (2016). Lift: Learned invariant feature transform. In *European Conference on Computer Vision*, pages 467–483. Springer. 10, 24

[167] Yi, K. M., Trulls, E., Ono, Y., Lepetit, V., Salzmann, M., and Fua, P. (2018). Learning to find good correspondences. *CVPR*, pages 2666–2674. 15, 50, 75

[168] Yoon, J. H., Lee, C.-R., Yang, M.-H., and Yoon, K.-J. (2019). Structural constraint data association for online multi-object tracking. *International Journal of Computer Vision*, 127(1):1–21. 93

[169] Yu, T., Li, D., Yang, Y., Hospedales, T. M., and Xiang, T. (2019). Robust person re-identification by modelling feature uncertainty. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 552–561. 114

[170] Yugendar, P. and Ravishankar, K. (2018). Crowd behavioural analysis at a mass gathering event. *Journal of KONBiN*, 46(1):5–20. ix

[171] Zagoruyko, S. and Komodakis, N. (2015). Learning to compare image patches via convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4353–4361. 25

[172] Zair, S. and Le Hégarat-Mascle, S. (2017). Evidential framework for robust localization using raw gnss data. *Engineering Applications of Artificial Intelligence*, 61:126–135. 66, 82

[173] Zaman, K. and Saha, S. K. (2018). An efficient methodology for robust assignment problem. *International Journal of Operational Research*, 33(2):239–255. 114

[174] Zhang, J., Sun, D., Luo, Z., Yao, A., Zhou, L., Shen, T., Chen, Y., Quan, L., and Liao, H. (2019). Learning two-view correspondences and geometry using order-aware network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5845–5854. 25

[175] Zhang, S., Zhu, Y., and Roy-Chowdhury, A. (2015). Tracking multiple interacting targets in a camera network. *Computer Vision and Image Understanding*, 134:64–73. 93

[176] Zhang, Z. (1997). Parameter estimation techniques: A tutorial with application to conic fitting. *Image and vision Computing*, 15(1):59–76. xxi, 11

[177] Zhang, Z. (1998). Determining the epipolar geometry and its uncertainty: A review. *IJCV*, 27(2):161–195. 40, 45, 47

[178] Zhao, Y., Jia, R., and Shi, P. (2016). A novel combination method for conflicting evidence based on inconsistent measurements. *Information Sciences*, 367:125–142. 66

[179] Zhou, K., Martin, A., and Pan, Q. (2018). *A belief combination rule for a large number of sources*. Infinite Study. 67, 68

[180] Zoller, T. and Buhmann, J. M. (2007). Robust image segmentation using resampling and shape constraints. *IEEE transactions on pattern analysis and machine intelligence*, 29(7):1147–1164. 42

[181] Zou, T., Yang, S., Zhang, Y., and Ye, M. (2020). Attention guided neural network models for occluded pedestrian detection. *Pattern Recognition Letters*, 131:91–97. 81

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Alignement de vues pour la localisation collaborative dans des applications de sécurité

Mots clés : système de multi-camera, fusion de données, fonctions de croyance, association de données

Résumé : Cette thèse s'intéresse à la localisation collaborative à partir d'une caméra mobile et d'une caméra statique pour des applications de vidéo-surveillance. Pour la surveillance d'évènements sensibles, la sécurité civile recours de plus en plus à des réseaux de caméras collaboratives incluant des caméras dynamiques et des caméras de surveillance traditionnelles, statiques. Il s'agit, dans des scènes de foules, de localiser d'une part le porteur de la caméra (typiquement agent de sécurité) mais également des évènements observés dans les images, afin de guider les secours par exemple. Cependant, les différences de point de vue entre la caméra mobile située au niveau du sol, et la caméra de vidéo-surveillance située en hauteur, couplées à la présence de motifs répétitifs et d'occlusions rendent les tâches de calibration et de localisation ardue. Nous nous sommes d'abord intéressés à la façon dont des informations issues de capteurs de localisation et d'orientation (GPS-IMU) bas-coût, pouvaient contribuer à raffiner l'estimation de la pose relative entre les caméras. Nous avons ensuite proposé de localiser la caméra mobile par la localisation de son épipole dans l'image de la caméra statique. Pour rendre robuste cette estimation vis-à-vis de la présence d'outliers en termes d'appariements de points clés, nous avons développé deux algorithmes. Le premier est basé sur une approche cumulative pour construire la carte d'incertitude de l'épipole. Le second, qui exploite de cadre de la théorie des fonctions de croyance et de son extension aux cadres de discernement 2D, nous a permis de proposer une contribution à la gestion d'un grand nombre de sources élémentaires, dont certaines incompatibles, basée sur un clustering des fonctions de croyances, particulièrement intéressant en vue de la combinaison avec d'autres sources (comme les détecteurs de piétons et/ou données GPS pour notre application). Enfin, la dernière partie concernant la géolocalisation des individus dans la scène, nous a conduit à étudier le problème de l'association de données entre les vues. Nous avons proposé d'utiliser des descripteurs et contraintes géométriques, en plus des descripteurs d'apparence classiques, dans la fonction de coût d'association. Nous avons montré la pertinence de ces informations géométriques qu'elles soient explicites, ou apprises à l'aide un réseau de neurones.

Title : Registration of egocentric views for collaborative localization in security applications

Keywords : multi-camera network, data fusion, belief functions, data association

Abstract : This work focuses on collaborative localization between a mobile camera and a static camera for video surveillance. In crowd scenes and sensitive events, surveillance involves locating the wearer of the camera (typically a security officer) and also the events observed in the images (e.g., to guide emergency services). However, the different points of view between the mobile camera (at ground level), and the video surveillance camera (located high up), along with repetitive patterns and occlusions make difficult the tasks of calibration and localization. We first studied how low-cost positioning and orientation sensors (GPS-IMU) could help refining the estimate of relative pose between cameras. We then proposed to locate the mobile camera using its epipole in the image of the static camera. To make this estimate robust with respect to outlier keypoint matches, we developed two algorithms: either based on a cumulative approach to derive an uncertainty map, or exploiting the belief function framework. Facing with the issue of a large number of elementary sources, some of which are incompatible, we provide a solution based on a belief clustering, in the perspective of further combination with other sources (such as pedestrian detectors and/or GPS data for our application). Finally, the individual location in the scene led us to the problem of data association between views. We proposed to use geometric descriptors/constraints, in addition to the usual appearance descriptors. We showed the relevance of this geometric information whether it is explicit, or learned using a neural network.