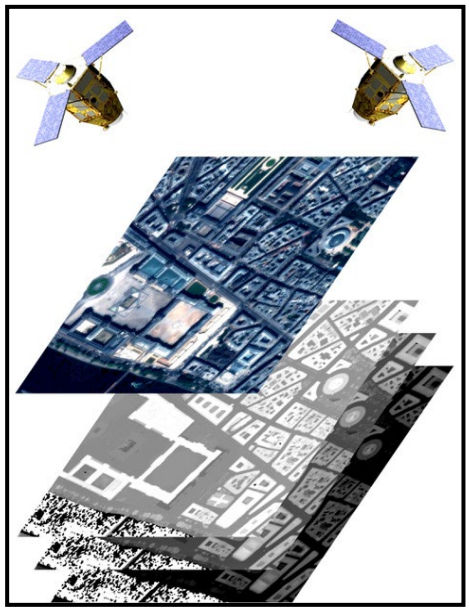
Uncertainty characterisation in stereophotogrammetry using satellite images

Roman Malinowski

Par **Roman MALINOWSKI**

Uncertainty characterisation in stereophotogrammetry using satellite images

Thèse présentée
pour l'obtention du grade
de Docteur de l'UTC



Soutenue le 19 décembre 2024
Spécialité : Informatique : Unité de recherche Heudyasic (UMR-7253)
D2842

Uncertainty characterisation in stereophotogrammetry using satellite images

Caractérisation d'incertitude pour la stéréophotogrammétrie à partir d'images satellites

Spécialité : Informatique

Laboratoire Heudiasyc

19/12/2024

Par **Roman MALINOWSKI**

A thesis presented under the supervision of:
Sébastien DESTERCKE

Reviewers:
Emmanuel Dubois, Luc Jaulin, Olivier Strauss, Emmanuelle Sarrazin

Centre National D'Etudes Spatiales
CS
Heudiasyc - Université de Technologie de Compiègne

cnes CENTRE NATIONAL D'ÉTUDES SPATIALES

a Sopra Steria company

ALLIANCE SORBONNE UNIVERSITÉ | utc

Région Hauts-de-France

Remerciements

Je tiens tout d'abord à remercier l'Université de technologie de Compiègne, le CNES et CS Group, qui ont su s'entendre pour me permettre de réaliser cette thèse dans de bonnes conditions. J'ai eu la chance de pouvoir travailler entre Compiègne et Toulouse, et d'y rencontrer de nombreuses personnes qui m'ont épaulé durant ces trois années particulièrement enrichissantes. Je tiens ensuite à remercier les membres du jury, notamment M. Luc Jaulin pour avoir suivi mes travaux de thèse depuis nos échanges lors de la journée Imagin. Je remercie également Mme Isabelle Bloch et M. Marc Pierrot Deseilligny pour avoir accepté de rapporter ce travail et pour le temps qu'ils y ont consacré ainsi que pour leurs remarques pertinentes. Je voudrais aussi remercier chaleureusement M. Olivier Strauss, qui a manifesté son intérêt pour mes travaux depuis les premiers instants, m'a permis de découvrir son laboratoire du LIRMM, et a piqué ma curiosité avec des problèmes de traitement d'image auxquels je réfléchis encore aujourd'hui. Un grand merci à M. Benjamin Quost, qui a accepté de présider ce jury, mais aussi de partager son bureau pour que s'y déroulent mes réunions de thèse lorsque j'étais un peu perdu.

Je voudrais remercier Etienne Berthier et Simon Gascoin, pour tout l'intérêt qu'ils ont porté à mes travaux. Mais aussi pour m'avoir généreusement partagé leurs données, et mis en contact avec Liss Marie Andreassen et Brian Menounos qui ont fait de même. C'était pour moi bien plus que de simples données. J'aimerais également remercier Yoann Steux, pour tout le temps passé à résoudre mes problèmes sur CARS.

Bien entendu, merci à Simon, Tatha, Tom, pour avoir égayé mes soirées compiègnoises. Cassandra, Gabriela, Lou, Marion, Alice, Martin, et surtout Camille, pour tout ce qu'on a vécu au CNES, mais aussi en dehors.

Je ne dis pas merci à :

- La personne qui m'a dit qu'installer CARS était facile
- Mon vélo, et surtout mes freins. Trop de temps et d'efforts passés à le réparer, et encore plus à me réparer moi-même

Un grand merci à ma soeur, mes parents et grands-parents, qui se sont intéressés à mes travaux et m'ont toujours souhaité le meilleur. Merci également à mon autre famille, chinoise, pour les déménagements et pour avoir pris soin de mes papilles gustatives. À toi, Virginie, pour m'avoir soutenu, écouté, t'être intéressée à mes travaux quand bien même ce que je racontais n'avait que peu d'intérêt. M'avoir changé les idées quand je voulais vivre autre chose que cette thèse, et avoir accepté que je n'aie parfois pas d'autre vie que celle-ci. J'espère pouvoir te rendre, un jour, tout ce que tu m'as donné.

Un grand merci à Manu, qui a fait la jonction entre ces deux mondes que sont la galaxie CNES et la communauté des probas imprécises. Sans toi, cette thèse n'aurait sûrement pas eu lieu, ou du moins pas sous cette forme. Un grand merci à Loïc, avec le regret de n'avoir pas passé encore plus de temps à travailler à tes côtés. Je t'admirerai toujours pour tes conseils et remarques qui touchent toujours juste. Mais également pour tes commentaires qui m'ont beaucoup fait rire lors de la rédaction de ce manuscrit. Sébastien, bien entendu, je te serai à jamais reconnaissant pour m'avoir guidé, conseillé et épaulé lorsque j'étais perdu dans des concepts théoriques qui me dépassaient. Pour avoir dit ce qui n'allait pas, mais aussi toujours ce qui allait bien. Merci pour m'avoir toujours accordé ton temps, même quand tu ne l'avais pas.

Enfin, un grand merci à Manue, qui m'a convaincu de me lancer dans cette aventure, et qui est restée jusqu'au bout — et même parfois très tard — pour enrichir mon travail de ses idées et de ses conseils. Tu m'as dit au début de ma thèse, que je la commencerai en étant sous ta supervision, et que je la terminerai en étant ton collègue. Plus qu'une collègue, j'ai terminé cette thèse avec un modèle, un guide, une amie. Tu es la flamme d'Udûn qui éclairait les recoins les plus sombres du cluster ou de l'écriture de manuscrit. Chanceux seront les prochains doctorants qui t'auront comme maître Jedi.

Contents

Acronyms

B/H Base-to-Height ratio.

CARS Chaîne Automatique de Restitution Stéréoscopique.

CDD Charge Coupled Device.

CDF Cumulative Distribution Function.

CNES Centre National d'Etudes Spatiales.

CO3D Constellation Optique 3D.

DEM Digital Elevation Models.

DSM Digital Surface Model.

DTM Digital Terrain Model.

EO Earth Observation.

GIS Geographic Information Systems.

IGN Institut National de l'Information Géographique et Forestière.

IP Imprecise Probabilities.

LiDAR Light Detection And Ranging.

NIR Near Infra Red.

PDF Probability Density Function.

Radar Radio Detection and Ranging.

RGB Red Green Blue.

RPC Rational Polynomial Models.

SAD Sum of Absolute Differences.

SGM Semi-Global Matching.

SIFT Scale-Invariant Feature Transform.

UTM Universal Transverse Mercator.

VHR Very High Resolution.

ZNCC Zero Normalized Crossed Correlation.

Résumé

Actuellement, les Modèles Numériques de Surface (MNS) sont nécessaires pour de nombreuses applications, telles que la gestion des ressources en eau, le suivi de la biomasse, l'évaluation des dommages causés par les catastrophes naturelles ou la planification urbaine. Les MNS peuvent principalement être produits par interférométrie Radar, photogrammétrie ou en utilisant des instruments LiDAR. Dans ce contexte, le CNES et Airbus préparent le lancement de la constellation de satellites CO3D afin d'assurer la production massive de MNS à haute résolution par photogrammétrie. Fournie avec le MNS, une carte de performance permettra de caractériser les erreurs liées aux incertitudes dans les données d'entrée ainsi qu'aux incertitudes des méthodes utilisées.

L'objectif de cette thèse et de caractériser l'incertitude associée à la production de MNS par photogrammétrie. Nous utilisons des modèles d'incertitude spécifiques, à savoir des probabilités imprécises, et plus particulièrement des distributions de possibilité, afin de caractériser l'incertitude résultant du traitement des images stéréo. Ces modèles définissent des "ensembles crédaux", qui sont des ensembles convexes de distributions de probabilité. L'intérêt de ces ensembles crédaux est d'être mieux adaptés pour représenter l'incertitude résultant de connaissances incomplètes ou imparfaites, par rapport aux simples distributions de probabilité. En présence de plusieurs sources d'incertitudes, il est également nécessaire de considérer leurs relations de dépendance. Pour cela, il est courant d'utiliser des copules, qui sont des modèles représentant la dépendance entre plusieurs variables aléatoires. Dans cette thèse, trois méthodes distinctes sont introduites afin de joindre des ensembles crédaux marginaux en des ensembles crédaux multivariés à l'aide de copules. Les relations entre ces méthodes sont ensuite étudiées pour des copules spécifiques ainsi que pour différents modèles de probabilités imprécises. Une application de ces ensembles crédaux multivariés est ensuite proposée, afin de propager l'incertitude d'images stéréo dans un problème d'appariement. Différentes optimisations et façons de faciliter la propagation de l'incertitude sont présentées. La propagation correcte de l'incertitude est enfin validée à l'aide de méthodes de Monte-Carlo.

Une seconde contribution de cette thèse concerne la modélisation de l'incertitude intrinsèque de l'algorithme d'appariement en utilisant des distributions de possibilité.

Une méthode est proposée pour générer des intervalles de confiance associés aux résultats de l'étape d'appariement, et ces intervalles sont propagés jusqu'à la fin du pipeline stéréo, produisant ainsi des intervalles de confiance d'élévation pour les MNS. La taille et la précision de ces intervalles est évaluée en utilisant des images satellites réelles et des MNS pour lesquels une vérité terrain est disponible. Les intervalles ainsi créés contiennent correctement la vérité terrain au moins 90 % du temps.

Abstract

Currently, Digital Surface Models (DSMs) are required in many applications, such as for managing water resources, monitoring biomass, evaluating damages caused by natural catastrophes, or for urban planning. DSMs can mainly be produced by Radar interferometry, photogrammetry or LiDAR scanning. In this context, CNES and Airbus are planning the launch of the CO3D constellation of satellites to massively provide highly accurate DSMs using photogrammetry. A performance map will also be provided alongside the DSM to characterize potential errors resulting from the uncertainty on input data or on its processing.

The objective of this thesis is to characterize the uncertainty associated with the production of DSMs using photogrammetry. To do so, special uncertainty models, namely imprecise probabilities, and more specifically possibility distributions, are employed to characterize the uncertainty arising from stereo images processing. Those models define credal sets, which are convex sets of probability distributions. Credal sets are well-suited to represent uncertainty resulting from incomplete or imperfect knowledge, which can be a limitation for a single probability distribution. In the presence of multiple sources of uncertainty, their dependency must also be considered. For this purpose, it is possible to consider copulas, which are models used to represent the dependency between multiple random variables. In this thesis, three different methods are introduced to join marginal credal sets into multivariate credal sets using copulas. The relationships between those methods are then investigated, for specific copulas and different models of imprecise probabilities. An application of those multivariate credal sets is then proposed, for propagating the uncertainty of stereo images in a dense stereo-matching problem. Different optimizations and ways to facilitate the uncertainty propagation are presented. The correct uncertainty propagation is validated using Monte Carlo sampling.

A second contribution of this thesis concerns the uncertainty modeling of the dense-matching algorithm itself using possibility distributions. A method is presented for generating confidence intervals associated with the results of the dense-matching step. Those intervals are then propagated to the end of the stereo pipeline, therefore producing elevation confidence intervals for the DSMs. The size and accuracy of intervals are then

evaluated, using real satellites images and DSMs for which a ground truth is available. Elevation intervals correctly contain the ground truth at least 90% of the time.

Foreword

Before delving into the subject of this manuscript, we would like to give some advice on how to efficiently navigate through it. When writing this thesis, we made extensive use of the *hyperref* package, so that reading it on a PDF viewer was made easier. You can thus click on citations, figure numbers, equation numbers, chapters and sections numbers, acronyms *etc.* to directly jump to the concerned part. When following a reference to a citation, a previous chapter, equations or figures located in a different part of the manuscript, it can be a arduous process to go back to the section you were reading. Depending on the OS of your computer and the app used to read the PDF document, there usually exist shortcuts to jump back to the previous view. This allows to quickly switch back and forth between chapters and sections.

For instance, imagine that you are in Chapter 5 and we make a reference to an equation from Chapter 2. If you do not recall the equation, and quickly want to see what it is about, simply click on the hyperlink to directly go to the relevant equation from Chapter 2. Then use your system's shortcut to go back to where you were in Chapter 5.

- Using Acrobat Reader: the shortcut Alt+← (left arrow key) on Windows or Linux brings you to the previous view after clicking on a hyperlink. Afterwards, you can alternate views with Alt+→ and Alt+←. On macOS, the Alt key is replaced by the ⌘ key.
- Using Preview on macOS, you can add the Page History button to the toolbar, by right-clicking on the toolbar and selecting Customize Toolbar
- Using Okular on Linux, Alt+⇧+← (left arrow key) brings you to the previous view after clicking on a hyperlink. Afterward, you can alternate views with Alt+⇧+→ and Alt+⇧+←

Hopefully, this makes the reading of this thesis a more pleasant experience.

You never talk of likelihoods on Arrakis.
You speak only of possibilities.

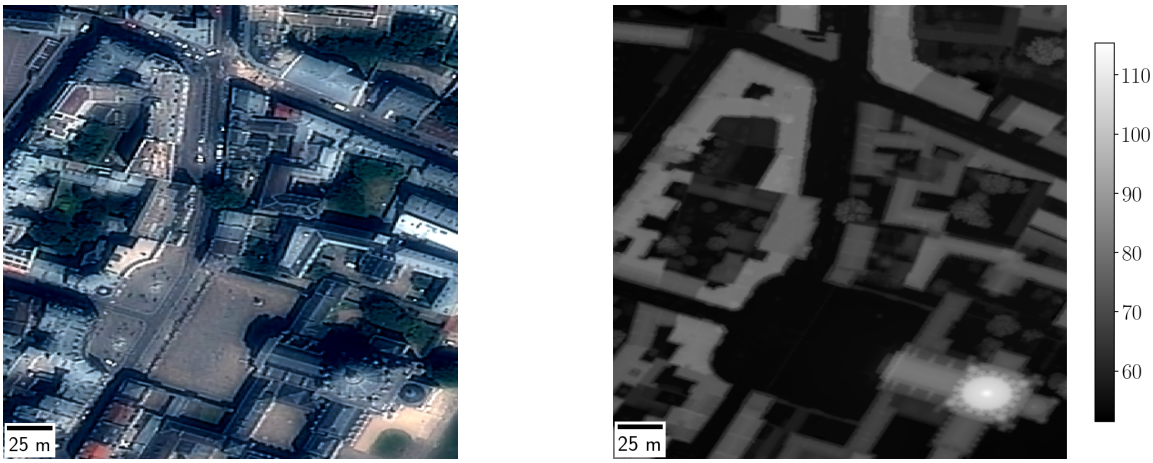
Frank Herbert, *Dune*

Introduction

Knowing the Earth's topography is crucial for modern geosciences. Depending on the level of detail needed, different models can be used: the Earth ellipsoid, its geoid (gravity equipotential surface), topographic maps (*i.e.* contour lines of hiking maps) *etc.* One of those models is the Digital Surface Model (DSM), which is a representation of a surface's elevation on a regular grid. This type of model appears as a natural solution in many Geographic Information Systems (GIS). Indeed, they can easily be handled and provide georeferenced information regarding the topography of an area. Figure 3 presents an example of a DSM.

DSMs find usage in various contexts and for a wide range of applications. In Earth Observation (EO) for instance, DSMs are used to monitor changes in vegetation [Sadeghi et al., 2016], melting rates of glaciers [Berthier et al., 2014] or water resources [Yamazaki et al., 2019]. DSMs can also be employed for catastrophe management, for instance to predict the potential damage caused by earthquakes or floods [Jenkins et al., 2023]. DSMs are also crucial for ortho-rectifying images, *i.e.* geometrically correcting the effects of distortion due to the sensor's angle of view and the terrain's topography. Finally, high resolution DSM can help drone navigation in urban settings, for Defense applications, or more broadly for urban planning [Velazco, 2012].

There are multiple ways of creating a DSM from remote sensors such as planes, drones, or satellites. The first way is to use Radar interferometry, as done by Sentinel-1 satellites [Geudtner et al., 2014], the Shuttle Radar Topography Mission (SRTM) [Farr et al., 2007] or TanDEM-X ([Krieger et al., 2007]). Typical planimetric resolutions obtained are in the range of a dozen meters (10 m for TanDEM-X or 30 m for the SRTM). Another method is to construct DSM by means of stereophotogrammetry [Tao and Hu, 2001], *i.e.* the science of recovering 3D information from optical images. For this method, images of a scene are acquired from different points of view. Depth information is recovered from the parallax effect between images, *i.e.* the fact that objects closer to the sensors present a greater shift between images than objects in the background. This effect is also what enables depth perception in human vision. Figure 4 illustrates the parallax effect, where the top of Eiffel Tower has a greater position shift in both images than its basis. As



(a) Pléiades image ©CNES 2017, Distribution AIRBUS DS

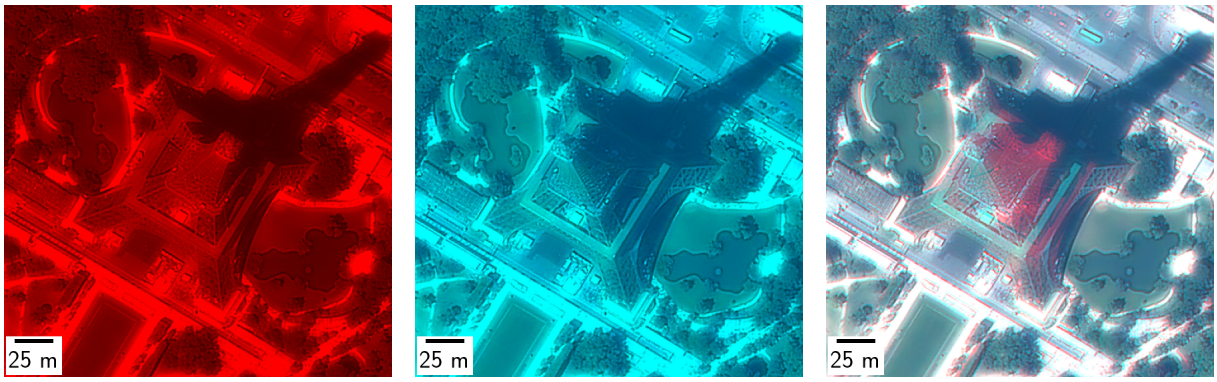
(b) Digital Surface Model from LiDAR HD (unit: m)

Figure 3: Satellite image over Val-de-Grâce, Paris at 0.5 m of resolution, and a DSM over the same area.

current optical satellites have a sub-meter resolution, it is possible to massively produce DSM covering the globe using photogrammetry at a relatively low cost. The altimetric resolution is typically around one meter, although it depends on the different acquisition angles of the satellites. The final method for producing DSM is to use Light Detection And Ranging (LiDAR) [Khosravipour et al., 2016]. Using LiDAR allows obtaining very good accuracy. For instance, the Institut National de l'Information Géographique et Forestière (IGN) is using LiDAR to cover the French territory, with a planimetric accuracy of 25 centimeters, and an altimetric resolution of 5 centimeters allowing to observe very small objects. Acquisition campaigns such as this one are carried out using airborne vehicles, and are thus costly and take a lot of time. Therefore, photogrammetry DSMs are currently the best solution to produce DSMs covering the globe with a sub-meter resolution for relatively low costs. This thesis will therefore mainly consider DSMs produced using stereophotogrammetry.

In this context, CNES - the French Space Agency - is planning to launch 4 low-cost optical satellites with Airbus Defense and Space, in order to massively produce DSM using stereophotogrammetry. This mission, named CO3D (for *Constellation Optique 3D*, [Melet et al., 2020]), was conceived jointly with IGN to provide high resolution DSM over the globe, at a 50 cm resolution.

For this purpose, CNES developed a pipeline dedicated to process all images provided by the CO3D satellites automatically and at a very large scale. This pipeline is called Chaîne Automatique de Restitution Stéréoscopique (CARS), and is composed of many image processing steps. These steps will be detailed in the first chapter, but can be summarized as follows:



(a) Left Image (Red channel) (b) Right Image (Blue and Green channel) (c) Anaglyph

Figure 4: Example of the parallax effect: the top of the Eiffel Tower has a greater shift in-between images than its basis. The anaglyph contains the red band of the left image and the green and blue bands of the right image. (Pléiades © CNES 2023, Distribution AIRBUS DS)

- Resampling of images in a convenient geometry for matching pixels
- Dense matching of every pixel between stereo images
- Triangulation of matched pixels into 3D points
- Rasterization, *i.e.* projection of the 3D points onto a regular grid, therefore yielding the final DSM

The hardest and most crucial part of this pipeline is the dense matching of pixels. It is also a famous problem in computer vision, which finds applications in robotics and autonomous cars, for instance [Geiger et al., 2013]. Alongside the DSM computation, one of the requirements of the CO3D mission is to produce a performance map indicating the estimated quality of each cell of the DSM. This has motivated CNES to lead research in order to estimate the uncertainty amongst the CARS pipeline. The main objective of this thesis is therefore to characterize and propagate the uncertainty throughout this photogrammetry pipeline. Considering the complexity of the CARS pipeline, as well as the time constraint resulting from the future launch of the CO3D satellites, we mainly focus on characterizing the uncertainty of the dense matching step. After quantifying this uncertainty, we propagate it to the end of the pipeline to the final DSM.

Characterizing and quantifying uncertainty has many benefits, although it can sometimes be computationally expensive to deal with. It indeed provides additional information for better decision-making and risk management. It can also allow for a better understanding of the underlying processes at stake regarding the value of interest. In many cases, uncertainty estimation is treated as a secondary objective in applications. Jointly estimating a value and its uncertainty, however, can lead to new strategies to

reduce uncertainty or even improve performance of the main applications [Qin et al., 2022, Chen et al., 2023, Jiang et al., 2024b].

Before delving into uncertainty estimation and its propagation, we first need to specify what we mean by uncertainty. Uncertainty is a situation where a measure or value of interest is not known, or not known with precision. It is subject to change, as additional information, measures or a different acquisition protocol may reduce how uncertain a value is. It can also be subjective. For instance, someone may be uncertain about the launch date of CO3D satellites, while someone else working at the launch pad might have the answer. This highlights the fact that while everyone has an understanding of what uncertainty is, it encompasses very different concepts in nature. It is common to differentiate the various types of uncertainty by dividing it into two categories: stochastic uncertainty (also called random uncertainty) and epistemic uncertainty [Hora, 1996, Frank, 1999].

Stochastic uncertainty refers to every situation of purely aleatoric nature. For instance, the result of a coin throw, random noise on a CDD captor or the Brownian movement of a particle. An operator typically encounters this kind of uncertainty in a situation where they have access to many measures or observations of the same value of interest. It is usually modeled mathematically with a frequentist approach, using probability measures such as the uniform distribution, Gaussian distribution or the Student's t distribution.

On the other hand, epistemic uncertainty refers to a situation where the value of interest is not known or ill-known due to a lack of knowledge. Think of the previous example with the launch date of satellites, or if someone was asked to guess Io's mass, one of the moons of Jupiter. There is no random process at stake here, and there is usually no point of acquiring multiple samples of the measure if we have a reliable and precise sensor. Once the value of interest is known, the uncertainty disappears. It has been proposed to model this kind of uncertainty using a Bayesian approach for probability, by opposition with the frequentist approach. Probabilities here represent a state of knowledge, or degree of belief, one has over the value of interest. It can be updated with additional knowledge, thus leading to the notion of prior and posterior probabilities. We will see during this thesis that other models can be used to characterize this uncertainty, for instance Imprecise Probabilities (IP) and more specifically possibility distributions. In satellite imagery, those models have been used to detect land changes [Lesniewska-Choquet, 2020] for instance.

During this thesis, we contributed both to the field of photogrammetry and to the field of IP. Here is a quick overview of the contents that can be found in the following chapters:

- Chapter 1 introduces the different stereophotogrammetry concepts considered in this thesis. It focuses on the stereo pipeline developed by CNES and its sources of uncertainty, which will be considered in our applications.
- Chapter 2 will introduce the different uncertainty models considered in this thesis, mainly possibility distributions and copulas.
- In Chapter 3, we propose different methods for creating specific multivariate uncertainty models based on the models introduced in Chapter 2. We also study the relationships between the methods we introduced.
- Chapter 4 uses the concepts of Chapter 2 and the results of Chapter 3 to propagate uncertainties modeled by possibility distributions in a part of the dense matching step of the stereo pipeline.
- Chapter 5 also uses possibility distributions, but this time to characterize the uncertainty of the dense matching step itself. Using this method, we are able to obtain confidence intervals at the end of the dense matching step.
- Chapter 6 propagates the disparity intervals to the end of the pipeline, in the form of elevation intervals on the final DSM.

As our contributions concern two distinct fields of research, multivariate uncertainty and photogrammetry, readers with a level of expertise in one field might be less interested in the second field. We tried, as much as possible, to write each chapter so it can be read and followed by everyone, although some details might need additional knowledge in a field of expertise. To help readers navigate through chapters according to their areas of interest, here is an attempt to classify each chapter into its field of research.

- Chapter 1 focuses on stereophotogrammetry.
- Chapters 2 and 3 focuses on the modeling of uncertainty, with Chapter 3 delving into more advanced concepts.
- Chapter 4 joins both fields, but leans a bit more towards uncertainty propagation than towards photogrammetry.
- Chapters 5 and 6 also attempts to join both fields of research, but focuses almost completely on photogrammetry.

The rest of this section lists the contributions and research events that occurred during this thesis.
National conferences:

- LFA 2022: “Copules, probabilités inférieures et ensembles aléatoires : comment et quand les appliquer ?” [Malinowski and Destercke, 2022b]

International conferences:

- SMPS 2022: “Copulas, Lower Probabilities and Random Sets: How and When to Apply Them?” - [Malinowski and Destercke, 2022a]
- ISIPTA 2023 (Special jury recognition Award): “Uncertainty Propagation using Copulas in a 3D Stereo Matching Pipeline” - [Malinowski et al., 2023]
- IGARSS 2024: “Robust Confidence Intervals For Digital Surface Models Using Satellite Photogrammetry” - [Malinowski et al., 2024b]

International journals:

- International Journal of Approximate Reasoning: “Uncertainty propagation in stereo matching using copulas” - [Malinowski et al., 2024a]

Not yet published pre-print:

- Available on ArXiv: “Robust Confidence Intervals in Stereo Matching using Possibility Theory” - [Malinowski et al., 2024c]

Workshops, poster sessions:

- Belief 2022 conference: Poster presentation "Using Copulas with Random Sets"
- Workshop Imagin “journée imprécision et incertitude en analyse et traitement d’images”: Funding of the event and communication, and oral presentation “Uncertainty Propagation in Dense Matching”
- SFPT “Pléiades Neo: de nouveaux satellites pour de nouveaux usages”. Oral presentation: “Confidence Intervals for Digital Surface Models”
- GdR IASIS “Télédétection et Climat”. Oral presentation “Estimation d’Incertitudes dans la Création de Modèles Numériques de Surface issus d’Imagerie Spatiale”

Chapter 1

Principles of Stereophotogrammetry using Satellite Imagery

This section will present important concepts regarding photogrammetry that will be relevant throughout this thesis. Section 1.1 will present the main 3D product we will consider, *i.e.* DSM, while Section 1.2 will introduce the different sensors and satellites employed. Section 1.3 presents in details the different processing steps for creating DSMs from stereo images. Finally, Section 1.4 addresses the modeling of uncertainty in photogrammetry.

1.1 Digital Surface Models

Topographical information is crucial for many applications in Earth Observation (EO), or more generally when manipulating georeferenced data. A popular and simple model containing such topographical data named Digital Elevation Models, and represents elevation data using a regular grid as in Figure 3(b). In the literature, a distinction is usually made between two types of DEM: Digital Surface Model and Digital Terrain Model. DTMs only represent the ground surface, without man-made structures (buildings, houses) or rapidly evolving volumes such as vegetation or seasonal snow. Those finer details are instead included in DSMs. Figure 1.1 illustrates the difference between the two.

DTMs are used for hydrology applications, such as for estimating water reservoirs [Yamazaki et al., 2019], river flow [Miguez-Macho et al., 2007], global snow resources [Gascoin et al., 2019] or modeling potential flood [Yamazaki et al., 2014]. On the other hand, the high level of details of DSMs are used to monitor changes in vegetation [Sadeghi et al., 2016], melting rates of glaciers [Berthier et al., 2014, Rieg et al., 2018], changes in volcanoes [Ganci et al., 2022] or to precisely map snow depth on mountainous terrain [Marti et al., 2016]. In urban environments, DSMs allow predicting the potential

Figure 1.1: Difference between Digital Terrain Model and Digital Surface Model

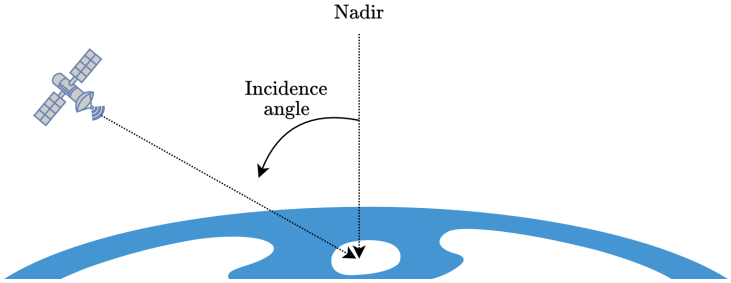


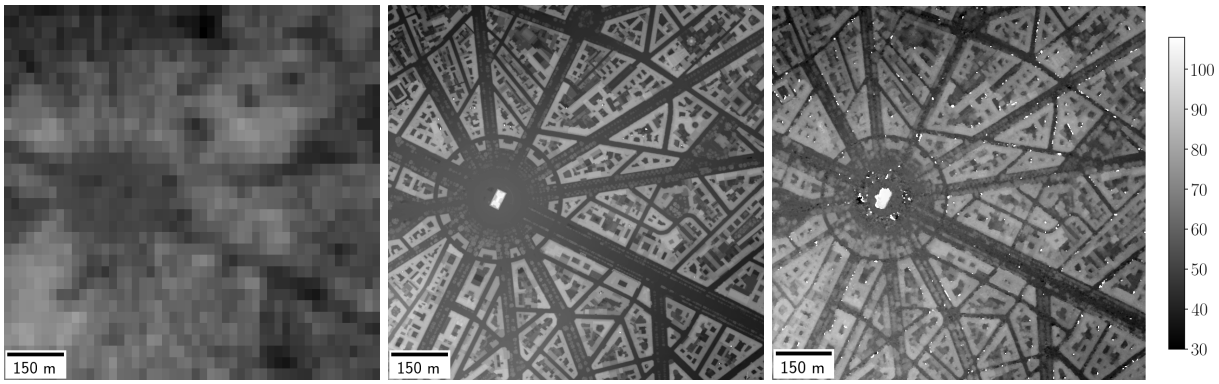
Figure 1.2: Incidence angle of a satellite, and nadir direction

damage caused by floods [Jenkins et al., 2023] or to do urban planning [Velazco, 2012]. For instance, possessing DSMs on the same city but for different acquisition dates allow measuring the growth rate of the city [Warth et al., 2019], or damages caused by an earthquake [Erdogan and Yilmaz, 2019] in-between acquisitions. DSMs are also crucial for ortho-rectifying images, *i.e.* geometrically correcting the effects of distortions between the sensor and the terrain [Toutin et al., 2012]. When capturing an optical image, the satellite is not necessarily right above its target, as illustrated in Figure 1.2. However, if the image must be used as a background for a map, it is usually required that the image must be captured vertically, referred to as nadir, in order to be free of any perspective effect. DSMs allow to reproject images as if they were taken at nadir. This rectification is used for many satellite products [Hagolle et al., 2017].

Many applications depend on the resolution of DSMs: evaluating water reservoirs in mountainous regions does not necessarily require the same accuracy as 3D urban modeling, for instance. The accuracy of a DSM is defined by its planimetric resolution, *i.e.* the size of a cell, and its altimetric resolution, *i.e.* the smallest elevation variation it can detect. The altimetric resolution is itself characterized by two notions: the absolute and relative accuracy. Absolute accuracy is the precision of a DSM at a pixel level, *i.e.* how close the predicted elevation is to the true elevation at the geographic coordinates of a given cell. Relative accuracy characterizes the quality of the 3D reconstruction of a scene. For instance, a DSM precisely reproducing the shape of a house and its chimney will have

a high relative accuracy. However, if the house is not properly georeferenced then the DSM will have a poor absolute accuracy. Relative accuracy is therefore always greater than the absolute accuracy. In this thesis, we will mainly consider relative accuracy when evaluating the quality of DSMs, as our sources possess different georeferencing systems which naturally induces small planimetric errors.

There are multiple ways of creating a DSM from airborne sensors such as planes, drones, or satellites. The first way is to use Radar interferometry, as done by Sentinel-1 satellites [Geudtner et al., 2014] for instance. This method presents the advantage of being able to acquire data by day and night, even in the presence of clouds. Radar sensors, such as those present in the Shuttle Radar Topography Mission (SRTM) [Farr et al., 2007] or TanDEM-X [Krieger et al., 2007], possess a large swath of around 30 km, which was used to produce DSMs covering the majority (or totality for TanDEM-X) of emerged land. The planimetric accuracy is in the range of a dozen meters (10 m for TanDEM-X or 30 m for the SRTM), and relative altimetric resolution is typically of a few meters (2 m for TanDEM-X and 8 m for the SRTM). This is not sufficient to distinguish objects such as buildings or trees, but rather global terrain topography. Figure 1.3(a) presents a TanDEM-X DSM. SRTM and TanDEM-X DSMs are not meant to be produced on a regular basis, and thus do not allow for temporal analysis as such. Another method for constructing DSMs is to use stereophotogrammetry [Tao and Hu, 2001]. Photogrammetry is the science of recovering 3D information from optical images. For this method, images of a scene are acquired from different points of view, and depth information is recovered from the parallax effect between images, already presented in Figure 4. Photogrammetry is not restricted to the production of DSMs, as it has many usages in robotics, for instance on Martian rovers [Goldberg et al., 2002] or autonomous driving [Geiger et al., 2013]. Current optical satellites have a sub-meter resolution and a large swath (20 km for Pléiades for instance [Coeurdevey and Gabriel-Robez, 2012]), allowing to massively produce high-resolution DSMs covering the globe for a relatively low cost using photogrammetry. The altimetric resolution is typically around one meter, although it depends on the different acquisition angles of the satellites. With this resolution, buildings, trees, and even cars can be detected on the DSM. Another method for producing DSMs is to use LiDAR (laser sensors) [Khosravipour et al., 2016]. Space-borne LiDARs are mostly used for atmospheric or discrete measurements [Fouladinejad et al., 2019]. For instance, NASA's ICESat-2 [Jasinski et al., 2020] measures elevation of seas and glaciers using 6 lasers taking measurements along track, which is not adapted for reconstructing high-accuracy DSM. On the other hand, planes or drones equipped with LiDAR sensors usually have very good accuracy, allowing to model tower cranes, lamp post or small details on buildings. In particular, IGN uses LiDAR to create point clouds with a planimetric accuracy of 25 centimeters and an altimetric resolution of 5 centimeters, and then produce DSMs out of it [Monnet,



(a) Radar DSM (b) LiDAR DSM (c) Photogrammetry DSM

Figure 1.3: DSMs over Arc de Triomphe, Paris, obtained with different sensors.

2023, IGN, 2024]. This is part of a 5-year campaign named LiDAR HD that started in 2021, where planes equipped with LiDAR sensors will cover the whole French territory (except French Guiana). Those types of campaign are long and costly, and can hardly be reproduced at a global scale. Therefore, photogrammetry DSMs are currently the best solution to produce DSMs covering the globe, with a sub-meter resolution for relatively low costs. Figure 1.3 presents examples of DSMs produced by Radar, photogrammetry and LiDAR.

In this thesis, we use LiDAR DSMs as our reference DSMs due to their high quality. They will be used to evaluate the errors or uncertainties in DSMs produced using stereophotogrammetry. The next section details which stereo images we will be using to produce those DSMs.

1.2 Satellite Photogrammetry

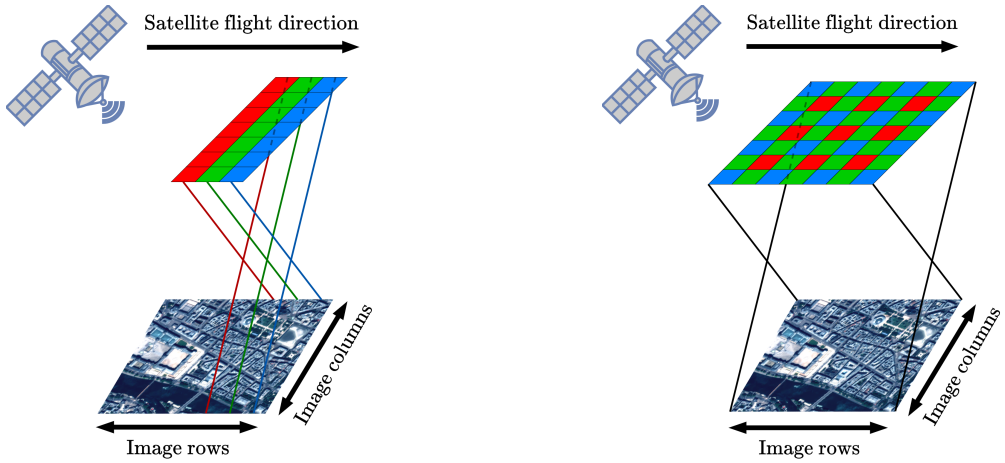
This section presents the different sensors, satellites, and geolocation models that will be used to acquire and process stereo images.

1.2.1 Different sensors

Different types of sensors can be used to acquire optical satellite images. We detail in this section a (non-exhaustive) list of sensors of interest [CNES et al., 2008].

Push-broom sensor

For each observed wavelength, push-broom sensor are composed of a single cell row, acquiring simultaneously radiometric information alongside a line perpendicular to the direction of the satellite. We use the satellite movement along track to acquire the different rows of the image, as seen in Figure 1.4. As only one line of cells is needed, push-broom



(a) Push-broom sensor (b) CDD sensor

Figure 1.4: Schematic representation CDD and push-broom sensors

sensors are simple systems which can capture images continuously, while guaranteeing good geometrical quality along the rows of the images. A variation of those sensors are TDI sensors (Time Delay Integration). Those sensors function as a push-broom except that each row has the ability to transfer its photon charges to the next row. This allows to capture signals over a longer period of time, thus improving the signal-to-noise ratio. Harder to produce, TDI sensors also require a precise control of the satellite so that observed objects stay within a column of the TDI sensor.

CCD sensors

Charge Coupled Device (CDD) are classical sensors, used in current digital cameras, for instance. They consist of a grid of cells sensitive to a given wavelength, often configured in a checkerboard pattern. A specific type of CDD sensors are Bayer matrices, where the sensor only possesses a single type of photo-sensitive cell and the different colors are captured by applying a color filter on the sensor. CDD sensors possess multiple advantages, such as good geometrical quality as all pixels are acquired simultaneously, or the possibility to perform many acquisitions with various angles possible. CDD sensors are relatively recent in spatial imagery, as they are technologically difficult to build. Augmenting the number of pixels complicates the shutter function and occupies more space compared to push-broom sensors. More radiometric calibration is required since each pixel results from a different photo-sensitive cell. Acquire long segments of an image is not natural, as it is the case for push-broom sensors.

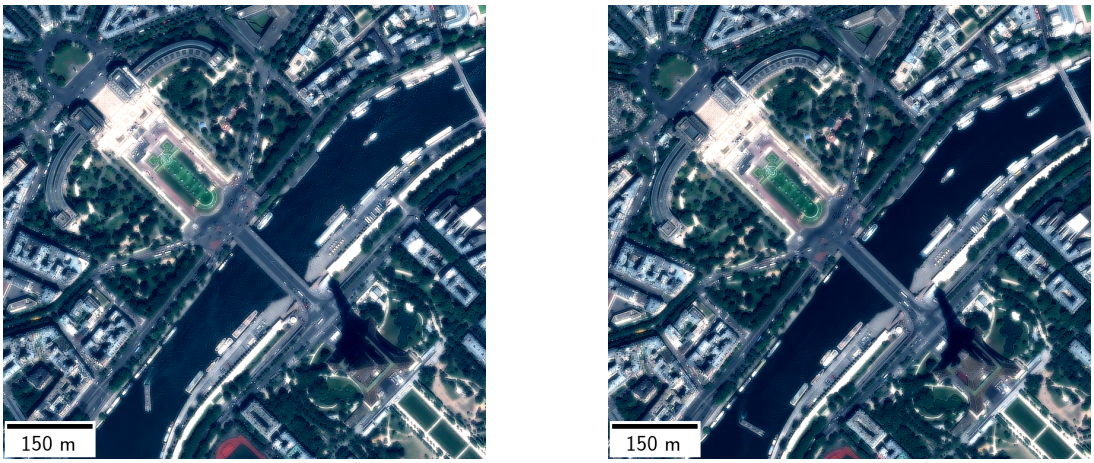
Those different sensors are used in most Very High Resolution (VHR) optical satellites, as we will see in the next section.

1.2.2 VHR satellites

In this section, we present different VHR optical satellites which can be used to acquire stereo images.

Across the years, different constellations of VHR satellites have been launched, either for civilian (SPOT 1-7), defense (CSO) or commercial use (Ikonos, QuickBird, Worldview 1-3). Most of them are agile satellites, using push-broom sensors that can now reach resolutions near 30 cm [CNES et al., 2008, Coffer et al., 2022]. One constellation of interest that we will consider in this thesis is the Pléiades constellation. Developed by Airbus, this constellation is composed of two identical satellites, 1A and 1B. Satellites were launched in 2011 and 2012 in a heliosynchronous orbit at 690 km, for both civilian and defense usages. They provide panchromatic images with a resolution of 70 cm (resampled to 50 cm), and RGB-NIR images at a resolution of 2 m, with a 20 km swath (`https://dinamis.data-terra.org/pleiades/`) using a push-broom TDI sensor. Their high agility and revisit rate allow them to capture stereo and tri-stereo images for any location on the globe, ideal to produce DSMs with high accuracy. Figure 1.5 provide an example of a Pléiades stereo acquisition. A "video" mode is even available, where dozens of images of the same scene can be acquired in the span of a few minutes [Lebègue et al., 2015]. However, stereo acquisitions are not the only objective of this mission, even though the demand for those products is increasing [Berthier et al., 2014, Poli et al., 2015, Rieg et al., 2018, Loghin et al., 2020]. The acquisition of stereo images is thus provided on command, which can conflict with other usages of the satellite, and can become costly when trying to cover large areas. Pléiades is also a satellite for both defense and civilian usages, military acquisitions having the priority over civilian one. Pléiades acquisitions can also be requested using the "Disaster Chart" (`https://disasterscharter.org/fr/web/guest/home`) in order to evaluate damages caused by floods, landslide or earthquakes for instance, and to better plan and provide emergency relief to victims of those disasters.

In order to produce a worldwide DSM with 1 m resolution by 2025, the Centre National d'Etudes Spatiales (CNES) and Airbus are launching the Constellation Optique 3D (CO3D) mission [Melet et al., 2020]. It will be composed of two pairs of low-cost satellites (Figure 1.6) equipped with VHR optical sensors. The mission will produce images in the RGB and NIR spectrum at 0.5 m of resolution [Lebègue et al., 2020] using a CDD sensor cell, and more specifically a Bayer matrix. The pairing of satellites and CDD sensor used allow for *almost* simultaneous stereo image acquisition, cutting short the transient object problem (*i.e.* objects moving/disappearing between stereo images). To be able to process the amount of data provided by the CO3D mission (40 000 images a day, at 50 cm resolution and covering a footprint of 7 km $\times$ 5 km [Melet et al., 2020, Lebègue et al., 2020]), every step of the pipeline has been developed to be highly scalable. Different acquisition



(a) Left image (b) Right image

Figure 1.5: Pansharpened Pléiades stereo images over Paris at 0.5 m of resolution. The change of point of view can be easily observed by looking at the Eiffel Tower. Pléiades ©CNES 2023, Distribution AIRBUS DS

Figure 1.6: Pairing of satellites for the CO3D mission. (Credit: CNES)

schemes can also be used, such as the video mode, or even the 'diamond' geometry acquisition, which acquires a quadri-stereo over a couple of days. In parallel with image quality specifications, the CO3D products need to abide to an elevation relative accuracy of 1 m on low slopes. In addition, the CO3D mission plans to produce a performance map supporting the output DSM. Therefore, investigating uncertainty inside a 3D stereo pipeline is essential for the implementation of this performance map.

> **Remark:** When an image is acquired both in panchromatic and RGB mode, it is possible to leverage the high resolution of the panchromatic image to improve that of the color image. This fusion technique is called *pansharpening* [Loncan et al., 2015]. We use this technique for clarity in figures and other illustrations of this thesis (for instance Figures 3(a) and 1.5). It is important to remember that the processed images are the panchromatic images, and not the pansharpened ones which are only used for

> the final visualization. The CO3D satellites are not concerned with pansharpening, as the used Bayer matrix directly provides RGB images at 50 cm resolution without a panchromatic band.

Although this thesis contributes to the ground segment of the CO3D mission, we used Pléiades stereo images in the absence of CO3D images (at the time this thesis is written, the CO3D mission has not yet been launched).

1.2.3 Geolocation Models

A crucial part of satellite imagery is the ability to perform georeferencing, or georegistratation, of every pixel, *i.e.*, locate their coordinates in an Earth system of coordinates.

Different systems of coordinate exist, for instance the Geodetic Coordinates use latitude and longitude to represent the relative position of a point with regard to a reference ellipsoid, *i.e.* a smooth and regular mathematical approximation of the Earth's shape. Another widely used reference system is the Universal Transverse Mercator (UTM) coordinate system, which divides the globe into sixty north-south areas. Each area is 6° of longitude wide, and is approximated by a plane. The coordinates of a point in each area are Cartesian coordinates, with the origin being the intersection of the equator and the meridian of the area. UTM coordinates are employed in GPS systems, for instance. In Geodetic Coordinates or UTM, the elevation of a point is defined using the ellipsoid as reference. It is also possible to use the geoid, *i.e.* the gravity equipotential surface, which is less smooth than the ellipsoid but is closer to the actual irregular shape of the Earth.

Physical models possess high geolocation accuracy, but are sensor-specific and are computationally complex. For stereo reconstruction, generalized sensor models are preferred. We will focus on Rational Polynomial Models (RPC), provided alongside images of many satellites [Grodecki, 2001, Devika et al., 2006], and specifically for the CO3D and Pléiades satellites. Sometimes called Polynomial Mapping Functions [Baltsavias and Stallmann, 1992] or Rational Function Models [Tao and Hu, 2001], RPC are functions allowing to transform a pixel's ground location in Geodetic Coordinates, *i.e.* its longitude, latitude and geodetic height $(X,\ Y,\ Z)$, into its coordinates $(row,\ col)$ in the image space. RPC encode lines of sight of the satellite, *i.e.* the line joining the center of the sensor's cell to the ground and going through the optical center of the sensor. To improve numerical stability and minimize computation errors, the image coordinates and ground coordinates are normalized between -1 and 1, using their scale factors SF and mean values:

$$\begin{aligned} SF_X &= \max(X_{\max} - \overline{X},\ \overline{X} - X_{\min}) \\ \tilde{X} &= \frac{X - \overline{X}}{SF_X} \end{aligned}$$

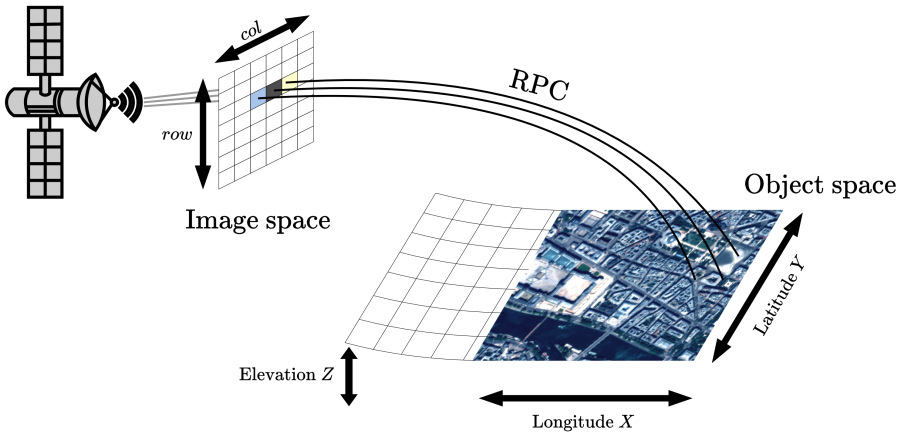


Figure 1.7: Schematic representation of RPC models.

The same process is applied to $Y,\ Z,\ row$ and col. To avoid cumbersome notation in this section, we will refer to every normalized coordinate using their non-normalized symbol.

Formally, RPC are defined as rational fractions of polynomials:

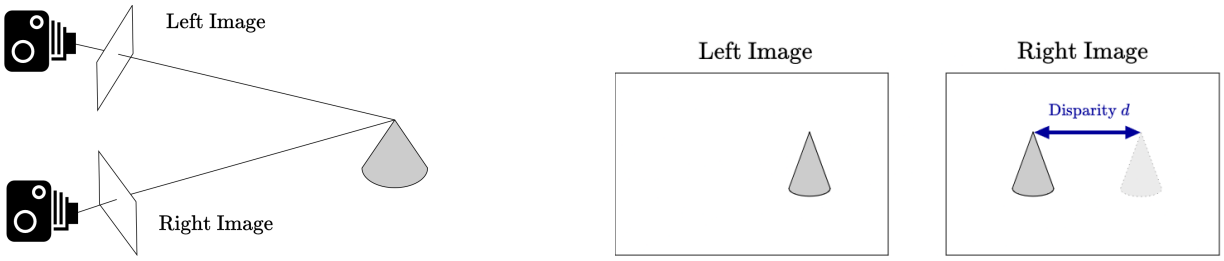
$$\begin{aligned} \text{RPC} : \mathbb{R}^3 &\rightarrow \mathbb{R}^2 \\ (X,\ Y,\ Z) &\mapsto \left(\frac{Num_{row}(X,Y,Z)}{Den_{row}(X,Y,Z)}, \frac{Num_{col}(X,Y,Z)}{Den_{col}(X,Y,Z)} \right) \\ &\mapsto (row, col) \end{aligned} \tag{1.1}$$

where $Num_{row},\ Den_{row},\ Num_{col}$ and Den_{col} are the numerators and denominators for rows and columns respectively, expressed as polynomials with a maximum order of 3:

$$\begin{aligned} Num_{row}(X,\ Y,\ Z) &= \sum_{i=0}^{3} \sum_{j=0}^{3-i} \sum_{k=0}^{3-i-j} a_{ijk} X^i Y^j Z^k \\ &= a_{000} + a_{100}X + a_{010}Y + a_{001}Z + a_{110}XY + a_{101}XZ \\ &\quad + a_{011}YZ + a_{200}X^2 + a_{020}Y^2 + a_{002}Z^2 + a_{111}XYZ \\ &\quad + a_{210}X^2Y + a_{201}X^2Z + a_{120}XY^2 + a_{102}XZ^2 \\ &\quad + a_{021}Y^2Z + a_{012}YZ^2 + a_{300}X^3 + a_{030}Y^3 + a_{003}Z^3 \end{aligned}$$

$Den_{row},\ Num_{col}$ and Den_{col} respectively possess different coefficients a_{ijk}. The order or indexing of a_{ijk} may differ in the literature. For instance, they can be numbered from 0 to 19 or 1 to 20, and do not refer to the same indeterminate. Numerical values of RPC coefficient are computed using a set of reference ground control points.

It has been shown in [Baltsavias and Stallmann, 1992] that RPC are well suited to be used for ortho-rectification and stereophotogrammetry, as they possess good accuracy and are computationally fast. For stereophotogrammetry, it is often required to use the inverse RPC model. As RPC encode a line of sight given an image coordinate, the use of



(a) (b)

Figure 1.8: Principle of photogrammetry. The same object is captured by two different cameras in (a). The position of the object in both images spaces as shown in (b) is characterized by its displacement, called *disparity*.

an additional elevation coordinate is required to move from the image space to the object space (*i.e.* to go from a 2D space to a 3D space). It can be a geoid modeling the Earth's surface, or a DSM with higher resolution. Knowing the true elevation Z of a pixel, we define the inverse model as:

$$\begin{aligned} \mathrm{RPC}^{-1} : \mathbb{R}^3 &\rightarrow \mathbb{R}^3 \\ (row,\ col,\ Z) &\mapsto (X,\ Y,\ Z) \end{aligned} \tag{1.2}$$

An illustration on how RPC can be used to go from the image space (row, col) to the object space (X, Y, Z) is presented in Figure 1.7. Knowing the position of pixels in the object space using RPC allows triangulating the position of an object using multiple images of the same scene. This is how photogrammetry pipelines processing satellite images are computing the elevation of the different cells of a DSM. The next section will explain the inner workings of those pipelines.

1.3 Structure of the Stereophotogrammetry Pipeline

This section delves deeper into details of the inner workings of stereophotogrammetry, and presents the stereo pipeline mainly considered in this thesis. Photogrammetry is the science of deducing information from photographic images [Kasser and Egels, 2001]. A subdomain of photogrammetry is stereophotogrammetry, which specifically consists of deducing 3D information from multiple photographic images. Although multiple stereophotogrammetry setups can be achieved, for instance using structured light [Scharstein and Szeliski, 2003] or different wavelengths [Geng, 1996], we focus here on the pipelines designed for processing satellite images, which are used and studied in this thesis.

The main idea of stereophotogrammetry is to identify the parallax of objects between multiple images, and to deduce the distance between the object and the sensors from this displacement. Figure 1.8 illustrates how the position of an object can be triangulated from

multiple images of a scene. The two lines of sight used to triangulate the position of an object are determined by finding the location of the object in both image spaces. Given the coordinates of an object in a reference image space (for instance the left image in Figure 1.8), the corresponding line of sight in the second image is completely determined by the position of the object in the reference image space, and its displacement between the two images spaces. When expressed in pixels, this displacement is called disparity, as presented in Figure 1.8(b). Determining the disparity of every pixel of a reference image is called dense-matching, and is done by a dense-matching algorithm, also called *correlator* (as it computes the best correlation between pixels of stereo images). It allows creating pairs of pixels representing the same object in their respective image space. Using the geometrical model of the camera, or the geolocation model of sensors in the case of satellite imagery, the 3D coordinates of the corresponding object of a pair of matched pixels can then be determined by intersecting lines of sight (or best approximation if they do not strictly intersect). This results in a point cloud that encodes the 3D position of every visible object in both images. In satellite photogrammetry, the point cloud can be projected onto a regular grid to obtain a DSM. The following section delves deeper into details of satellite photogrammetry pipelines, and specifically on the CARS stereo pipeline.

1.3.1 Different Stereophotogrammetry Pipelines

Several stereo pipelines processing satellite images exist in the literature. We can think of NASA's *ASP* [Shean et al., 2016], Centre Borelli's *s2p* [Franchis et al., 2014], DLR's *CATENA* [Krauß et al., 2013], Ohio State University's *RSP* and *SETSM* [Qin, 2016, Noh and Howat, 2017], IGN's *MicMac* [Rupnik et al., 2017]. Concerning MicMac, we only consider the methods for satellite photogrammetry, as many other photogrammetry applications and methods are available in this software. For a comparison of the performance of those pipelines, we refer to [Haala, 2014, Bosch et al., 2017, Qin et al., 2021].

These pipelines roughly share the same structure, which will be presented in detail in the following sections. They can be briefly summarized as follows:

- **Pre-processing**. All those pipelines undergo a bundle adjustment step to refine images geometries and correct sensors' misalignment. The pre-processing steps can sometimes be found in a module separated from the main pipeline.
- **Images resampling in a convenient geometry for pixel matching**. Most pipelines propose multiple geometries for the resampling of images: the matching can directly be done in the object space, or the secondary image is resampled in the geometry of the reference images, or both images are resampled into an epipolar geometry, *i.e.* a geometry where objects move horizontally in both image spaces.

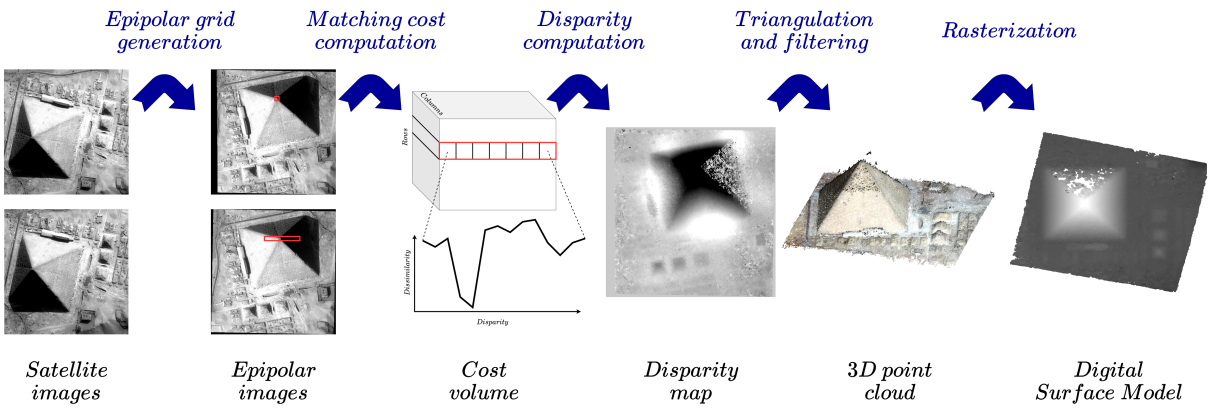


Figure 1.9: Different steps of the CARS pipeline

- **Dense matching**. RSP and S2P restrict the choice of the geometry to the epipolar geometry and therefore only propose to do 1D (row-wise) dense matching. Other pipelines allow for a 2D matching.
- **Triangulation** of matches to get 3D point cloud.
- **Rasterization to project elevation on a regular grid**. When working with more than two images, pipelines all offer the possibility to fuse the obtained pair-wise DSMs to obtain a global DSM.

It is also noteworthy that when working with more than two images, only MicMac and ASP allow considering all images at once, while other pipelines use multiple pairings of images. The way those pipelines handle uncertainty will be detailed in section Section 1.4.1.

We will focus on the CARS pipeline used in this thesis, developed by CNES [Michel et al., 2020]. A schematic of the different steps of the pipeline are presented in Figure 1.9.

The next sections will look specifically at the CARS stereo pipeline that will be used throughout this thesis.

1.3.2 Resampling in Epipolar Geometry

The CARS pipeline takes as inputs single channel images in sensor geometry alongside their geometric model (RPC in our case). As the dense matching step compares a single channel in both images, the input images are single channel images. When working with Pléiades images, we use the panchromatic channel as it has the best resolution. For CO3D images, the green channel will be used as the Bayer matrix contains more green pixels.

In many stereo setups, cameras are aligned in such a way that objects only move horizontally between images [Geiger et al., 2012, Scharstein et al., 2014, Keselman et al., 2017]. This allows to restrict the search space for pixel matches to a single row instead

of the whole image. Most people's eyes also present this alignment. Due to the satellites' orbit and their push-broom acquisition mode, it is not possible to maintain the alignment of sensors for satellite photogrammetry as it would be the case for classical stereo setups.

The first step of the CARS pipeline is to resample the stereo acquisitions into a convenient geometry to carry out the dense matching (Section 1.3.3). This geometry is the epipolar geometry [CNES et al., 2008], which is constructed so that each line of an image follow the movement of the satellite, *i.e.* the objects move only horizontally between the reference and secondary stereo images. This greatly facilitates the matching step, as the search space for a match is limited to a one-dimensional space instead of a two-dimensional space.

Sensors using a pinhole camera model have a perfectly defined epipolar geometry, modeled by an affine transform [Hartley and Zisserman, 2004]. This is not true for push-broom sensors [Morgan et al., 2004] such as those used in Pléiades satellites. In the case where no analytical model exist, approximations of the epipolar geometry must be computed [Oh et al., 2010, Koh and Yang, 2016, Michel et al., 2020]. The epipolar geometry is computed using the respective geolocation models of both images RPC_1 and RPC_2. Indeed, given the altitude Z of the object represented by a pixel $(row_1,\ col_1)$ in the reference image space, we can deduce its ground location $(X,\ Y,\ Z)$ using RPC_1^{-1}. The position in the secondary image space $(row_2,\ col_2)$ can then be retrieved using RPC_2:

$$(row_2,\ col_2) = \mathrm{RPC}_2 \circ \mathrm{RPC}_1^{-1}(row_1,\ col_1,\ Z) \tag{1.3}$$

The function $\mathrm{RPC}_2 \circ \mathrm{RPC}_1^{-1}$ is called the co-location function $f_{1\to2}$ and allows switching from the reference image space to the secondary image space given an elevation Z. By varying the elevation in the range of considered elevations $[Z_{min},\ Z_{max}]$, $f_{1\to2}$ provides a characterization of the parallax between images. The lines described by $f_{1\to2}$ are thus the epipolar curves in the secondary image. Similarly, $f_{2\to1} = \mathrm{RPC}_1 \circ \mathrm{RPC}_2^{-1}$ provides the epipolar curves in the reference image. The range of considered elevations $[Z_{min},\ Z_{max}]$ can be determined using any elevation model. The geoid of the Earth can be used, and there also exists open elevation data such as the NASA's SRTM [Farr et al., 2007] (30 m and 90 m of resolution between $-56°$ and $60°$ of latitude), or ESA's Copernicus DEM (30 m and 90 m of resolution worldwide).

In practice, determining the epipolar grids using $f_{1\to2}$ for every pixel is computationally heavy. Instead, $f_{1\to2}$ can be used to compute local affine approximation of the epipolar geometry (similar to what is used with pinhole cameras) as in [De Franchis et al., 2014]. When working with large images, tiling effects appear at the border between local approximations. To solve this issue, the CARS pipeline computes a deformation grid

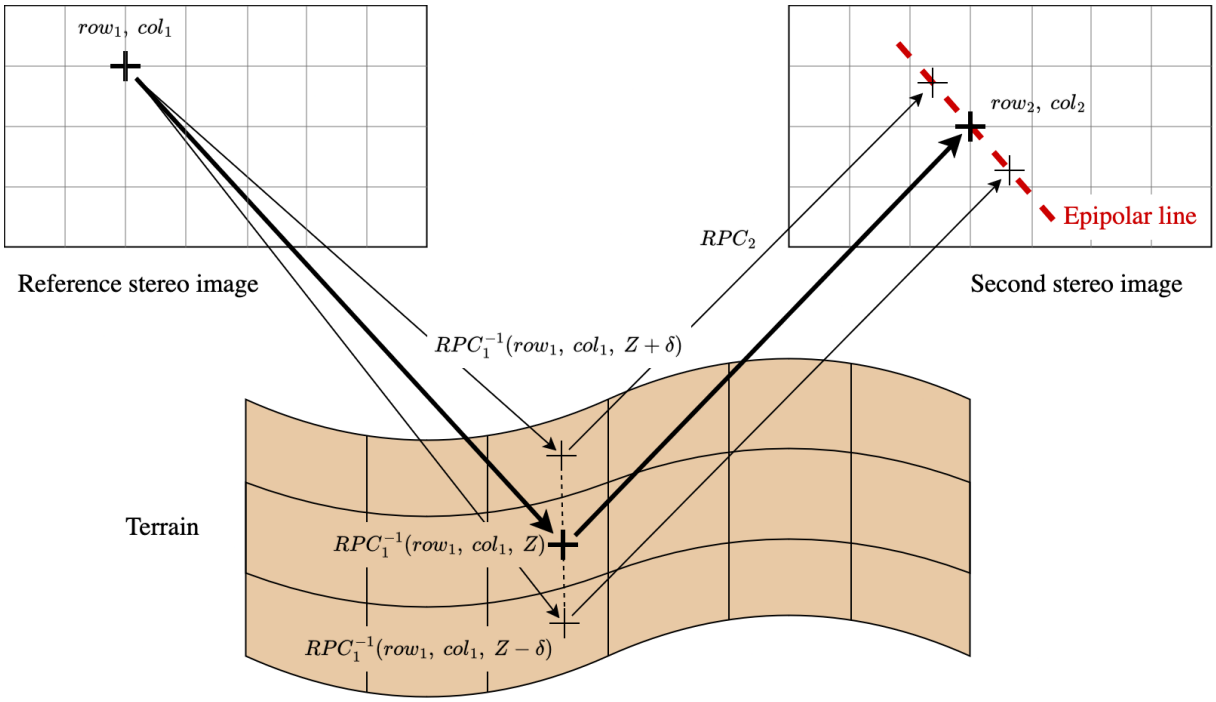


Figure 1.10: Computation of an epipolar line for a pixel using small elevation variations

approximating the epipolar geometry, as epipolar lines fluctuate slowly inside an image. The sampling rate of the grid is chosen as to be large enough to facilitate the computation while ensuring to grasp variations of the epipolar geometry.

In order to determine the deformation grid, Equation (1.3) is evaluated with an elevation Z_{coarse} extracted from the low resolution elevation model, and for small variations of altitude $Z_{coarse} \pm \delta$. These three points allow computing the direction of epipolar lines for every pixel of both images, as in Figure 1.10. This method generates the deformation grid g_{e1} joining every epipolar coordinate $(row_e,\ col_e)$ to its position in the reference image $(row_1,\ col_1)$:

$$g_{e1}(row_e,\ col_e) = (row_1,\ col_1) \tag{1.4}$$

A similar grid g_{e2} is determined for the secondary image, which is computed jointly with g_{e1}. For more details on the way epipolar grids are computed, we refer to [Michel et al., 2020]. Using those grids, it is possible to resample the reference and secondary images in their respective epipolar geometry.

> **Remark:** With this method, an object whose elevation equals that of the reference elevation model Z_{coarse} (*i.e.* the geoid, or a low resolution DSM such as the Copernicus DEM) has a disparity of 0 between epipolar images. After the dense matching step, a disparity of 0 would mean that the object has the same altitude as the low resolution

elevation model.

Because the geolocation models have a limited accuracy, there might be a misalignment left in the epipolar grids. To correct this error, a set of Scale-Invariant Feature Transform (SIFT) points [Lowe, 2004] is computed between the reference and secondary epipolar images. For every match, the difference between their rows is computed. The secondary epipolar grid is then corrected so that row differences are null on average. Once aligned epipolar grids have been obtained, stereo images can be resampled in a epipolar geometry, which minimizes the errors between SIFT points.

Alongside epipolar grids, the disparity to altitude ratio r_{alt} can be computed, which corresponds to the altimetric shift resulting from a shift of 1 in disparity. In other words, if a pixel has a disparity of 0, then its altitude will be Z_{coarse}, and if it has a disparity of 1, then its altitude will be $Z_{coarse} + r_{alt}$. r_{alt} is also called the altimetric ratio, and provides an estimation of the altimetric resolution of the final DSM. The altimetric ratio varies along an image, but its variations are small enough so that we can safely approximate it by a constant. We will make extensive use of the altimetric ratio r_{alt} in Chapter 6 to compare the uncertainty of DSMs with different altimetric resolutions.

Once the input images in sensor geometry have been resampled into epipolar geometry, the disparity of every pixel can be computed using dense matching algorithms.

1.3.3 Stereo Matching

Dense matching refers to the pairing of every pixel between two images. It differs from sparse matching, where only a restricted set of pixels must be matched in both images. Sparse matching is used for registering the different channels of an image if they have been acquired separately, for instance, using push-broom sensors. For constructing high-resolution DSM, dense matching is necessary, as we are interested in estimating the disparity at a pixel scale.

As stereo matching is an important problem in computer vision, multiple algorithms have been proposed to compute a dense disparity map $\mathcal{D}$ mapping each pixel $(row,\ col)$ to its disparity d. Dense matching algorithms can be broadly classified into two categories: classical approaches following the steps outlined by Scharstein *et al.* [Scharstein et al., 2001], and deep-learning based methods [Laga et al., 2022].

Remark: In the domain of stereo matching, the reference image is often referred to as the *left* image, and the secondary image is the *right* image.

Recently, deep-learning methods have greatly improved the results of stereo matching algorithms for both remote sensing applications [Chebbi et al., 2023] and more general applications [Tosi et al., 2024] such as robotics, autonomous cars or augmented reality. Best results on famous benchmarks have been obtained using 2D an 3D convolutional neural networks and end-to-end deep-learning approaches [Guo et al., 2024, Liu et al., 2024]. Those models usually undergo a supervised training, with many datasets being available for stereo processing, especially for autonomous cars [Geiger et al., 2012, Geiger et al., 2013].

There are fewer open satellite datasets due to the costs and copyrights of satellite images, even though more datasets tends to be released [Bosch et al., 2018, Le Saux et al., 2019, Huang et al., 2022]. Obtaining ground truth data for satellite imagery is also challenging: as classical approaches, such as structured light, cannot be used for producing ground truth disparity. Instead, costly airborne campaigns are carried-out, and the acquired elevation data must be converted into disparity maps, which is a complex process [Cournet et al., 2020]. Additionally, large models often face generalization challenges, particularly when applied to images that differ from their training datasets. It is especially true in the case of satellite imagery [Marí et al., 2022, Jiang et al., 2024a], as there is a large variety of landscapes, sensors, and acquisitions angles to consider. Furthermore, the radiometry can change depending on the time of the day when the acquisition occurred, and landscapes change can also vary greatly between two seasons. All those factors make the training of a performing and generalizable network for satellite imagery quite challenging. Non-supervised algorithms could avoid some of those drawbacks, but they do not present the same performances as supervised networks.

Even though deep stereo algorithms produce exciting results, we will not focus on deep end-to-end algorithms in this thesis; instead, we will restrict ourselves to so-called *classical* methods used in satellite stereo pipelines. Those methods have the advantage of relying on an extensive literature on the subject. More importantly, those methods are explainable as they do not have a "black-box" structure similar to that of end-to-end networks. Their interpretability is crucial when modeling and propagating their uncertainty, which is particularly relevant in the context of this thesis.

Classical approaches usually encompass the following steps[Scharstein et al., 2001]:

- matching cost computation
- cost aggregation
- disparity optimization and computation
- disparity refinement

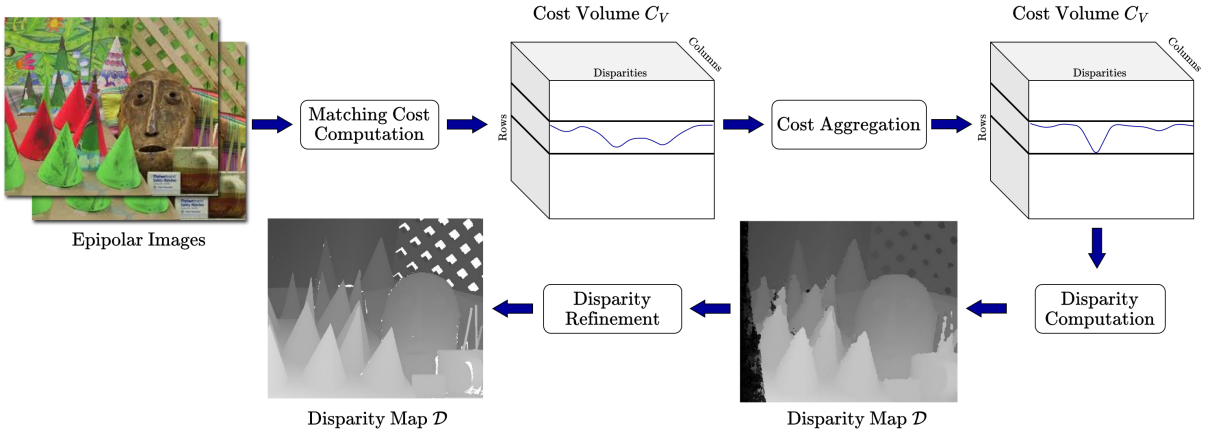


Figure 1.11: The different steps of classical dense stereo matching algorithms.

Figure 1.11 illustrates those different steps. We will detail each step into more details in the following sections. We consider the case of the Pandora correlator developed at CNES (`https://github.com/CNES/Pandora`), which is used by the CARS pipeline for the dense matching step.

Matching Cost Computation

To determine pairs of matching pixels, we first start by measuring the cost of matching two pixels together. Pixels with similar values and similar surroundings will have a low matching cost, while dissimilar pixels will have a high matching cost. Each matching cost value is evaluated using a cost function, which is a mapping f from subsets of the left and right images to $\mathbb{R}$. A cost function f measures the dissimilarity between the two subsets of the left and right images. Example 1 provides different instances of cost functions.

Example 1: Simple instances of cost functions include the Sum of Absolute Differences (SAD), the Zero Normalized Crossed Correlation (ZNCC) [Hannah, 1994], the CENSUS transform [Zabih and Woodfill, 1994], and MC-CNN [Žbontar and LeCun, 2016], which will be detailed in the following paragraphs.

Given two windows W_L and W_R from the left and right images, the SAD cost function is defined as follows:

$$f_{SAD}(W_L, W_R) = \sum_i \sum_j |W_L(i,j) - W_R(i,j)| \tag{1.5}$$

Low f_{SAD} values indicate that the windows are similar, while high values indicate noticeable differences. A f_{SAD} value of 0 indicates that the windows are identical. This is one of the simplest cost functions, and will be used in Chapter 4 to didactically illustrate how uncertainty models can be propagated throughout a cost function. However,

this cost function is not usually employed in practice as it is based on intensity differences, and is thus very sensitive to radiometric changes. It can however be a fast and easy way to have a first estimate of similarities between multiple patches. The SAD can also be used for motion estimation and image/video compression [Richardson, 2006].

The ZNCC cost function is defined as the correlation coefficient between both images:

$$f_{ZNCC}(W_L, W_R) = \sum_i \sum_j \frac{(W_L(i,j) - \tilde{W}_L)(W_R(i,j) - \tilde{W}_R)}{\sigma_L \sigma_R} \tag{1.6}$$

where $\tilde{W}$ refers to the mean value of a window, and σ its standard deviation. Negatively correlated windows would present a ZNCC value of -1 and positively correlated windows present a ZNCC value of 1. Contrary to the SAD cost function, matching windows will be indicated by a high value of the ZNCC. It is thus not *strictly* a cost function but rather a similarity function. It is not a problem, as multiplying f_{ZNCC} by -1 will transform it into a cost function. Another formulation could be to say that the ZNCC is a *maxitive* cost function, in the sense where potential matches are found by searching for its maximum. Conversely, the SAD is a *minitive* cost function in the sense where potential matches are indicated by a minimal cost. The ZNCC cost function performs well in homogeneous areas when computed over large windows. However, in an urban setting, it struggles to correctly estimate buildings boundaries.

The CENSUS cost function needs to be a bit more detailed. For a squared window W with a side of $2n+1$ pixels, we first compare the value of each pixel of the window with the center pixel. This gives a binary string where 1 indicates that the value of the pixel is superior to that of the center pixel. For instance, if we consider the two 3×3 following windows W_L, W_R:

$$\begin{bmatrix} 155 & 133 & 97 \\ 80 & 110 & 132 \\ 100 & 102 & 120 \end{bmatrix} \qquad \begin{bmatrix} 175 & 153 & 133 \\ 100 & 130 & 152 \\ 120 & 135 & 125 \end{bmatrix}$$

Then comparing each of their pixels to the center of the windows will yield the following binary strings (expressed here as matrices):

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

The cost function f_{CENSUS} is finally obtained by taking the Hamming distance (*i.e.* the number of different bits between those two strings):

$$f_{CENSUS}(W_L, W_R) = 3$$

The CENSUS cost function compares relative intensity variations, it is thus less sensitive to variations of intensities between images, such as a change of exposure for instance. Similarly to the SAD, two similar patches will tend to have a low value.

The MC-CNN cost function [Žbontar and LeCun, 2016] is using a convolutional neural network architecture to measure the similarity between patches. It was trained on 11×11 patches from stereo images from the Kitti [Geiger et al., 2013, Menze and Geiger, 2015] and Middlebury [Scharstein et al., 2001, Scharstein and Szeliski, 2003, Hirschmuller and Scharstein, 2007, Scharstein and Pal, 2007, Scharstein et al., 2014] datasets. They first trained a siamese neural network to compute a vector of features for each patch, and then compute the dot product between both vectors to obtain a similarity measure. MC-CNN is thus a maxitive cost function. Figure 1.12 details the architecture of the network. Although it has been trained on stereo images of autonomous cars (Kitti) and stereo images of toys (Middlebury), it generalizes well to satellite images [Defonte et al., 2021].

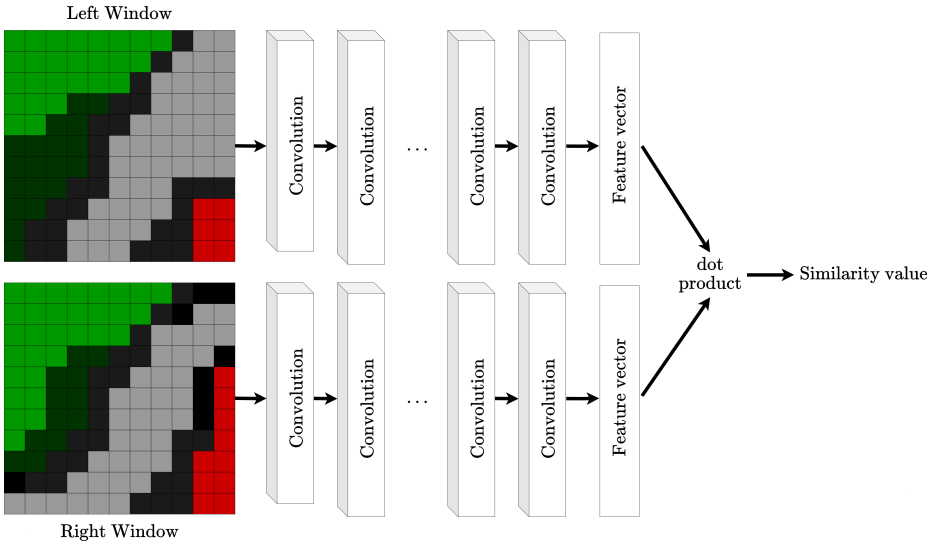


Figure 1.12: MC-CNN architecture

Matching costs are evaluated for every potential pairs of pixels whose corresponding disparity lies in a given disparity range. Matching cost values are stored in an array of data, called *cost volume* (Figure 1.13). The term volume is used as a matching cost value is determined by three coordinates in the cost volume: the row and column (row, col) of

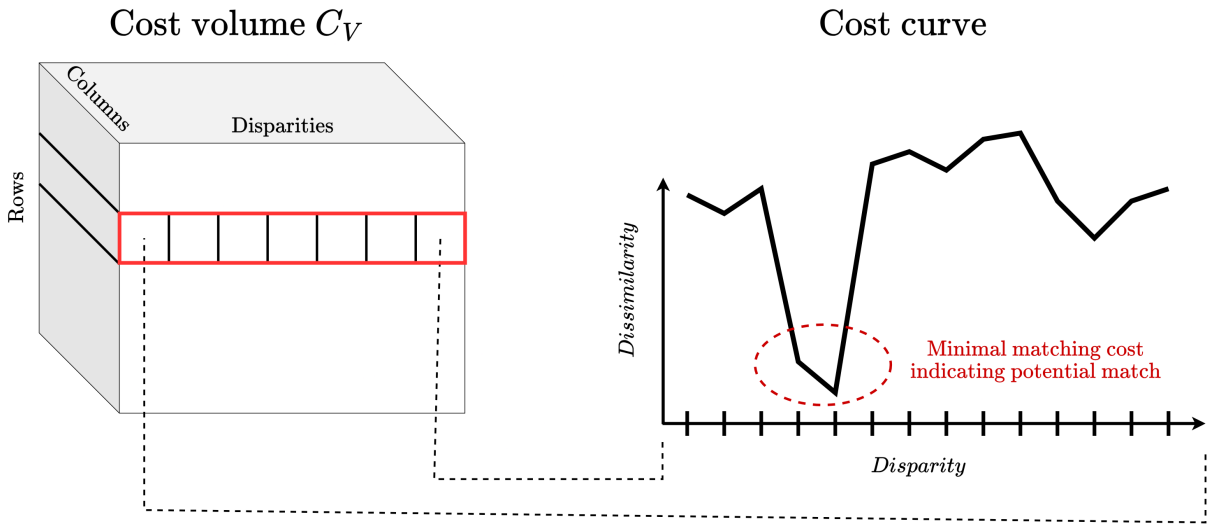


Figure 1.13: Matching cost volume and one of its cost curve

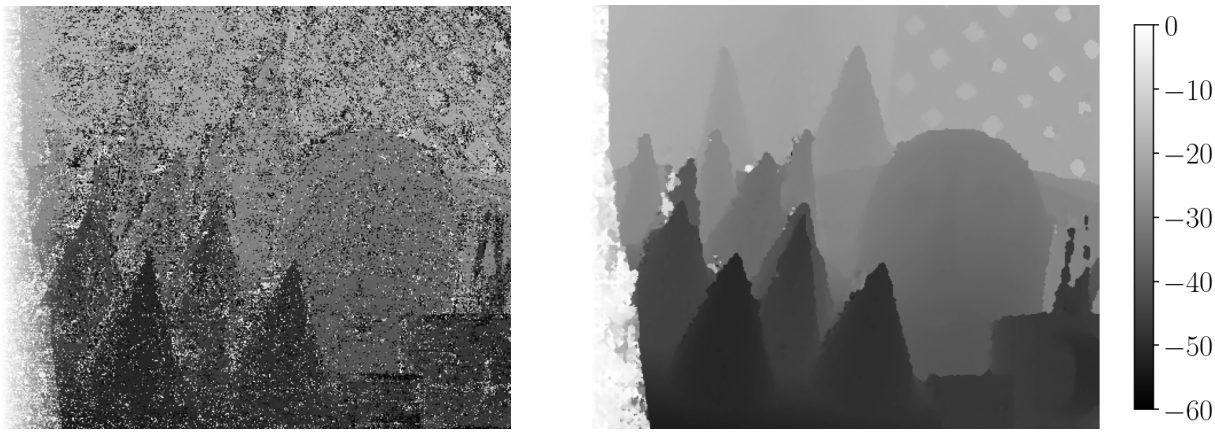
the left pixel as well as the considered disparity d:

$$C_V(row,\ col,\ d) = f(W_L(row,\ col), W_R(row,\ col + d)) \tag{1.7}$$

where $W_L(row,\ col)$ is a window centered on the pixel at coordinates $(row,\ col)$ in the left image, and $W_R(row,\ col+d)$ is a window centered on the pixel at $(row,\ col+d)$ in the right image. Some padding is usually added to the images to avoid problems near borders where the column $col+d$ would not be defined. Given a pixel in the left image $(row,\ col)$, matching cost values for every considered disparity form what is called a *cost curve*. In theory, the correct disparity for a match is determined by finding the minimum of the cost curve (for minitive cost functions such as SAD or CENSUS). Figure 1.13 represents a cost volume, and one of its cost curves where potential matches have been highlighted. In the rest of this thesis, we will consider that a cost function is always minitive unless specified otherwise. In practice, directly estimating the disparities from the cost volume is not efficient, as we only consider local information. The resulting disparity map is often very noisy, as can be seen in Figure 1.14(a). In order to reduce this noise, one solution is to aggregate the costs of neighboring pixels in the cost volume.

Cost Aggregation

The usage of windows in the calculation of the cost function allows taking into consideration the surrounding of pixels to better measure their similarity. However, window based approaches, which are mostly square-shaped, also present the disadvantage of struggling to correctly identify matches near object borders [Hirschmüller et al., 2002]. This phenomenon is referred to as the adherence effect, as represented in Figure 1.15. In Figure 1.15(b), we can see that the considered pixel is at the border of an object, with the left window W_L spanning over both part of the border. In Figure 1.15(d), we can see that



(a) Disparity with local cost only (b) Disparity with SGM regularization

Figure 1.14: Disparity map without and with SGM regularization.

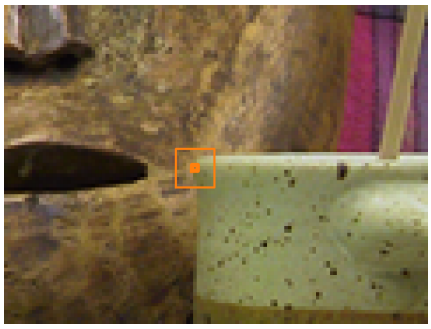
the minimum of the cost curve (around disparity -40) is not exactly located at the true disparity. This is in part due to the fact that W_L contains pixels from two objects with different disparities, which influences the matching cost values. A low matching cost thus does not necessarily mean that the center pixels constitute a match. The term "adherence" is used as the matching windows tend to falsely estimate the shift in disparity near objects' border, as if it adhered to the object. Other work have been proposing to use a spatial weighting [Kuk-Jin Yoon and In-So Kweon, 2005], segmentation [Tombari et al., 2007], windows with different shapes [Ke Zhang et al., 2009, Buades and Facciolo, 2015] to solve this problem.

After computing the matching cost, information from neighboring pixels can be incorporated in the cost volume. A first approach is to aggregate different parts of the cost volume. Usually, costs of pixels belonging to the same objects are aggregated using different methods for segmentation [Ke Zhang et al., 2009, Ji et al., 2021]. This step is not always included in algorithms, as we can also consider global information directly when determining the disparity map.

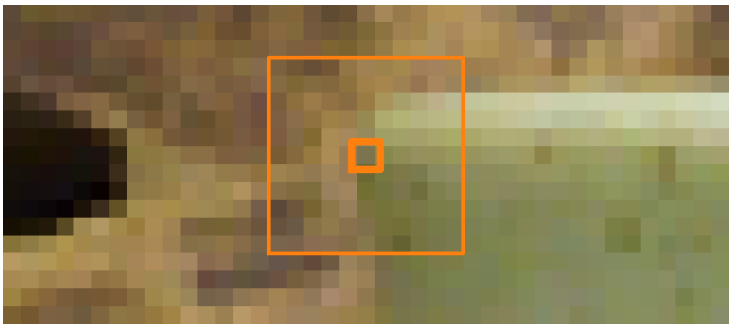
Disparity Optimization and Computation

Computing the disparity map from the cost volume can be done in several ways. So-called local methods apply a direct *winner-takes-all* strategy, where the arg min of every cost curve is kept as the selected disparity. On the other hand, global methods use the information contained in the cost volume to solve an optimization problem, where the objective is to compute the disparity map $\mathcal{D}$ minimizing an energy function expressed as follows:

$$E(\mathcal{D}) = E_{data}(\mathcal{D}) + \lambda E_{smooth}(\mathcal{D}) \tag{1.8}$$

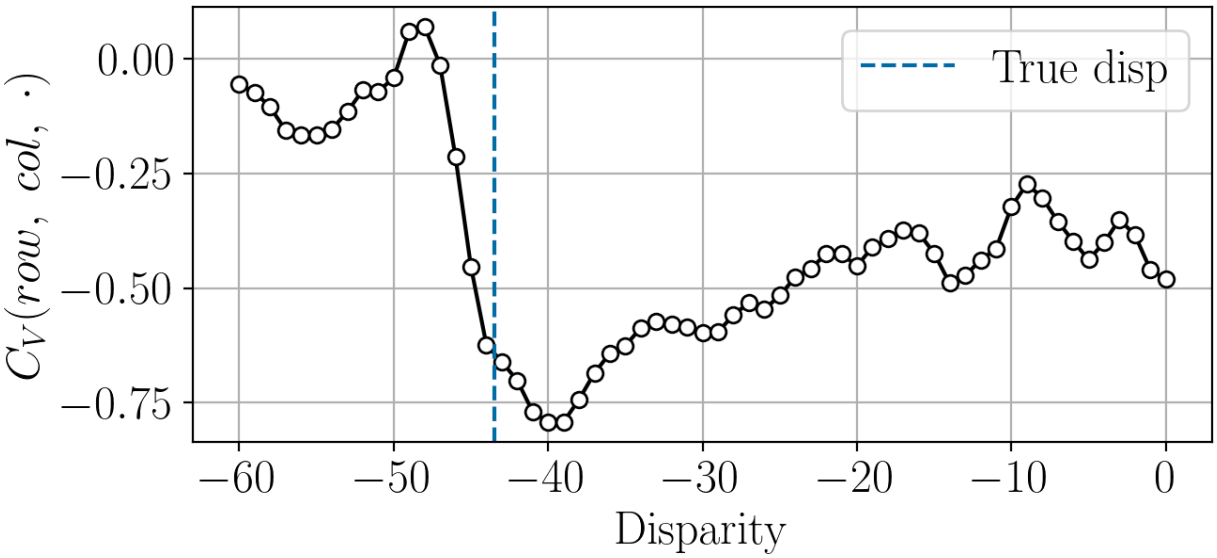


(a) Left image with W_L in orange



(b) Zoom over W_L, in orange

(c) Right image. Considered windows in orange, and W_R for the true disparity in blue



(d) Corresponding minitive cost curve "$-f_{ZNCC}$"

Figure 1.15: Example of the adherence effect. (a) and (b) present the left window W_L at the border of an object. (c) presents all the windows from the right image in the considered disparity range, that will be compared to W_L. The window corresponding to the true disparity appears in blue. (d) details the corresponding cost curve, where the true disparity does not correspond to the minimum of the cost curve.

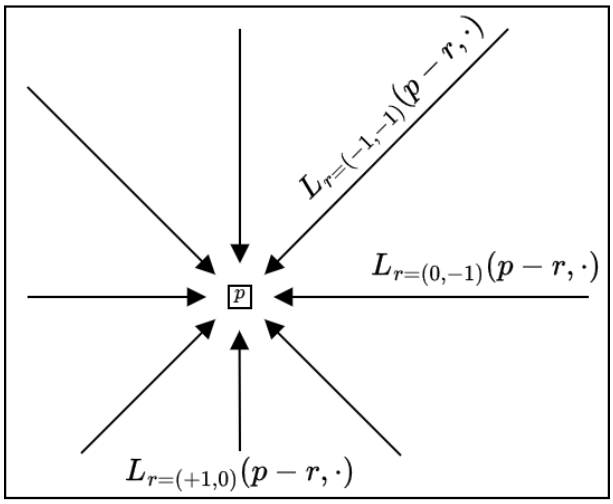


Figure 1.16: SGM regularization with 8 directions

where λ is a scalar for tuning the importance of the regularization term E_{smooth}. Usually, the data term is directly computed from the cost volume as:

$$E_{data}(\mathcal{D}) = \sum_{row,\ col} C_V(row,\ col,\ \mathcal{D}(row,\ col)) \tag{1.9}$$

The regularization term E_{smooth} can take numerous forms, usually measuring if the neighboring disparities possess similar values [Scharstein et al., 2001]. Then a local minimum for this energy is found using various methods, such as Markov Random Fields [Boykov et al., 1998, Sun et al., 2003], graph cuts [Kolmogorov and Zabih, 2001] or minimum spanning trees [Zureiki et al., 2008, Qingxiong Yang, 2012]. Those algorithms improve performance in comparison to local methods, but can be computationally expensive.

One of the most popular methods is called Semi-Global Matching (SGM) [Hirschmüller, 2005]: it aims at incorporating regularization constraints to the cost volume similarly to global methods, while being processed with relative low computational cost like local methods. It is, in a way, in-between local and global methods. This is the method used in the CARS pipeline for experiments in this thesis. In a few words, for each pixel, the SGM algorithm computes a regularized cost volume, which is based on the regular cost volume (the E_{data} term), and adds cost penalties to the disparities differing from those of their neighboring pixels (the E_{smooth} term). Although the formulation of the SGM algorithm can be expressed in a few equations, understanding its inner workings is more complex. We will first present its mathematical formulation in Equations (1.10) and (1.11), and use Figure 1.18 to illustrate its effect on (a portion of) a cost curve. Figure 1.17 presents the effect of SGM regularization on different cost curves and Figure 1.14 presents the effect of SGM regularization on the final disparity map. Formally, for every pixel $p = (row,\ col)$, the cost volume is explored in multiple directions, as in Figure 1.16.

The cost volume is regularized for each direction r to take into account the best disparity d along that direction. A direction can be, for instance, $r = (0,\ 1)$, meaning that we will look at the same row and travel to the right of the image when browsing direction r. Given two positive scalars $P_1 < P_2$, the regularized cost L_r along direction r and at disparity d is expressed with the following recursive formulation:

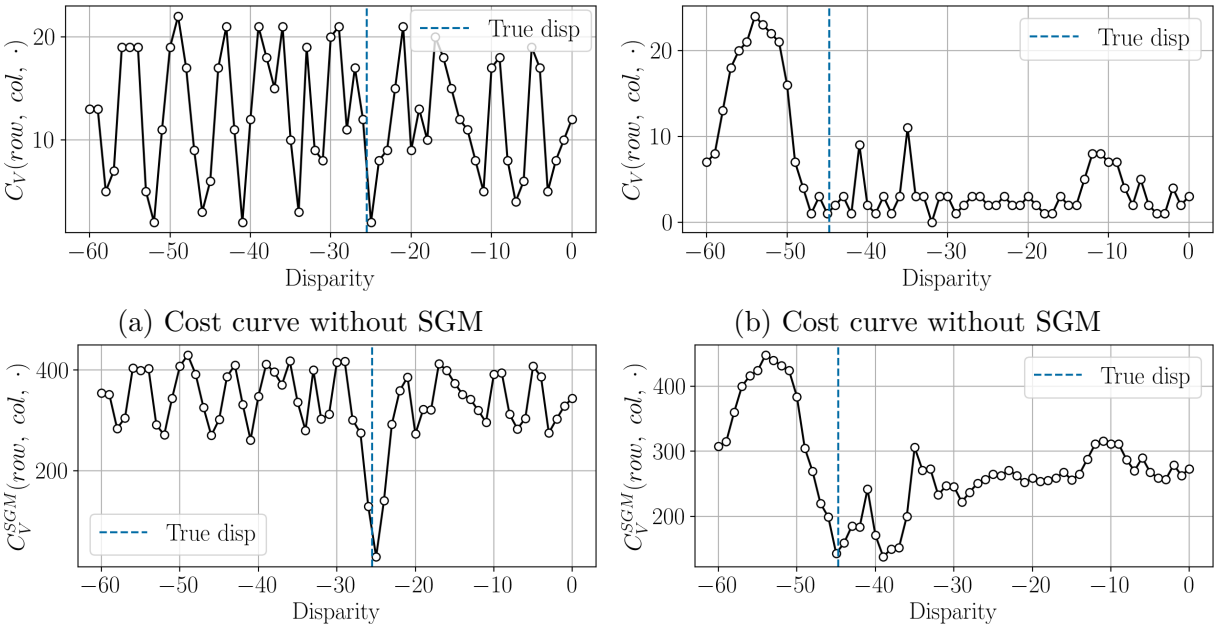
$$L_r(p, d) = C_V(p, d) + \min_{\delta} \left(L_r(p - r,\ \delta) + R(d,\ \delta)\right) \tag{1.10}$$

where δ is a dummy variable to explore the disparity range, and where $R(d,\ \delta)$ equals:

$$R(d,\ \delta) = P_1 \cdot \mathbb{1}(|d - \delta| = 1) + P_2 \cdot \mathbb{1}(|d - \delta| \geqslant 2) - \min_{k} L_r(p - r,\ k) \tag{1.11}$$

Here, $\mathbb{1}$ is the indicator function. The first term of Equation (1.10) is the cost volume, which can be compared to E_{data} in Equation (1.9). The second term is similar to the regularization term E_{smooth}. This term can be seen as the regularized cost $L_r(p - r,\ d)$ from the previous pixel in direction r, but shifted of P_1 for neighboring disparities $d \pm 1$, and of P_2 for further disparities. The term $\min_k(L_r(p - r,\ k))$ prevents $L_r(p, \cdot)$ from diverging to very high values by ensuring that the minimum of $L_r(p - r,\ \delta) - \min_k(L_r(p - r,\ k))$ always equals 0. Figure 1.18 presents $L_r(p - r,\ \delta) + R(d,\ \delta)$ for two different disparities d and $d+1$ in the blue frame. The minimum of each curve $L_r(p - r,\ \delta) + R(d,\ \delta)$ (blue arrows in the figure) is then added to the matching cost $C_V(p, d)$ to obtain the final regularized cost $L_r(p, d)$. Note that in Figure 1.18, the minimum of $L_r(p - r,\ \delta) + R(d + 1,\ \delta)$ equals 0, thus C_V is unchanged for this disparity. This is because $d + 1$ is the minimum of $L_r(p - r,\ \cdot)$. In a way, the regularized matching cost will be a mixture between $C_V(p, \cdot)$ and $L_r(p - r,\ \cdot)$. Indeed, the regularized cost $L_r(p - r, \cdot)$ indicates that disparity $d + 1$ seems likely, while disparity d is unlikely. On the other hand, the matching cost $C_V(p, \cdot)$ indicates that both disparities d and $d+1$ are likely. The final regularized cost $L_r(p - r, \cdot)$ takes into account that both $C_V(p, \cdot)$ and $L_r(p - r, \cdot)$ agree that $d + 1$ is likely, but that there is a disagreement on d, and thus slightly increases the regularized cost $L_r(p, d)$. Disparities that appear unlikely to both $C_V(p, \cdot)$ and $L_r(p - r, \cdot)$ result in a larger increase in the regularized cost $L_r(p, \cdot)$.

> **Remark:** For the sake of the argument, let's assume that $P_1 = P_2$ for now. The value added to $C_V(p,\ d)$ in (1.10) will be lower than P_2 only if $(L_r(p - r,\ d) - \min_k L_r(p - r,\ k))$ is less than P_2. This means that we will always add a penalty of P_2 to the cost C_V, except if $L_r(p - r,\ d)$ is less than P_2 away from its minimum. P_2 thus represents the penalty that must be overcome in $L_r(p - r,\ d)$ in order to not penalize $C_V(p, d)$, or slightly penalize $C_V(p, d)$.
>
> If $P_1 < P_2$, then we can draw similar conclusions, except that we also accept to



(a) Cost curve without SGM

(b) Cost curve without SGM

(c) Cost curve from (a) with SGM

(d) Cost curve from (b) with SGM

Figure 1.17: Different cost curves, with and without SGM regularization

> reduce the penalty to $C_V(p, d)$ if a neighboring disparity, *i.e.* $d \pm 1$ is less than P_1 away from the minimum.

Formulation of Equation (1.10) is recursive, and thus must be initialized. The regularization curves thus begin at the borders of the image and their value is, by convention, 0 when undefined. For instance, with $r = (0, 1)$:

$$L_r((row, 0), d) = C_V(row, 0, d)$$

When all regularized cost curves L_r have been computed, they are summed to obtain the regularized cost volume C_V^{SGM}:

$$C_V^{SGM}(p, d) = \sum_r L_r(p, d) \tag{1.12}$$

The cost volume C_V^{SGM} contains the regularized cost for every considered direction. The number of direction used in our experiments is 8 as in Figure 1.16, but more directions can be considered in order to consider a more global coverage. Since the original paper [Hirschmüller, 2005], different variations of the SGM algorithm have been proposed. For instance, using different regularization paths [Facciolo et al., 2015] or a different strategy for the aggregation of costs [Poggi and Mattoccia, 2016].

In order to ensure smooth surfaces, SGM penalties P_1 and P_2 must be high enough. This however leads to a reluctance to detect discontinuities in the disparity map. In-

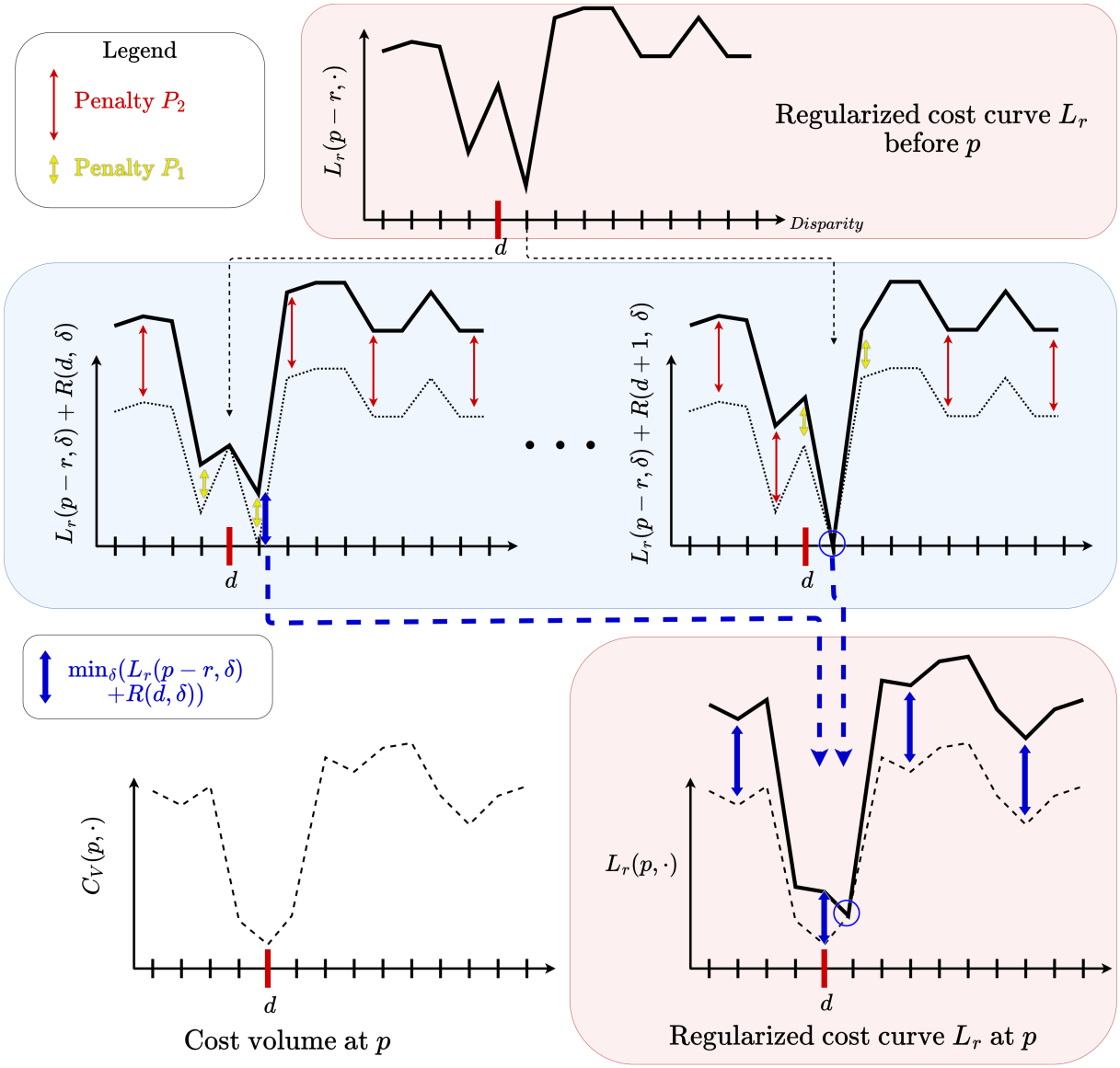


Figure 1.18: Schematic explanation of the SGM algorithm in a single direction r. Top: regularized cost L_r at $p-r$. In the blue frame $L_r(p-r,\ \delta)+R(d,\ \delta)$ for two consecutive disparities d and $d+1$ ($L_r - \min_k L_r$ appears in dotted line for clarity). Penalty P_1 appears in yellow, penalty P_2 appears in red, the minimum of each curve is denoted by a blue arrow. Bottom left: cost volume C_V at p. Bottom right: regularized cost L_r at p where the minima of *all* $L_r(p-r,\ \delta)+R(d,\ \delta)$ have been added

deed, SGM penalizes disparity changes, therefore strong variations of disparity are badly reconstructed. For instance near the border of a building, the term E_{smooth} tends to be predominant, leading to rounded borders and soft edges of buildings, instead of the expected sharp disparity variations. This is even reinforced with the window adherence problem presented in 1.15. Figure 1.19 presents a comparison of two DSM: one obtained directly from LiDAR HD data [Monnet, 2023], and the other using the CARS pipeline with the CENSUS cost function over a 5×5 window and with SGM regularization. We can see on this figure that the border of the buildings obtained using SGM regularization are not as sharp and precise as the ground truth provided by the LiDAR DSM. To provide a solution to this problem, one might limit the SGM regularization to pixels from the same object by using a segmentation [Dumas et al., 2022]. This method relies on the quality of the segmentation method used, and can become quite costly when segmenting large images. It has not been considered in the context of this thesis.

Once the regularized cost volume C_V^{SGM} has been computed, the classical approach in the literature is to apply a *winner-takes-all* strategy to determine the disparity map $\mathcal{D}$:

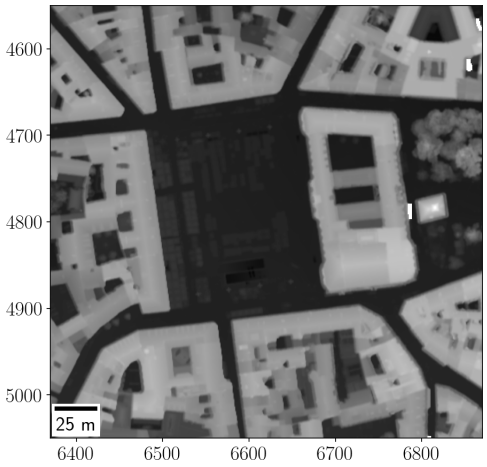
$$\mathcal{D}(row,\ col) = \arg\min_{d} C_V^{SGM}(row, col, d) \tag{1.13}$$

Disparity Refinement

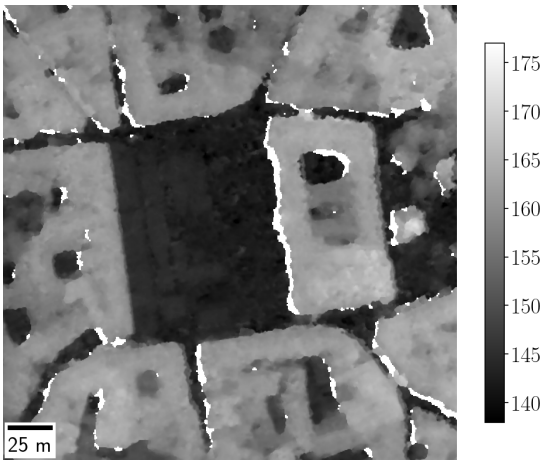
Once the disparity map has been computed, it is usually post-processed to remove artifacts and improve disparity resolution. It is common to add a sub-pixel refinement step, where a non-integer disparity is interpolated around the selected disparity. The main idea is to interpolate a model through the selected disparity $d = \arg\min_\delta C_V(row,\ col,\ \delta)$ and its two direct neighbors. The refined disparity d_{interp} is then defined as the arg min of this interpolation model. Figure 1.20 presents examples of interpolated disparities from [Haller et al., 2010], mainly a "V"-like shape as in 1.20(a) and a parabola as in 1.20(b). Carrying out sub-pixel refinement suggests we assume the algorithm can attain a significant level of accuracy, which is debatable (see Section 1.4.2), and that the cost function is sufficiently sampled to be correctly interpolated.

The disparity map can also be filtered in order to remove local outliers. Classical strategies include the use of a mean or median filter, for instance. There are more advanced filtering methods, such as bilateral filtering [Tomasi and Manduchi, 1998] which performs a weighted average, where weights depend on the proximity to other pixels both in the spatial and spectral domains.

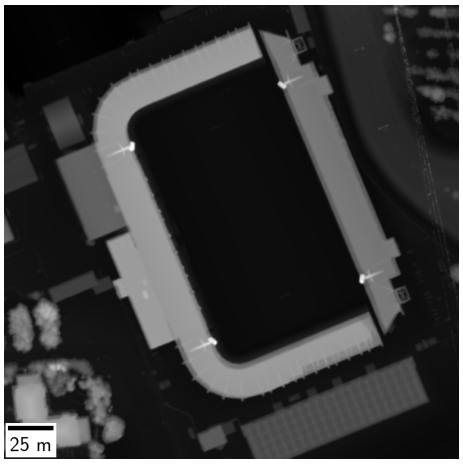
There exist strategies allowing to detect dubious matches. For instance, cross-checking [Fua, 1991] verifies if the disparity selected would stay consistent with the one computed



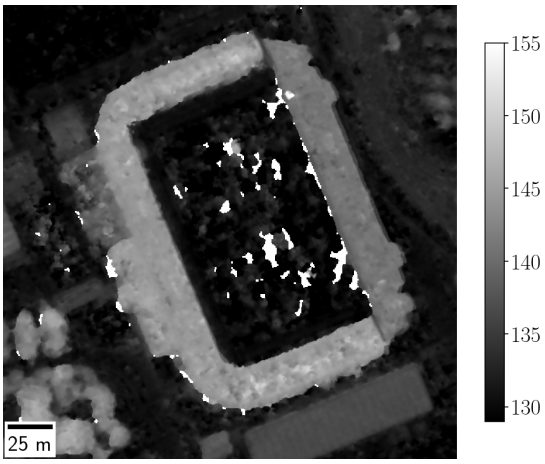
(a) DSM from LiDAR HD data, Place du Capitole



(b) CARS DSM from Pléiades images, Place du Capitole

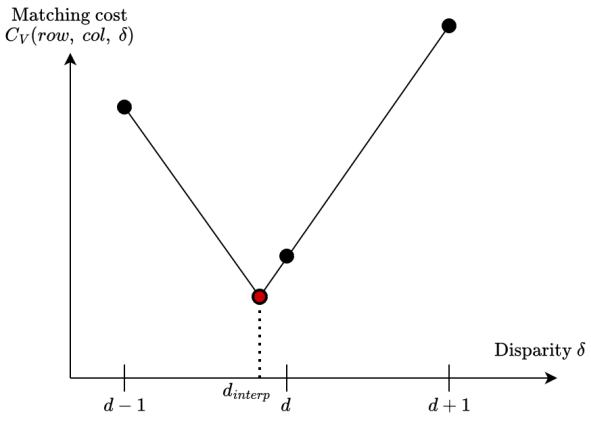


(c) DSM from LiDAR HD data, Ernest Wallon stadium

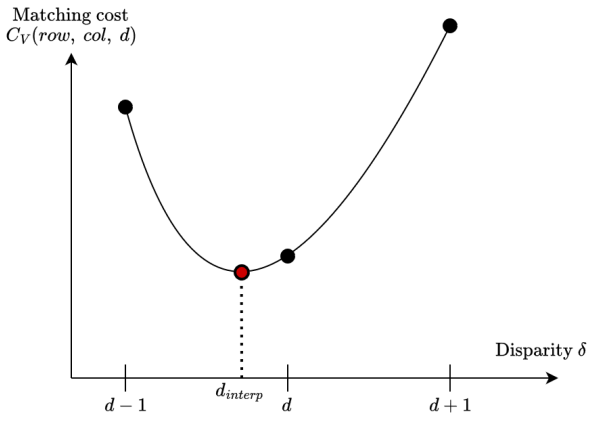


(d) CARS DSM from Pléiades images, Ernest Wallon stadium

Figure 1.19: Different DSMs over Toulouse, France. DSMs were obtained by rasterizing LiDAR HD data or by processing Pléiades stereo images with the CARS pipeline. The CARS pipeline uses a 5×5 CENSUS cost function and SGM regularization for stereo reconstruction. ©CNES 2017, Distribution AIRBUS DS



(a) V-fit sub-pixel refinement



(b) Parabola sub-pixel refinement

Figure 1.20: Different methods for refining the disparity $d = \arg\min_\delta C_V(row,\ col,\ \delta)$

by reversing the roles of the images: the reference image becomes the secondary image. We first start by computing a secondary cost volume C'_V by reversing the role of the images. We then obtain two disparity maps $\mathcal{D}$ and $\mathcal{D}'$, and check that they are consistent, as in theory we would that $\mathcal{D} = -\mathcal{D}'$. A pixel $(row,\ col)$ is considered consistent across disparity maps if it verifies:

$$|\mathcal{D}(row, col) + \mathcal{D}'(row, col + \mathcal{D}(row, col))| \leqslant \tau \tag{1.14}$$

where τ is a consistency threshold usually set to 1. For errors that do not satisfy this consistency check, it has been proposed in [Hirschmuller, 2008] to differentiate between mismatched pixels and occluded pixels. Mismatch pixels are pixels for which there exists a correct match, whereas occluded pixels are pixels that are visible in an image but masked by an object in the other. The difference is determined as follows: if there is a disparity $d \in \mathcal{D}$ such that $\mathcal{D}'(row, col + d) = -d$ then it is considered a mismatch. Otherwise, the pixel $(row,\ col)$ is occluded. Occluded regions of an image can be filled by interpolation with the closest valid disparities, or left as *nodata*.

> **Remark:** Equation (1.14) requires the computation of a second cost volume, which effectively doubles the processing time. As dense stereo matching is also the longest part of a photogrammetry pipeline, one might be reluctant to carry out a cross-checking step. However, we can make the following observation: the cost volume contains the dissimilarity between every considered pair of pixels in the disparity range, so the cost of every considered match in the first cost volume is also present in the second cost volume. In theory, for every pixel $(row,\ col)$ and for every disparity d in the (reference) disparity range, it holds [Ernst and Hirschmüller, 2008]:
>
> $$C_V(row,\ col,\ d) = C'_V(row,\ col + d, -d) \tag{1.15}$$
>
> Equation (1.15) holds only for cost volumes obtained after the matching cost step. However, when SGM regularization or another aggregation cost method modifies cost volumes, then there can be differences between $C_V(row,\ col,\ d)$ and $C'_V(row,\ col + d, -d)$. In the case of SGM regularization, series of tests show that the differences between cost volumes are small and marginally modify the disparity maps. Using (1.15) then allow obtaining both cost volumes by only computing one and re-indexing it to obtain the other.

Once every step of the dense matching algorithm have been carried out, the disparity map obtained is used to create pairs of lines of sight to triangulate 3D point clouds.

1.3.4 Triangulation

The disparity maps allow deducing elevation information contained in the shift of objects in both image spaces. This information is computed during the triangulation step of the stereo pipeline.

The type of optical sensor used determines the resolution of the disparity map, and, therefore, the density of the resulting point cloud. It will therefore influence the planimetric resolution of the final DSM. We usually consider that the planimetric resolution of the DSM can be similar to the resolution of the sensors. On the other hand, the altimetric resolution of the DSM is determined by the altitude and positions of the satellites [Qin, 2019]. This position can be characterized by the Base-to-Height ratio (B/H) ratio, as in Figure 1.21. This ratio is computed by dividing the distance separating the stereo acquisitions by the altitude of the satellite. It indicates the angle formed between the lines of sight originating from the satellites towards an object of the scene. A high B/H allows for high elevation accuracy, but possesses more occluded regions (for instance a narrow street between two high buildings), and conversely for a low B/H [Delon and Rougé, 2007]. In natural landscapes, stereo acquisitions present high B/H ratio in order to have a high altimetric accuracy, while the B/H ratio is smaller in urban landscapes to limit the occlusions. In our experiments, the B/H ratio for stereo acquisitions varies between 0.1 and 0.4. The CO3D mission will use B/H ration between 0.2 and 0.3. The B/H also partly determines the value of the altimetric ratio r_{alt} presented in Section 1.3.2.

The quality of the final DSM is also influenced by the solar angles [Qin, 2019], with the zenith angle being as small as possible to limit the projected shadows over the scene. Sun-synchronous satellites, such as Pléiades or CO3D, benefit from advantageous sun angles in order to favor good images acquisitions and good stereo reconstruction.

The computed disparity map allows determining the position of 3D points. When working with pinhole camera models, the depth z of a pixel is computed using the following formula:

$$z = \frac{Bf}{d} \tag{1.16}$$

where B is the baseline between cameras, f is the focal length of the camera, and d is the disparity of a pixel. This formula can be found using optical geometry [Bolles et al., 1987], and illustrates the fact that pixels closer to the camera present a bigger position shift in between images. As mentioned previously, the pinhole camera model is not valid in the case of satellite imagery, and we instead use other sensor models (see Section 1.2.3). Equation (1.16) thus cannot be used as such to provide accurate results.

Instead, the disparity is used to determine pairs of lines of sight pointing to the same

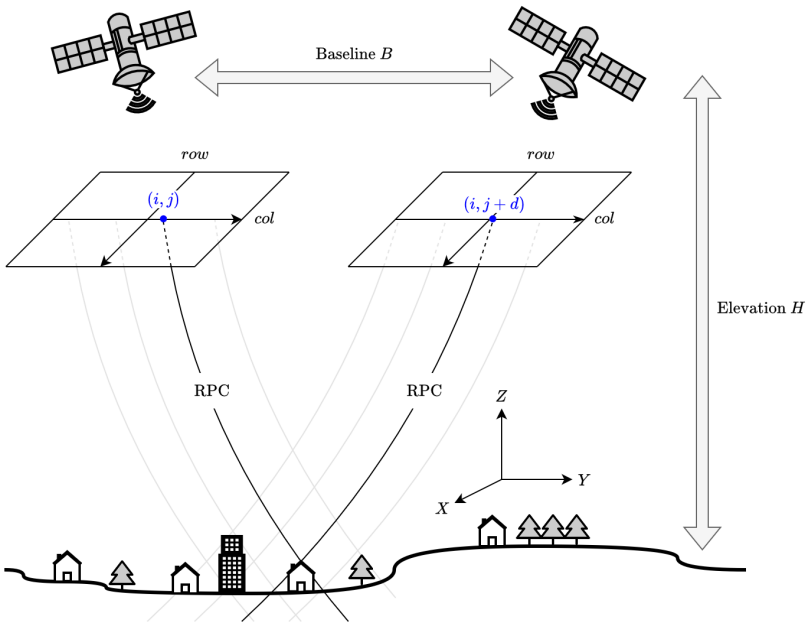


Figure 1.21: Triangulation of the position of a point using RPC models. The distance B between satellites as well as the global elevation H can be used to determine the angle between satellites.

object. Using RPC models, we can compute the 3D coordinates of every intersection of pairs of lines of sight. To do so, we first consider the RPC models RPC_1, RPC_2 of two stereo images defined in Section 1.2.3. We consider as well their epipolar grids g_{e1}, g_{e2} from Equation (1.4). For every pixel $(row_e,\ col_e)$ from the reference epipolar image, whose disparity is d, the 3D point (X, Y, Z) represented by $(row_e,\ col_e)$ is the point verifying the following equations:

$$(X, Y, Z) = \text{RPC}_1^{-1}(g_{e_1}(row_e,\ col_e),\ Z) \tag{1.17}$$

$$(X, Y, Z) = \text{RPC}_2^{-1}(g_{e_2}(row_e,\ col_e + d),\ Z) \tag{1.18}$$

If the lines of sight intersect, then Z is found by solving the following equation:

$$\text{RPC}_1^{-1}(g_{e_1}(row_e,\ col_e),\ Z) = \text{RPC}_2^{-1}(g_{e_2}(row_e,\ col_e + d),\ Z) \tag{1.19}$$

Knowing Z and $g_{e_1}(row_e,\ col_e)$, Equation (1.17) provides the X and Y coordinates as well.

A known problem is that lines of sight rarely intersect, even if we consider the right disparity, because the bundle adjustment is not perfect. By approximating lines of sight by a starting point P and a direction vector $\overrightarrow{V}$, we can instead define the coordinates (X, Y, Z) as the point minimizing its distance to both lines. Approximating RPCs by segments is valid on the limited range of considered altitudes. The exact formula for

computing (X, Y, Z) is provided in [Delvit et al., 2006]:

$$(X, Y, Z) = \left[Id - V_1 V_1^T + Id - V_2 V_2^T\right]^{-1} \left[(Id - V_1 V_1^T)P_1 + (Id - V_2 V_2^T)P_2\right] \quad (1.20)$$

where Id is the identity matrix, and V^T is the transposed vector V. The error due to enforcing RPCs intersection is however small (a few centimeters) in comparison to the errors occurring in the disparity map. In this thesis, we mainly focus on processing the disparity errors and do not consider the approximation made during the lines of sight intersection.

Triangulating every point using the disparity map leads to a 3D point cloud. Because we know from which pixel each 3D point originates from, we can associate to every 3D point additional information such as:

- The color of the reference pixel (if provided). The point cloud is thus a colored point cloud.
- Confidence measures computed during the dense matching step, that will be presented in Section 1.4.3

The 3D points can be filtered to remove obvious errors. Advanced filtering methods can be implemented, such as bilateral filtering [Digne and De Franchis, 2017], and filtering using color information or confidence measures from the disparity map (see Section 1.4.3) [Youssefi et al., 2024]. In the CARS pipeline, two different filtering steps are carried out: one for removing statistical outliers, and one for removing so-called "small-components".

Statistical outliers filtering

Statistical outliers are determined by considering the positions of their neighbors. For each point P, we compute the mean distance μ_P to its N neighbors. Then we can compute for each point P the mean distance μ and standard deviation σ of its N neighbors as:

$$\mu = \frac{1}{N} \sum_{i=1}^{N} \mu_i \quad (1.21)$$

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} (\mu_i - \mu)^2} \quad (1.22)$$

A point P is considered to be a statistical outlier if the difference between its mean distance μ_P and the mean distance μ of the whole neighboring is too large:

$$\mu_P > \mu + k\sigma \quad (1.23)$$

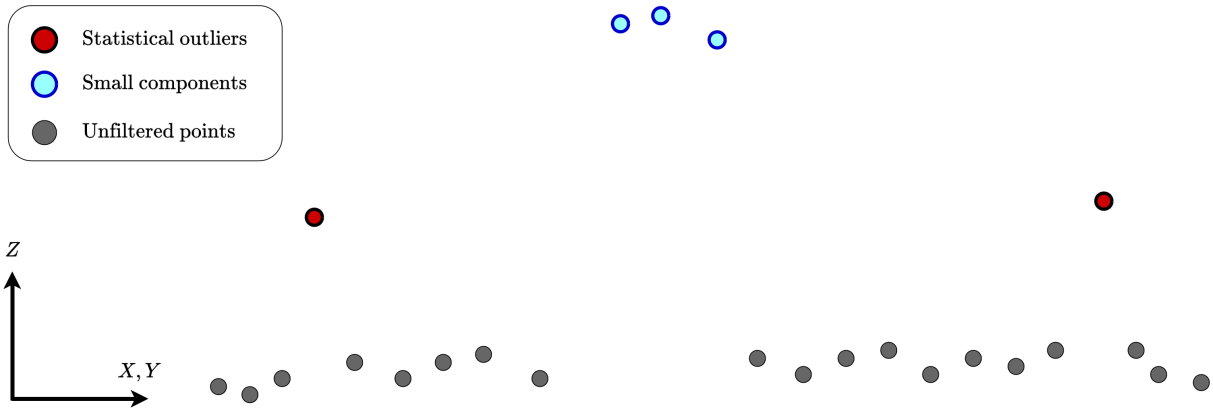


Figure 1.22: Example of statistical outliers and small components filtering of a 3D point cloud.

Where k is a constant, usually set to 5 by the user. We also consider $N = 50$.

Small-components filtering

The other filtering method, named small-components filtering, attempts to remove small isolated clusters of points. For each point, we count the number of neighbors N present within a distance D_{max}. If this number is inferior to a given threshold N_{min}, then we consider the point belongs to a small component and is removed. Formally, a point P is removed if:

$$\#\{ \text{ Points } Q \mid \sqrt{(P-Q)(P-Q)^T} \leqslant D_{max} \} \leqslant N_{min} \tag{1.24}$$

where # is the cardinal of a set. We usually set D_{max} to 3 m and N_{min} to 50.

Figure 1.22 illustrates the two filtering methods presented.

Filtering the point cloud results in a 3D product that can already be provided as such to users. However, point clouds, while containing more 3D information than DSMs, can be hard to manipulate in conjunction with other GIS data. Projecting the point cloud on a regular grid to produce a DSM is thus often preferred, which constitutes the final step of the stereo pipeline.

1.3.5 Rasterization

Rasterization consists in projecting the 3D points onto a regular grid over the (X, Y) plane to produce the final DSM. One of the challenges faced when projecting the point cloud is that of the low density of the point cloud relative to the DSM grid. Indeed, if the density of the point cloud is high enough, multiple 3D points can be projected to the same cell, which raises the question on how to merge their 3D information. On the other hand, if the density is not high enough, there may be some cells where no points are projected.

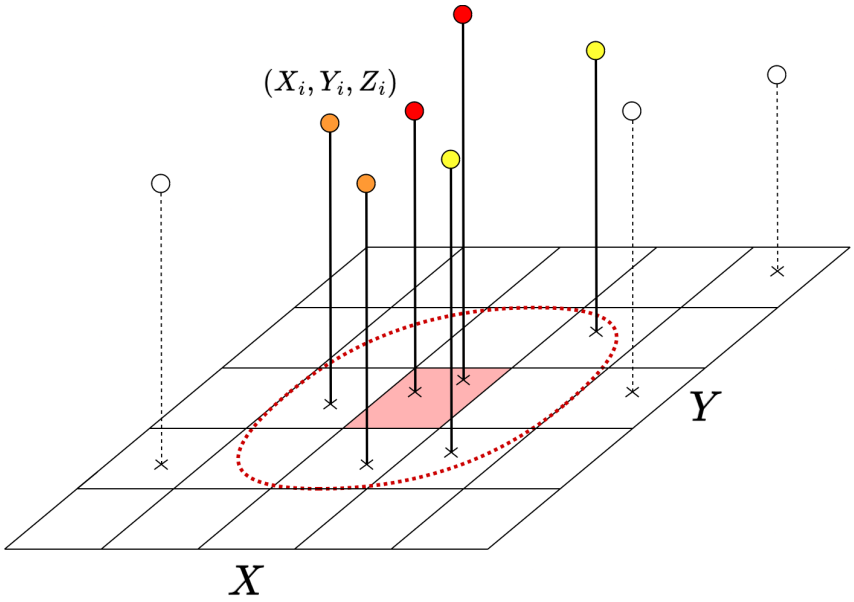


Figure 1.23: Rasterization of the point clouds on a regular grid. Red points are nearer to the center (X, Y) of the considered cell, while yellow points are further away.

The CARS stereo pipeline uses a Gaussian interpolation to fuse the information of point clouds. Given a cell with coordinates (X, Y) of the DSM, we consider every point $P_i = (X_i, Y_i, Z_i)$ in a given radius r of (X, Y), and note PC_{XY} the point cloud containing those points. The final value of the DSM is then computed as the following mean with Gaussian weights:

$$\mathrm{DSM}(X,Y) = \frac{\sum_{P_i \in PC_{XY}} Z_i \cdot e^{-\frac{(X_i-X)^2+(Y_i-Y)^2}{2\sigma^2}}}{\sum_{P_i \in PC_{XY}} e^{-\frac{(X_i-X)^2+(Y_i-Y)^2}{2\sigma^2}}} \tag{1.25}$$

with σ usually set to 0.3 m and the radius r being 3 m. Figure 1.23 illustrates the rasterization process.

Rasterizing with this method provides the advantage of smoothing the potential elevation variations still present in the 3D point cloud, while allocating more weight to points that are near the cell center. This method can also be found in other stereo pipelines [Shean et al., 2016], while other pipelines use different weighting methods, such as Inverse Distance Weightings (IDW) [Rupnik et al., 2017], producing similar results.

> **Remark:** If no points are projected in a given cell (or its direct neighbors), then the cell can be filled with *nodata* values. Different methods can be considered to fill holes, such as directly using the values of nearest valid neighboring cells, or interpolating their values. More advanced methods consists in simulating a cloth-like surface to fill the holes as in [Lallement et al., 2022].

As we have seen in the previous sections, a photogrammetry pipeline consists of multiple processing steps with intermediary products. For each step of the pipeline, different methods (*e.g.* matching, filtering *etc.*), parameterization and post-processes are available. This broad range of solutions allows adapting our processes to the type of images and terrain observed, but it sometimes makes it difficult to determine the configuration producing the best quality DSM, or to single out a general good-working configuration. The values of the different step parameters presented previously, and that we will be using throughout this thesis, result from testing and sensitivity analyses. In this thesis, we will therefore not discuss their value and instead focus on quantifying and propagating the uncertainty of the CARS stereo pipeline.

1.4 Uncertainty in Stereophotogrammetry

Producing high resolution DSMs is a complex task, where many uncertainties arise. Those uncertainties can be associated with input data (noise on images, sensor model location uncertainty) or be caused by the processing of those data (resampling errors, disparity computation, rasterization method).

1.4.1 Related Work

We will first look at attempts to characterize the uncertainty associated with DSMs. The quality of DSM can greatly vary depending on the quality/resolution of the data used, the accuracy of the geolocation, the performances of the processing algorithm *etc.*

For instance, in products like TanDEM-X, quality masks can be provided alongside the DEM, usually based on the sensor or method used for producing the DEM. TanDEM-X produces an elevation error map [Wessel, 2018], representing for each DEM pixel the standard deviation corresponding to the elevation error. The value is derived from the interferometric coherence, and is considered to be a stochastic error. It does not include any contributions of epistemic errors, such as erroneous orbital parameters. The altimetric and planimetric accuracy of IGN's LiDAR HD 3D points have been measured using ground control points (25 cm of planimetric relative accuracy, 5 cm of altimetric relative accuracy). Other intrinsic quality parameters of the IGN's LiDAR HD numerical models have not been measured [IGN, 2024]. Another IGN's DTM product, the RGE ALTI®, possesses a mask indicating the quality of the data provided. This quality mask is composed of the data source used (LiDAR, stereo photogrammetry, Radar) for which the mean squared altimetric error has been determined for each source [IGN, 2013]. In particular, the photogrammetry altimetric accuracy is estimated between 1.5 and 2.5 m according to [IGN, 1994].

Other works try to estimate the uncertainty associated with a DSM based not on the method or sensor used, but on the values of the DSM itself. [Mesa-Mingorance and Ariza-López, 2020] reviews those different methods for assessing the accuracy of DSMs. Previous work mostly produces confidence intervals associated with a DSM [Oksanen, 2006, Panagiotakis et al., 2018, Deschamps-Berger, 2021]. In [Wang et al., 2015], robust estimators are used to evaluate the mean, median, and variance of a DSM altitude distribution. In [Hugonnet et al., 2022], advanced statistical methods such as variograms are used to measure the spatial correlation of errors, depending on the values of the slope of the terrain. All those methods produce a confidence interval that globally describes the uncertainty associated with a given DSM (or a single interval per slope category in [Hugonnet et al., 2022]). They also estimate the uncertainty *a posteriori*, meaning that they do not consider the means by which the DSM was created. The uncertainty is estimated using a set of reference points, either extracted from a better resolution DSM, or from GPS ground control points. In [Wang et al., 2015], a confidence interval is computed based on the residuals between the DSM and a bilinear interpolation of samples from the same DSM, assuming that the quality of a DSM can be evaluated without using an external reference.

Photogrammetry pipelines presented in Section 1.3.1 also have different methods of estimating the uncertainty of the produced DSMs. For instance, the NASA's ASP pipeline can take as inputs camera planimetric standard deviation of position uncertainty (expressed in meters), and propagate it during the triangulation and rasterization steps. The documentation [NASA, 2020] details the method used for propagating the covariance matrix. It explicitly states that the propagated uncertainty does not represent the error between the predicted elevation and a hypothetical ground truth, even though they are expressed in the same unit of measure. It is rather the propagated covariance from the camera position projected in the horizontal and vertical directions, regardless of the matching errors or triangulation errors. We did not find clear documentation on how the S2P pipeline handles uncertainties, but the available code of the algorithm suggests that a confidence measure can be computed during the dense matching step of the algorithm (presented later in Section 1.4.3) and provided alongside the final DSM as a quality mask. The confidence measure used is a measure of the consensus between the different regularized directions during the SGM regularization. It is thus a dimensionless parameter, indicating if the predicted elevation is confident or not. IGN's MicMac software computes different errors such as the residual errors from Bundle Adjustment step [IGN, 2022], and provides a correlation map alongside the DSM. SETSM also computes matching scores [Noh and Howat, 2017], but it does not seem to be propagated until the final DSM. DLR's CATENA code is not available, and no documentation indicates the existence of a quality map. In any case, it seems as most pipelines compute a confidence measure (usually

during the dense matching step) and propagate it to create a quality mask alongside the final DSM. Different methods for producing such a quality mask have also been investigated in [de Joinville, 2001, Boudet, 2007], where different confidence scores from the dense-matching step are compared. This quality mask produced indicates the amount of confidence associated with each DSM cell value but does not reflect its potential altimetric error.

To the best of our knowledge, there is currently no method for producing elevation confidence intervals at a pixel level, by opposition to the global or almost global intervals in [Oksanen, 2006, Wang et al., 2015, Hugonnet et al., 2022]. Furthermore, the uncertainty arising from the different photogrammetry algorithms is not considered when producing those intervals. The main contribution of this thesis is the production of pixel-wise confidence intervals for DSMs produced by photogrammetry. Those intervals are first computed during the dense matching step, and then propagated in the rest of the pipeline, resulting in elevation confidence intervals. In the following section, we present different sources of uncertainty present in the CARS pipeline.

1.4.2 Uncertainty in the CARS pipeline

In this thesis, we focused on quantifying the uncertainty alongside the creation process of DSM. We thus differ in this regard from previous work, which produce a single confidence intervals from the final produced DSM. Multiple sources of uncertainty influence the production of a DSM, which we will now detail.

A first source of uncertainty comes from the input data, *i.e.* the radiometric uncertainty associated with stereo images and the geometric uncertainty associated with sensor models. The radiometric uncertainty is mainly due to noise from the sensor and from atmospheric effects appearing on the acquisitions. Those radiometric errors can be estimated and are relatively moderate [Jacobsen et al., 2014]. Regarding the uncertainty associated with the sensor models, satellite movements not captured by models can have a big impact on the final images, especially for push-broom sensors. For instance, vibration of the satellite can lead to biases on the geolocation of the different rows of the image. Those biases will themselves be propagated to the final DSM, leading to errors of a few meters [Loghin et al., 2019]. Additionally, RPC models attempt to represent real lines of sight with polynomial coefficient, which is not exact and possess its own accuracy. The method used for coefficient calibration and the frequency of calibration also bring their share of uncertainty.

Another source of uncertainty arises from the different processing steps of the stereo pipeline. First, different resamplings occur in order to convert stereo images from sensor geometry to epipolar geometry one. Those resamplings are based on SIFT matches,

presented in Section 1.3.2. SIFT matches are used to refine the epipolar grid depend on the performance of the SIFT algorithm used to obtain them. In homogeneous and texture-less area, *e.g.* glaciers, false matches can arise. This leads to wrong epipolar lines, which can prevent accurate results in the dense matching step and result in a faulty 3D reconstruction. We assume in this thesis that the refinement of epipolar grids using SIFT points, which minimizes the mean error, provides accurate epipolar resampling grids with an epipolar error of around a tenth of a pixel [Franchis et al., 2014]. However, the resampling in epipolar geometry introduce errors if the input and target resolutions do not respect Shannon criteria [Delon and Rougé, 2007]. In the case of Pléiades images, the acquisition resolution is usually high enough to ensure correct resampling [Jacobsen et al., 2014].

The resampled images in epipolar geometry are then used as inputs for the dense matching step. Dense stereo matching is a complex task, for which many algorithms exist, each potentially presenting different performances. When compared to epipolar errors, that typically are less than a pixel [Franchis et al., 2014], dense matching errors can potentially reach higher magnitudes, depending on the size of the considered disparity range (sometimes reaching hundreds of pixels). Considering those orders of magnitude, estimating and quantifying the uncertainty of the stereo matching process is therefore crucial to control the uncertainty on the output DSM. Some sources of those errors are now presented. Because the correlator usually compares windows (in our case 5×5 windows using the CENSUS cost function or 11×11 windows using MC-CNN) and not single pixels, this can create an adherence effect near building borders, as in Figure 1.15. Note that in [Okutomi and Kanade, 1994], authors have been trying to adapt the window shape to reduce the uncertainty of stereo matching, but this method requires to iteratively compute costs on different windows, which can become quite expensive. Using a window-based correlator with SGM regularization, if correctly parameterized, usually presents good performance in areas without elevation discontinuities. This can become a problem in urban areas, where the presence of high buildings represents an additional challenge for the correlator. As SGM regularization also penalizes disparity changes, tops of buildings tend to be extended beyond their true footprint. Other processes, such as filtering or sub-pixel refinement, improve the quality of the disparity map but require special care in handling their induced uncertainty. Section 1.4.3 delves deeper into details regarding the different methods that have been developed to quantify the uncertainty of dense stereo matching.

Once the disparity has been estimated, 3D points can be triangulated by intersecting RPC lines from each matched pixel. However, we saw previously that there is no guarantee that the 3D lines do intersect. If they indeed do not intersect, the 3D point is defined as the point minimizing its squared distances to both lines of sight. An alternative is to

modify the geolocation line from the secondary image so that it intersects the line from the reference image. In both cases, the localization of the 3D point is not exact. This uncertainty stems from the fact that RPC have their own limited resolution, and that stereo images (resampled or not) do not necessarily point to the exact same location in the object space.

The final part of the stereo pipeline is to rasterize the point cloud onto a regular grid, thus yielding the DSM. When characterizing the uncertainty in the final result, we must first agree on what the DSM is assumed to represent. It is common to consider that each pixel's value should represent the average elevation over the cell. However, providing the maximum or minimal elevation might be more adapted to some scenarios: for instance if the DSM is used to prepare very low altitude flights (for drones *etc.*), the maximum elevation is more relevant as one would want to avoid any foliage or power line. Those elements could disappear in the final DSM if it represented the mean or median elevation. In this thesis, we will consider that the DSM represents the (weighted) average elevation. Depending on the resolution of input images and the desired output resolution of the DSM, the density of points per DSM cell will vary. In our applications, the input and output resolutions are identical, as we typically desire to produce a DSM at 50 cm resolution from 50 cm panchromatic images. This means that, in average, there is one 3D point per DSM cell. For occluded regions, or when we discarded stereo matches that seemed incorrect, there might be no point directly in the output cell. In this case, the value of the DSM cell will be determined entirely by the values of points in neighboring cells, even if they are far away and have small averaging weights. Interpreting the final DSM as the average elevation on each cell is thus debatable, as the average is computed on a limited number of points, and sometimes not even belonging to the considered cell. Note that if there are no points around in a given radius, the cell will be left empty.

Different sources of uncertainty occurring throughout the stereo pipeline were presented in the previous paragraphs. Characterizing, modeling and propagating all of those uncertainties could not be considered in the span of this thesis. We thus focus mostly on the uncertainty arising from the dense stereo matching step, as it is the source of the biggest errors in the pipeline. Chapter 4 investigates how uncertainty from the input epipolar images can be propagated in the stereo matching step, and Chapter 5 attempts to model the processing uncertainty of the stereo algorithm itself. We also propagate this uncertainty all the way to the output DSM and show that it can correctly estimate the errors made during the DSM production.

1.4.3 Uncertainty Quantification in Dense Stereo Matching

As stereo matching is a popular problem in computer vision, many methods for quantifying its uncertainty have been proposed in the literature. Without being exhaustive, this section presents a quick overview of the main approaches as well as the solutions currently implemented in the dense stereo correlator Pandora used in the CARS pipeline.

The way uncertainty is quantified in stereo matching is mainly done by producing *confidence maps*, *i.e.* a mapping for each pixel (row, col) to a real confidence value, usually between 0 and 1. By convention, a value of 0 means that we are not confident in the disparity value associated with (row, col). Conversely, a confidence value of 1 indicates that we are very confidence in the predicted disparity.

There are multiple sources of information that can be used to compute a confidence measure. Left and right input images and the predicted disparity map being those available to every method, but cost-based approaches can also make use of the cost volume, as it contains a quantity of useful information. Most confidence measures are handcrafted using those different information sources. For reviews on those methods, we refer to [Egnal et al., 2004, Hu and Mordohai, 2012, Poggi et al., 2017]. With the rising use of deep-learning in stereo, many networks have also been developed to estimate the uncertainty. A review of methods using regression forests can be found in [Min-Gyu Park and Yoon, 2015], and a more general review, including the use of 2D and 3D CNNs on the cost volume, can be found in [Poggi et al., 2021].

> **Remark:** Quantifying the uncertainty in stereo matching is a popular field of research. Recently, people have even been trying to evaluate the uncertainty of the confidence estimation itself, called *meta-confidence* [Kim et al., 2022].

In Example 2, we present some examples of confidence measures that use different sources of information.

Example 2: Regarding confidence measure based on the disparity map, we already presented a type of binary confidence measure in Section 1.3.3 with the cross-checking test from Equation (1.14). Other methods compute for instance the local variance of the disparity map, where a low variance suggests confident regions.

For methods using the cost volume, a simple method for measuring the cost would be using the value of the matching score measure (MSM, [Egnal et al., 2004]) for a

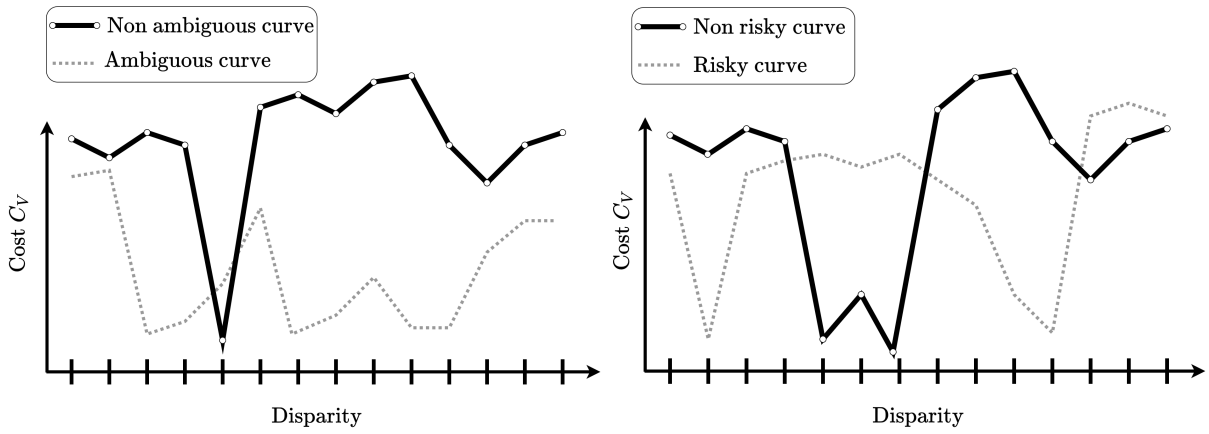


Figure 1.24: Cost curves with different levels of ambiguity (a) and risk (b)

given disparity at coordinates $(row,\ col)$:

$$MSM(row,\ col) = -C_V(row, col, \mathcal{D}(row,\ col)) \tag{1.26}$$

This measure can be normalized between 0 and 1 using the global minimum and maximum of the cost volume. The idea behind this measure is the following: a high matching cost for a selected disparity indicates that the two matched pixels are not that similar, and thus the match is not confident. Other measures using the matching cost compare the value of the first and second minimum of a cost curve, or measure the curvature of the cost curves. More advances measures make use of 3D CNNs on the entirety of the cost volume to learn an efficient confidence measure [Mehltretter and Heipke, 2019].

Measures based on the input images usually measure the gradient [Haeusler et al., 2013] or variance [Park and Yoon, 2019] of input images. High gradient or high variances indicate highly-textured regions, which are often easier to match. The confidence is therefore higher for those pixels.

Deep-learning approaches can combine multiple sources of information (input images, cost volume, disparity map) to learn a confidence measure [Tosi et al., 2018, Kim et al., 2020].

In this thesis, we will also consider another confidence measure that can already be computed with the Pandora correlator. This measure will help us to quantify the uncertainty in the photogrammetry pipeline. This measure is referred to as *confidence from ambiguity* [Sarrazin et al., 2021]. This method is based on the cost volume and quantifies how easy or hard it is to single out the correct disparity in a cost curve. Figure 1.24(a) presents two cost curves, one which is not ambiguous as there is a single well-defined

minimum, and an ambiguous cost curve which presents multiple values that are close to the minimum. To compute it, we first start by constructing what is called an *ambiguity curve*. We do that by counting how many disparities have a cost close within a threshold η to the minimal cost, for increasing values of η. Formally, we need to define for a pixel $(row,\ col)$ the set of all disparities whose cost is within a range η of the minimal cost, and then define the ambiguity curve as the cardinal of this set:

$$\mathcal{D}_\eta = \{d \mid C_V(row,\ col,\ d) \leqslant \min_\delta C_V(row,\ col,\ \delta) + \eta\} \tag{1.27}$$

$$amb(row,\ col,\ \eta) = \#\mathcal{D}_\eta \tag{1.28}$$

where $\#$ is the cardinal of a set. Evaluating amb for different η gives the ambiguity curve. Figure 1.25(a) presents a cost curve with different values of η. Figure 1.25(b) presents the resulting ambiguity curve. On those two figures we can see for instance that for η_3 there are 6 disparities whose cost is lower than $\min_\delta C_V(row,\ col,\ \delta) + \eta_3$. For non-ambiguous cost curves, amb will increase only for high values of η. On the contrary, for ambiguous curves, amb will be high for small values of η. To obtain a scalar value from amb, we compute the area under its curve, normalized by the range of η:

$$\mathrm{AUC}_{amb}(row,\ col) = \frac{1}{\max \eta - \min \eta} \int_\eta amb(row,\ col,\ \eta) d\eta \tag{1.29}$$

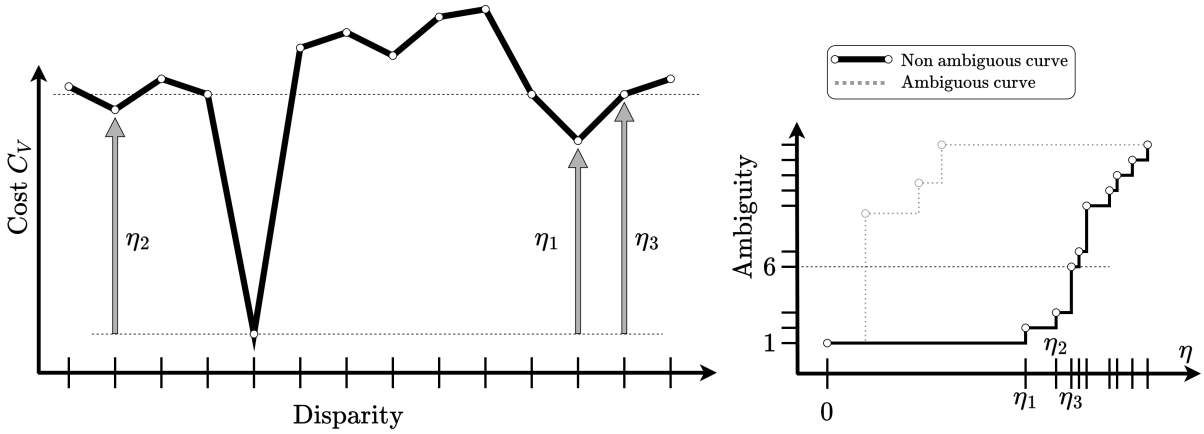
It results on low values for confident (non-ambiguous) cost curves, and high values for less confident (ambiguous) curves. Because confidence measures actually present high values for confident pixels, AUC_{amb} is normalized and inverted in order to obtain the confidence from ambiguity c_{amb}:

$$c_{amb}(row,\ col) = \frac{\max \mathrm{AUC}_{amb} - \mathrm{AUC}_{amb}(row,\ col)}{\max \mathrm{AUC}_{amb} - \min \mathrm{AUC}_{amb}} \tag{1.30}$$

This way, values of c_{amb} near 0 indicate that we are not confident in the predicted disparity. Reversely, values of c_{amb} near 1 indicate that we are confident in the predicted disparity.

> **Remark:** A similar notion of ambiguity can be traced back in [de Joinville, 2001], where the difference in disparity between the first and second minima is measured. In the following, we will not consider this notion when referring to the ambiguity, but rather the one introduced in equation (1.30).

The confidence from ambiguity, like other confidence measures, is designed to indicate potentially wrong matches. It however does not measure the extent of the potential error. After all, it is possible that we predict a wrong disparity, while the correct disparity is right next to our prediction. The impact of this error on the final result will be smaller



(a) Cost curve with different values of η. Horizontal dotted lines indicates the range of costs between $\min_\delta C_V(row,\ col,\ \delta)$ and $\min_\delta C_V(row,\ col,\ \delta) + \eta_3$

(b) Associated ambiguity curve in full line. Ambiguous curve in gray dotted line

Figure 1.25: Illustration of the computation of the ambiguity curve.

than if the true disparity is at the other side of the disparity range. There is a distinction to be made between the value of a confidence measures and the magnitude of a potential error, even if they can be correlated in some cases.

In Chapters 5 and 6, we introduce a method for computing confidence intervals, which aims to serve as a complement to confidence measures as it estimates the magnitude of the potential error. This method is motivated by discussions with different users and experts in 3D modeling from the AI4GEO consortium (`https://www.ai4geo.eu/`). Indeed, a common point emerging from those discussions is the desire to produce confidence intervals alongside photogrammetry DSMs. This also converges with the quality map requirement of the CO3D mission. We therefore develop our method with the objective of answering to this user need.

Conclusion

This chapter presented the principles of stereophotogrammetry using satellite imagery. We presented different satellites relevant in the context of this thesis (*i.e.* the CO3D and Pléiades constellations) focusing on their sensors and their geometry. We also detailed the global architecture of a stereo pipeline, and the specificities of the pipeline CARS that we will be considering in this thesis. We finally presented the different sources of uncertainty associated with this pipeline, and some measures that can be used to estimate the uncertainty in the dense matching step of the pipeline. Chapters 2 and 3 will present relevant uncertainty models and concepts for this thesis, while Chapters 4 to 6 will present a method for estimating the uncertainty at different steps of the photogrammetry pipeline.

Chapter 2

Mathematical Representations of Uncertainty

2.1 Introduction

This chapter takes some distance from the stereo vision problem presented previously, and instead describes more formally different representations of uncertainty and the tools that can be used to manipulate them. In this thesis, we consider classical probabilities (Section 2.3.1), possibility distributions (Section 2.3.4) and p-boxes (Section 2.3.5) to model uncertainty. We also consider the case where we have uncertainty on multiple variables simultaneously. In this setting, the dependency between the different sources must be taken into account. We thus present a dependency model called copulas in Section 2.4, and some of its properties. The thorough presentation of those concepts will then be put into relations in Chapter 3.

2.2 Notations

We introduce here a few notations that will be used in this chapter.

- *s.t.* means "such that"
- We will often present theorems or results with n different variables, spaces, or Cartesian products of n elements. Notations therefore quickly become quite heavy. For that reason, we often use the notation "..." to imply that we are enumerating all variables. For instance, if we apply a function f to four variables x_1, x_2, x_3, x_4, we will write $f(x_1, \ldots, x_4)$.
- When working with n variables, the index i will usually be used to refer to the i-th

variable (or one of its attribute), and will be appended as a subscript when possible, otherwise as a superscript. For instance, if $x \in \mathbb{R}^n$, then x_i will refer to the i-th component of x. If $m_\times \in \mathbb{R}^n$, then $m_\times^i$ will refer to i-th component of $m_\times$.

- $[\![\cdot, \cdot]\!]$ refers to an interval of integers. For instance $[\![1,\ 4]\!] = \{1,\ 2,\ 3,\ 4\}$.
- The power set of a set $\mathcal{X}$ is noted $2^{\mathcal{X}}$. It corresponds to the set of all sets included in $\mathcal{X}$. In the discrete case, if the cardinal of $\mathcal{X}$ is n, then the cardinal of the power set is 2^n, thus the notation.

2.3 Different Models to Represent Uncertainty

Assessing the reliability of an engineering system requires quantifying the uncertainty of the input parameters or of the system itself, using models of uncertainty. Modeling the uncertainty can be done in various ways, depending on the type of uncertainty considered and the available measures or *a priori* regarding the uncertain sources. Most common models are probability distributions, which have been studied extensively. When using those models, we know —or assume— that the information we try to estimate is of stochastic (or random) nature, and that we are able to precisely describe its structure using a probability distribution. However, there are many cases where such assumptions cannot be made: for instance when data is insufficient to determine the correct probability distribution, or when the uncertainty is not random, but epistemic. In those cases, we can use other models, such as:

- fuzzy sets [Zadeh, 1999] when trying to estimate the degree of truth of a statement such as "This person is tall"
- intervals [Jaulin et al., 2001], where no preferences are given inside a specific range of possible values
- Imprecise Probabilities (IP) which tries to extend the concept of probabilities in order to model epistemic uncertainty [Augustin et al., 2014]

This list is not exhaustive. Additionally, those different models can sometimes be equivalent. Choosing to use one or another depends on the nature of the problem and of the available data. In this chapter, we will mainly consider probability distributions (Section 2.3.1) and IP (Section 2.3.2). Two specific cases of IP will be detailed, namely possibility distributions in Section 2.3.4 and p-boxes in Section 2.3.5.

> **Remark:** Distinguishing between stochastic and epistemic uncertainty is a model accepted by many to distinguish between different situations of uncertainty. One

> could argue that stochastic uncertainty is, to some extent, equivalent to epistemic uncertainty. Indeed, if we had enough knowledge on initial conditions of a die throw for instance (force and torque applied on the dice, its exact shape and mass distribution *etc.*), as well as the exact physics model, one could predict with certainty on which side it would land. There would therefore be no aleatoric process at stake here. The question whether or not we should make the distinction between aleatoric and epistemic uncertainty is of interest regarding theoretical aspects of the nature of uncertainty. For real-life applications, however, differentiating between the two seems reasonable, as we cannot know every parameter and the exact model involved for every quantity of interest.

At the end of the day, choosing one model over another is not always straightforward. It requires being aware of the type of uncertainty faced, of the strengths and limitations of each model, of the tools available to manipulate the models *etc.* When it comes to less common models cited above, it fundamentally requires being aware of the existence of such models, which is not always the case for non-specialists. During this thesis, we tried to promote less common models, especially possibility distributions. We did so by presenting real life cases where they could be used while improving uncertainty modeling in the field of stereophotogrammetry (see Chapters 4 and 5).

2.3.1 Probabilities

Probability measures are a classical framework to represent uncertainty. There are multiple ways of interpreting them, mainly with a *frequentist* approach, or a *Bayesian* approach. From a frequentist point of view, probabilities are well-fitted to represent stochastic uncertainty, *i.e.* uncertainty regarding events that can get a different result each time we run an experiment or acquire a measure (typically, noise on a sensor). From a *Bayesian* point of view, probabilities represent a state of knowledge or degree of belief, and can be updated with additional information [Williamson, 2009]. This leads to the notion of prior and posterior probability that will not be considered in this thesis.

We remind here basic definitions regarding probability distributions.

Definition 1: Probability Space

We call a probability space $(\mathcal{X},\ \mathcal{A},\ P)$ a tuple where:

- $\mathcal{X}$ is the set of possible outcomes (for instance, head or tails for a toss coin)
- $\mathcal{A}$ is the set of all subsets of $\mathcal{X}$ for which a probability can be measured (for instance $\{\emptyset, \{\text{heads}\}, \{\text{tails}\}, \{\text{heads, tails}\}\}$)

- P is the probability measure assigning a probability to each of the sets of $\mathcal{A}$. For instance, for a fair coin, $P(\emptyset) = 0$, $P(\{\text{heads}\}) = P(\{\text{tails}\}) = 0.5$ and $P(\mathcal{X}) = 1$.

Note that $\mathcal{A}$ must be a σ-algebra, meaning that it is closed under complement, countable unions and countable intersections. P is a probability measure if it verifies all the Kolmogorov axioms:

- $\forall A \in \mathcal{A},\ P(A) \in [0, 1]$
- $P(\mathcal{X}) = 1$
- for any countable disjoint family of sets $A_i \in \mathcal{A}$, $P(\cup_i A_i) = \sum_i\ P(A_i)$

Definition 2: Random Variable

A random variable X is a measurable function from $\mathcal{X}$ to a measurable space (which is $\mathbb{R}$ or a subset of $\mathbb{R}$). We can then measure the probability that X takes its values in a measurable set $E \subseteq \mathbb{R}$:

$$P(E) = P(\{x \in \mathcal{X} \mid X(x) \in E\})$$

Using a random variable allows considering the probability measure on different spaces. We then call P the probability distribution of the considered random variable X.

Other useful concepts regarding probability distributions are Probability Density Function (PDF) and Cumulative Distribution Function (CDF):

Definition 3: Cumulative Distribution Function

A Cumulative Distribution Function (CDF) F_X of a random variable X with real values is the probability that X will be less or equal to a number x. Formally, we define $F_X : \mathcal{X} \to [0, 1]$:

$$\forall x \in \mathcal{X},\ F_X(x) = P(X \leqslant x)$$

Definition 4: Density Function

A random variable X is said to possess a Probability Density Function (PDF) if there exists a positive integrable function f over $\mathbb{R}$ *s.t.* $\forall (x_1, x_2) \in \mathbb{R}^2$:

$$P(x_1 \leqslant X \leqslant x_2) = \int_{x_1}^{x_2} f(x)dx$$

In the continuous case, the PDF f is also the derivative of the CDF of X. In the discrete case, the PDF is also called *probability mass function*, and is defined as:

$$f(x_1) = P(X = x_1)$$

In the rest of the chapter, we consider random variables X on discrete spaces $\mathcal{X}$. If $\mathcal{X} = \{x_1, \ \ldots, \ x_n\}$, we call $\{x_i\}$ an atom of $\mathcal{X}$, and the probability distribution P of X on $\mathcal{X}$ is completely determined by its PDF, which is the value of P on atoms. For simplicity of notation, we will not always use braces around atoms when computing their probability. So we will sometimes write $P(x_1)$ instead of $P(X = \{x_1\})$.

As stated previously, probability measures are fitted to represent stochastic uncertainty. The following example illustrates why probability measures are not adapted to represent epistemic uncertainty:

Example 3: Let consider a card facing down, with a number written on its hidden side. The person who wrote the number tells you that they chose to write either 1, 2 or 3 on it. We should note that because they chose to write a number, the uncertainty on its value is not random. They then ask you to evaluate your chances of guessing the correct number and its parity, *i.e.* if it is odd or even. We first consider the random variable X taking values in $\{1, \ 2, \ 3\}$. Because you have no further information and thus no preferences on the values of X, a common (yet arguably inadequate) decision is to associate the uniform distribution P to X:

$$P(X = 1) = P(X = 2) = P(X = 3) = \frac{1}{3}$$

Concerning the parity of the number, we may consider the random variable Y defined such that $Y = 0$ if "$X = 1$ or $X = 3$" and $Y = 1$ if "$X = 2$". Because we have no information on the number written on the card, we also do not have any preferences on the values of Y. Following the same reasoning as before, one might be tempted to associate a uniform distribution to it. However, deducing the PDF of Y from that of

X yields:

$$P(Y=0) = P(X=1) + P(X=3) = \frac{2}{3}, \qquad P(Y=1) = P(X=2) = \frac{1}{3}$$

which is clearly not the PDF of a uniform distribution. By assuming we have no preferences on the values of a variable, we actually deduced preferences on the values of another variable.

This example shows that one should use a uniform distribution only when they are certain that all values are equiprobable. This is due to the fact that uniform distributions are well suited for representing statements like "all values have the same likelihood" but not for statements like "I have no information over the values, and thus no preference". Indeed, a probability distribution actually contains a lot of information about a (random) variable, which is not suited to represent epistemic uncertainty.

> **Remark:** From a Bayesian point of view, a probability can also represent a degree of belief, allowing them to represent epistemic uncertainty and not only stochastic uncertainty, in theory. However, it does not solve the expressiveness problem raised by this example. Bayesians are still reasoning with probabilities, which make no difference between "I have no preference between these events" and "these events are equiprobable". Even though they can update their prior with additional information, this would not fix the problem presented here. In the absence of additional information, basing a decision on the prior would lead to debatable conclusions.

2.3.2 Imprecise Probabilities

As highlighted in Example 3, uncertainty cannot always be correctly modeled by probabilities, especially in a context where data is sparse. To overcome this problem, a generalization of probabilities has been introduced, called Imprecise Probabilities (IP), which provides a general framework for working with both aleatoric and epistemic uncertainty. It uses the concept of lower and upper probabilities, which are quite generic and flexible, and can be derived in more specific models. Here is a brief scope of the relevant tools it encompasses: the special case of *belief functions*, themselves containing specific sub-categories such as possibility distributions (Section 2.3.4) and probability boxes (Section 2.3.5). It also contains probabilities presented in Section 2.3.1, which we will call *precise* probabilities in contrast to *imprecise* probabilities. Figure 2.1 is a non-exhaustive overview of relationships and specificities of each imprecise model.

Remark: On a more generic level, IP can be described by sets of acceptable gambles and by lower and upper expectations [Walley, 1991, Augustin et al., 2014]. Although very interesting, we did not use them in our applications and thus do not consider them in this thesis.

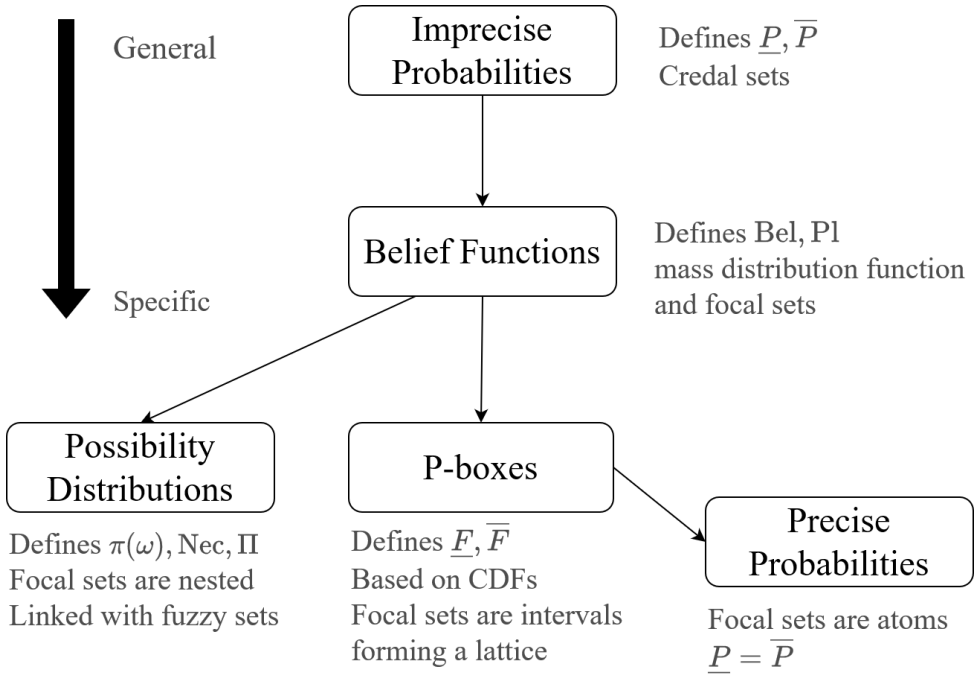


Figure 2.1: Diagram representing the relationship between different IP models presented throughout Section 2.3.

As stated previously, a core concept of IP is the concept of lower and upper probabilities. Similarly to precise probabilities, a lower probability $\underline{P}$ and an upper probability $\overline{P}$ are mappings from a σ-algebra $\mathcal{A}$ to $[0, 1]$. However, while a probability P gives a single measure of uncertainty for every event, lower and upper probabilities provide two bounds for every event, allowing them to express more complex uncertainty structures.

Remark: Formally, a lower probability $\underline{P}$ needs to be *super-additive*, *i.e.* to satisfy:

$$\forall A, B \in \mathcal{A}, \text{ if } A \cap B \neq \emptyset, \ \underline{P}(A \cup B) \geqslant \underline{P}(A) + \underline{P}(B) \tag{2.1}$$

Conversely, an upper probability is sub-additive, meaning that it verifies the same property as Equation (2.1) but with the inequality reversed.

Those properties are less constraining than their equivalent for precise probabilities in Definition 1, so lower and upper probabilities are generally *not* precise probabilities. The only case where they are precise probabilities is when $\underline{P} = \overline{P}$, because $\underline{P}$ is then additive as it is both super and sub-additive. In this case, IP are

actually a single precise probability. This illustrates the fact that precise probabilities are special cases of IP.

Precise probabilities define a measure of uncertainty towards a random variable X taking numerical values. IP however, can model the uncertainty towards a random variable taking set values instead of numerical one. We then say that X is a *random set* instead of a random variable. That being said, IP can also represent the uncertainty of random variables for which we assume a precise probability exists, but we are not able to determine it precisely. Indeed, lower and upper probabilities form the bounds of a family of precise probabilities called *credal set*.

Definition 5: Credal set

Given a lower probability $\underline{P}$ and an upper probability $\overline{P}$, a credal set $\mathfrak{M}$ is the set of all probabilities P that are greater that $\underline{P}$ and lower that $\overline{P}$:

$$\mathfrak{M}(\underline{P}, \overline{P}) = \{P \mid \forall A \in \mathcal{A},\ \underline{P}(A) \leqslant P(A) \leqslant \overline{P}(A)\} \tag{2.2}$$

We refer to $\mathfrak{M}(\underline{P}, \overline{P})$ as $\mathfrak{M}$ when no confusion is possible. Credal sets allow considering multiple probabilities at once, which improves on the limited expressiveness of a single probability measure. The gap between the two bounds of a credal set reflects how imprecise is the model, in terms of epistemic uncertainty.

Conversely, we can define lower and upper bounds from a set of probabilities $\mathfrak{M}$ as:

$$\forall A \in \mathcal{A},\ \underline{P}(A) = \inf_{P \in \mathfrak{M}} P(A)$$
$$\forall A \in \mathcal{A},\ \overline{P}(A) = \sup_{P \in \mathfrak{M}} P(A)$$

Note that $\mathfrak{M}$, contrary to $\mathfrak{M}(\underline{P}, \overline{P})$, is not necessarily defined by $\underline{P}, \overline{P}$. Therefore, even though $\mathfrak{M}$ and $\mathfrak{M}(\underline{P}, \overline{P})$ have the same bounds on events, they are not necessarily equal and only $\mathfrak{M} \subseteq \mathfrak{M}(\underline{P}, \overline{P})$ holds.

Although it is not required, we usually assume that credal sets verify additional properties expressed as follows:

Definition 6: Coherence and Avoiding sure loss

A credal set $\mathfrak{M}$ is said to avoid sure loss if it contains at least one probability measure,

i.e. if

$$\mathfrak{M} \neq \emptyset \tag{2.3}$$

In this thesis, we consider that a credal set $\mathfrak{M}$ induces *coherent* lower and upper probabilities if its lower and upper bounds on all events are attained by a probability measure, *i.e.* if:

$$\forall A \in \mathcal{A}, \exists P, P' \in \mathfrak{M} \ s.t. \ \underline{P}(A) = P(A) \text{ and } \overline{P}(A) = P'(A) \tag{2.4}$$

The bounds $\underline{P}, \overline{P}$ of a coherent credal set verify the following property:

$$\forall A \in \mathcal{A}, \ \underline{P}(A) = 1 - \overline{P}(A^c) \tag{2.5}$$

which is a generalization of the classical property for computing the complement of an event with precise probabilities. This allows us to only specify the lower bound $\underline{P}$ of a credal set to completely describe it, as its upper bound is determined by $\underline{P}$ through complementation. Defining a credal set requires specifying much more constraints on the probability space than in the case of probability distributions. Indeed, a lower probability must be defined on every possible event, while a precise probability can be defined by its PDF on possible values only (instead of possible events). In the case of a discrete space with n elements, a probability is completely determined by its values on the n atoms, while a lower bound is completely determined by its values on the 2^n considered events.

Remark: Sampling from a credal set is not straightforward. Multiple methods exist, the most intuitive consisting in sampling distributions from the credal set extreme points.

Remark: With the way we defined a credal set $\mathfrak{M}$, it is closed and convex. This is a common way of constructing credal sets, but is not the only way. We could for instance impose that all probabilities in $\mathfrak{M}$ belong to a family of Gaussian probabilities, which would prevent $\mathfrak{M}$ from being convex.

When constructing credal sets, it is common to possess a non-convex set of probabilities S. In that case, we can define the convex hull of a set of probabilities S:

Definition 7: Convex Hull

The convex hull (CH) of a set of probabilities S is the smallest (convex) credal set containing S.

Remark: The convex hull is computationally heavy to evaluate. Determining the infimum and supremum of a set of probability is not trivial, as it might be required to iterate through every element of S. It is however very useful, especially in the case where we do not know the set bounds on every event. In that case, computing the convex hull also allows us to determine the bounds on those events.

Example 4: Let us consider the scenario presented in Example 3 and model the uncertainty with a credal set instead of a single probability distribution. Because we have no information on the value written on the card except that it is in $\{1, 2, 3\}$, we characterize the uncertainty by lower and upper probability $\underline{P}$, $\overline{P}$ defined as follows:

$$\underline{P}(X = 1) = \underline{P}(X = 2) = \underline{P}(X = 3) = 0$$
$$\overline{P}(X = 1) = \overline{P}(X = 2) = \overline{P}(X = 3) = 1$$
$$\underline{P}(X \in \{1, 2, 3\}) = \overline{P}(X \in \{1, 2, 3\}) = 1$$

We can use Equation (2.5) for computing the bounds of remaining events.

Here, the credal set is the largest credal set possible, as its bounds are always 0 and 1 for events that are not $\emptyset$ or $\{1, 2, 3\}$. We say that it is the vacuous credal set, as it does not encode any information. It however solves the problem of a contradicting probability when evaluating the value of the card and its parity presented in Example 3

2.3.3 Belief Functions

A special case of IP are belief functions, which we will detail in this section. First, we will introduce a key concept that goes along belief functions: mass distribution functions. We will then derive belief functions from it.

Definition 8: Mass distribution function

Let $\mathcal{X}$ be a set of possible outcomes, and $2^{\mathcal{X}}$ its power set. A mass distribution function (or basic probability assignment [Shafer, 1976]) is a function $m : 2^{\mathcal{X}} \to [0, 1]$

s.t.:

$$m(\emptyset) = 0 \tag{2.6}$$

$$\sum_{a \subseteq \mathcal{X}} m(a) = 1 \tag{2.7}$$

There are multiple ways of interpreting $m(a)$, presented as follows:

- If we consider a random set X, then $m(a)$ encodes the probability mass that X takes a as its set value.
- If X is a random variable, then $m(a)$ encodes the available evidence that the numerical value of X is *exactly* in a, without any preferences for the values within a. This means there could also be other evidence that X is in $a' \subset a$, encoded with $m(a')$, and which could be either more or less than $m(a)$ depending on the amount of evidence available. Example 5 illustrates this with a toy scenario.
- $m(a)$ can also be linked to precise probability masses defined in Definition 4. If we assume there exists an unknown underlying probability measure P for X, $m(a)$ measures the probability mass that is assigned to a, but that can move freely to every point of a without any preference. In other words, with more information, we could distribute $m(a)$ to every element of a, and doing this for all a would lead to a precise probability mass distribution.

Remark: Equation (2.6) translates the fact that there is no evidence that the uncertain variable belongs to the empty set, *i.e.* that it is not defined. Relaxing this constraint allows accepting a certain amount of contradiction in our model. Equation (2.7) is a convention, which states that the total amount of evidence equals 1. It is similar to probabilities, which cannot be more than 1.

Definition 9: Focal set

Let $\mathcal{X}$ be a set of possible outcomes, $2^{\mathcal{X}}$ its power set, and $m : 2^{\mathcal{X}} \to [0, 1]$ a mass distribution function. A set $a \subseteq \mathcal{X}$ is called a focal set of m if and only if:

$$m(a) > 0 \tag{2.8}$$

Focal sets thus represent sets of the set of possible outcomes for which we have evidence. The set of all focal sets is sometimes referred to as the *core* of m.

Definition 10: Belief function, Plausibility function

Let $\mathcal{X}$ be the set of possible outcomes, $2^{\mathcal{X}}$ its power set, and $m : 2^{\mathcal{X}} \to [0, 1]$ a mass distribution function.

We define the belief function associated with m as the function $\text{Bel} : 2^{\mathcal{X}} \to [0, 1]$ who associates to all events A in $2^{\mathcal{X}}$:

$$\text{Bel}(A) = \sum_{a \subseteq A} m(a)$$

We define the plausibility function associated with m as the function $\text{Pl} : 2^{\mathcal{X}} \to [0, 1]$ who associates to all events A in $2^{\mathcal{X}}$:

$$\text{Pl}(A) = \sum_{\substack{a \\ a \cap A \neq \emptyset}} m(a)$$

Bel and Pl are special cases of lower and upper probabilities, and thus induce a credal set $\mathfrak{M}(\text{Bel}, \text{Pl})$ defined as:

$$\mathfrak{M}(\text{Bel}, \text{Pl}) = \{P \mid \forall A \subseteq \mathcal{X}, \text{Bel}(A) \leq P(A) \leq \text{Pl}(A)\}$$

We can interpret $\text{Bel}(A)$ as the amount of evidence that fully support A, and $\text{Pl}(A)$ as the amount of evidence that is consistent (or does not contradict) with A.

Bel and Pl are special cases of lower and upper probabilities, and verify (2.5) as for all events A it holds:

$$\begin{aligned}
\text{Bel}(A) &= \sum_{a \subseteq A} m(a) \\
&= \sum_{a \in \mathcal{X}} m(a) - \sum_{a \not\subseteq A} m(a) \\
&= 1 - \sum_{\substack{a \\ a \cap A^c \neq \emptyset}} m(a) \\
&= 1 - \text{Pl}(A^c)
\end{aligned}$$

Because Pl is completely determined by Bel through complementation, we will refer to $\mathfrak{M}(\text{Bel}, \text{Pl})$ simply as $\mathfrak{M}(\text{Bel})$.

Belief functions possess interesting properties, making the credal set they induce both coherent and avoiding sure loss, motivating their extensive usage.

Example 5: *Defining mass and belief functions*

Let us imagine an experiment where you try to estimate the pressure of the tires of your bike, but your bicycle pump only have graduation every 1 bar. You are able to do three measurements:

- The first measure, the needle seems to be between 4 and 5 bar.
- The second measure, the needle seems to be between 4.5 and 6 bar.
- The third measure, the needle seems to be between 4.5 and 5.5 bar.

Let us say that you trust your last measurement the most, because you got used to the movement of the needle. It is now possible to model the uncertainty of your tire pressure using available evidence, encoded in m:

$$m([4,\ 5]) = 0.3 \qquad m([4.5,\ 6]) = 0.3 \qquad m([4.5, 5.5]) = 0.4$$

Based on this, we are now able to express our degree of belief and of plausibility for all events. For instance:

- Our degree of belief that the pressure lies in $[4, 5]$ is $\text{Bel}([4, 5]) = 0.3$, that it lies in $[4, 5.5]$ is $\text{Bel}([4, 5.5]) = 0.7$ and that it lies in $[4, 6]$ is $\text{Bel}([4, 6]) = 1$
- The degree of plausibility that the pressure equals 5 is $\text{Pl}(5) = 1$ (totally plausible), that it equals 5.5 is $\text{Pl}(5.5) = 0.7$ and that it equals 4 is $\text{Pl}(4) = 0.3$.

2.3.4 Possibility Distributions

Other convenient models of uncertainty are possibility distributions, which are a specific case of imprecise probabilities, as presented in Figure 2.1. We will see that they induce a particular type of belief functions, and will be used in our applications.

Definition 11: Possibility distribution

Let $\mathcal{X}$ be the set of possible outcomes. A possibility distribution is a function $\pi : \mathcal{X} \to [0, 1]$ satisfying:

$$\exists x \in \mathcal{X}, \pi(x) = 1 \tag{2.9}$$

The value $\pi(x)$ represents the degree of possibility of x, with $\pi(x) = 1$ indicating full possibility, and $\pi(x) = 0$ indicating impossibility.

Another notion closely related to possibility distributions is that of α-cuts:

Definition 12: α-cut

Let $\pi : \mathcal{X} \to [0, 1]$ be a possibility distribution. Given any $\alpha \in [0, 1]$, we define the α-cut of π as:

$$\alpha_\pi = \{x \mid \pi(x) \geqslant \alpha\} \tag{2.10}$$

An α-cut is thus the set of all elements of $\mathcal{X}$ whose possibility level is more than α.

Definition 13: Necessity and Possibility measures

It has been proven in [Dubois and Prade, 1992] that a possibility distribution defines a specific type of plausibility functions called *possibility* function and noted Π. It also defines a specific belief function by duality called *necessity* function and noted Nec, as well as a credal set $\mathfrak{M}(\pi)$. They are defined as:

$$\Pi(A) = \sup_{x \in A} \pi(x) \tag{2.11}$$

$$\text{Nec}(A) = 1 - \sup_{x \in A^c} \pi(x) \tag{2.12}$$

$$\mathfrak{M}(\pi) = \{P \mid \forall A,\ P(A) \leqslant \sup_{x \in A} \pi(x)\} \tag{2.13}$$

$$= \{P \mid \forall A,\ P(A) \leqslant \Pi(A)\} \tag{2.14}$$

$$= \{P \mid \forall A,\ P(A) \geqslant \text{Nec}(A)\} \tag{2.15}$$

Example 6: *Defining a possibility distribution*

Let us imagine the same setting as Example 5, where we try to estimate the pressure of the tires of our bike, but our bicycle pump only has graduations every 1 bar. We are able to do the following measurements:

- During the first measurement, the needle seems to be around 4 bar.
- During the second measurement, the needle seems to be around 5 bar.
- During the third measurement, the needle seems to be around 4.5 bar.

Let us say that we trust our measurement with a precision of ± 0.5 bars. For simplicity, we also only consider pressure values that are integers or half integers. Taking into

considerations all the measurements, we can define a possibility distribution π as:

- The value of 4.5 being the most possible, $\pi(4.5) = 1$.
- Values between 4 and 5 bar being highly possible, we can for instance fix $\pi(4) = \pi(5) = 0.8$
- Values 3.5 and 5.5 are unlikely but not impossible, we can say that $\pi(3.5) = \pi(5.5) = 0.3$
- Other values are impossible, thus they have a possibility of 0.

The possibility distribution π is represented in Figure 2.2. Based on this, we are now able to express the degree of necessity and possibility for all events. For instance:

- The degree of necessity that the pressure lies between 4 and 5 bar is $\text{Nec}([4, 5]) = 1 - \sup_{\rho \notin [4,5]} \pi(\rho) = 0.7$
- The degree of necessity that the pressure lies between 3.5 and 5.5 bar is:

  $$\text{Nec}([3.5,\ 5.5]) = 1 - \sup_{\rho \notin [3.5,5.5]} \pi(\rho) = 1$$

  meaning that the pressure is necessarily in this range.
- The degree of possibility that the pressure is either 3.5 or 5 is $\Pi(\{3.5,\ 5\}) = \sup_{\rho \in \{3.5,\ 5\}} \pi(\rho) = 0.9$ (mostly possible).
- *etc.* for every possible event.

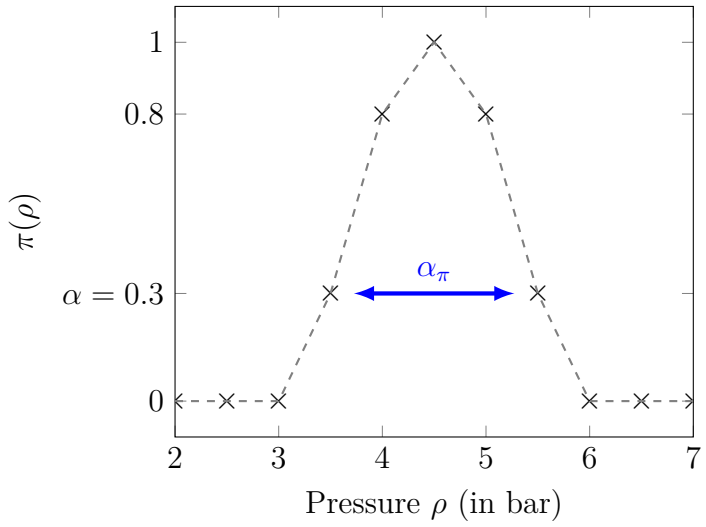


Figure 2.2: Possibility distribution of Example 6 and one of its α-cut in blue

Remark: If you are familiar with fuzzy sets, you may have noticed that possibility distributions are similar to membership functions of a fuzzy set. Links between fuzzy sets and possibility measures have been explored in [Zadeh, 1999].

Focal sets of necessity functions can be determined directly from the possibility distribution by looking at their α-cuts. It has been proven that the core $\mathcal{C}$ (the set containing all focal sets) of a necessity function is ([Couso et al., 2001]):

$$\begin{aligned}\mathcal{C} =&\{\alpha_\pi \mid \alpha \in [0,1]\}\\ =&\{\ \{x \in \mathcal{X} \mid \pi(x) \geqslant \alpha\} \mid \alpha \in [0,1]\ \}\end{aligned}$$

With the way focal sets are defined, they form a nested family of sets with regard to inclusion. Indeed, if an element of $\mathcal{X}$ belongs to an α-cut, then its possibility is greater than α and therefore belongs to any other α'-cut with a lower α'. For simplicity, we will assume that the focal sets $a_1, \ldots, a_n$ are already numbered using the inclusion order, *i.e.* $a_1 \subset \ldots \subset a_n$. In this case, we will refer to the inclusion order as the "natural" order for possibility distributions.

Remark: The fact that focal sets form a nested family of sets in the case of possibility distributions also implies that if $\mathcal{X}$ is finite and contains n elements, then there can be *at most* n focal sets. For comparison, belief functions can have a maximum of $2^n - 1$ focal sets (as the empty set cannot be a focal set). This means that necessity functions have fewer degrees of freedom than (some) belief functions, and thus can express fewer uncertainty structures. This drawback comes with the advantage of being more straightforward to construct, as we only need to specify the mass of n focal sets (or the possibility of the n elements of $\mathcal{X}$) instead of $2^n - 1$. Indeed, when we think of a random variable like the outcome of a die, it can seem more natural for someone to specify degrees of possibility for each side separately than it is to specify degrees of plausibility for different sets of outcomes.

As such, possibility distribution have been used to model experts' opinion in domain such as water contamination [Bárdossy et al., 1995], soil contamination and radioactive risk assessment [Baudrit and Dubois, 2005, Baudrit, 2005, Baudrit et al., 2007] or weather forecasting [Le Carrer and Ferson, 2021]. Following the same philosophy, we will use possibility distributions in Chapter 5 to model the uncertainty of a measure of similarity between two image patches.

Specifying a probability distribution often comes down to specifying the probabil-

ity mass function over all atoms of the set of possible outcomes. In a way, possibility distributions are constructed the same way, as we specify the possibility (or the upper bounds) of every atom. One important difference is that the condition "the sum of all masses must be equal to 1" is relaxed into a less constraining condition "the possibility distribution must be equal to 1 at least once". In that respect, it is easier to construct a well-defined possibility distribution than it is to construct a well-defined probability distribution. However, the comparison stops there, as the two models does not represent the same type of uncertainty.

> **Remark:** Any probability distribution P is a belief function Bel, for which focal sets are only composed of singletons (atoms) and the mass distribution function of Bel equals the probability mass function of P on atoms. However, a possibility distribution cannot model a probability distribution. Indeed, this would impose that its necessity Nec and plausibility Π functions verify:
>
> $$\begin{aligned} &\forall A,\ \mathrm{Nec}(A) = \Pi(A) \\ &\Leftrightarrow 1 - \sup_{x \in A^c} \pi(x) = \sup_{x \in A} \pi(x) \\ &\Leftrightarrow \sup_{x \in A} \pi(x) + \sup_{x \in A^c} \pi(x) = 1 \end{aligned}$$
>
> Consider this equation for any x' verifying $\pi(x') = 1$. This leads to the conclusion that any $x \neq x'$ has a possibility of 0, implying that it is impossible for a random set or random variable to take any other value than x', rendering it not-random.

2.3.5 P-boxes

Another special type of belief function that is commonly used is that of *probability boxes*, more commonly called p-boxes. Formally, a p-box is a pair of precise cumulative distribution functions $[\underline{F},\ \overline{F}]$ defining lower and upper bounds on all cumulative events:

Definition 14: P-box

Let $\mathcal{X}$ be the set of possible outcomes. A p-box is a pair of CDFs $[\underline{F},\ \overline{F}]$ from $\mathcal{X}$ to $[0,\ 1]$ such that:

$$\forall x \in \mathcal{X},\ \underline{F}(x) \leqslant \overline{F}(x) \tag{2.16}$$

If $\mathcal{X}$ is not a subset of $\mathbb{R}$, then there must exist a total order on $\mathcal{X}$ to define a generalized p-box [Destercke et al., 2008].

Remark: A probability distribution can be both determined by specifying its values on every atom or by specifying its values on cumulative events. We saw in Section 2.3.4 that possibility distributions are defined by bounds on atoms. Because p-boxes are defined by bounds on cumulative events, we could say that a p-box is the "imprecise way" of defining a probability using cumulative events, and a possibility distribution is the "imprecise way" of defining a probability using atoms.

The credal set $\mathfrak{M}$ induced by a p-box $[\underline{F},\ \overline{F}]$ is:

$$\mathfrak{M}([\underline{F},\ \overline{F}]) = \{\ F \mid \forall x \in \mathcal{X},\ \underline{F}(x) \leqslant F(x) \leqslant \overline{F}(x)\ \} \tag{2.17}$$

Definition 15: Focal sets of p-boxes

P-boxes are special cases of belief functions. It has been proven in [Destercke et al., 2008] that focal sets of p-boxes have a specific form. Although focal sets shapes are reminiscent of possibilities' α-cuts (see Figure 2.3), they are a bit more complex to express formally. If $\mathcal{X} = \{x_1,\ \ldots,\ x_n\}$ with $x_1 \leqslant \ldots \leqslant x_n$, then focal sets $\alpha_{[\underline{F},\ \overline{F}]}$ of $[\underline{F},\ \overline{F}]$ are given for every $\alpha \in [0,\ 1]$ by the following expression:

$$\alpha_{[\underline{F},\ \overline{F}]} = [\![\overline{F}^{-1}(\alpha),\ \underline{F}^{-1}(\alpha)]\!] \tag{2.18}$$

where $[\![\cdot,\ \cdot]\!]$ are intervals of integers, and $\overline{F}^{-1}$, $\underline{F}^{-1}$ are the respective pseudo-inverse of $\overline{F}$ and $\underline{F}$ defined for every $\alpha \in [0, 1]$ by:

$$\overline{F}^{-1}(\alpha) = \min\{x_i\ s.t.\ \overline{F}(x_i) \geqslant \alpha\}$$
$$\underline{F}^{-1}(\alpha) = \min\{x_i\ s.t.\ \underline{F}(x_i) \geqslant \alpha\}$$

Still in [Destercke et al., 2008]), it has been shown that the mass of each focal set $[\![x_i,\ x_j]\!]$ equals to :

$$m([\![x_i,\ x_j]\!]) = \min(\overline{F}(x_i), \underline{F}(x_j)) - \max(\overline{F}(x_{i-1}), \underline{F}(x_{j-1})) \tag{2.19}$$

With the convention that $\max(\overline{F}(x_{i-1}), \underline{F}(x_{j-1})) = 0$ if x_i is the first element and thus x_{i-1} is ill-defined.

Because of their shape, focal sets of p-boxes can be totally ordered through the so-called lattice ordering on $\mathcal{X}$. It can be easily observed by looking at Figure 2.3. If a and

b are two focal sets of the same p-box $[\underline{F},\ \overline{F}]$, then they are ordered as follows:

$$a \leqslant b \Leftrightarrow \min(a) \leqslant \min(b) \text{ and } \max(a) \leqslant \max(b) \tag{2.20}$$

As $\underline{F} \leqslant \overline{F}$, we are assured that there cannot be any case where $\min(a) < \min(b)$ and $\max(a) > \max(b)$. We can also define the order of focal sets using the definition of Equation (2.18) for every $\alpha,\ \beta \in [0,1]^2$:

$$[\![\overline{F}^{-1}(\alpha),\ \underline{F}^{-1}(\alpha)]\!] \leqslant [\![\overline{F}^{-1}(\beta),\ \underline{F}^{-1}(\beta)]\!] \Leftrightarrow \alpha \leqslant \beta$$

Given the shape of focal sets, there can be at most $2n-1$ of them in the set of n possible outcomes. This is more degree of freedom than possibility distributions, but less than general belief functions.

> **Remark:** Contrary to possibility distributions that cannot equal to a single probability distribution, if a p-box $[\underline{F},\ \overline{F}]$ verifies $\underline{F} = \overline{F}$, then its credal set is composed of a single probability distribution whose CDF F equals $\underline{F}$ and $\overline{F}$. P-boxes are thus generalizations of precise probability distributions.

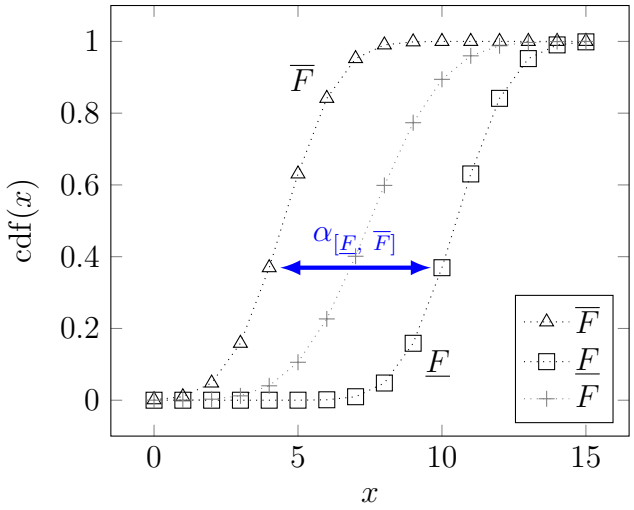


Figure 2.3: A p-box $[\underline{F},\ \overline{F}]$, a precise CDF F in its credal set, and one of its focal elements $\alpha_{[\underline{F},\ \overline{F}]}$ in blue

2.4 Dependency Models: Copulas

During previous sections, we presented different models of uncertainty that will be considered throughout this thesis. When we will be aggregating and propagating uncertainty over multiple variables in Chapters 3 and 4, we will need to take into account the dependencies between our uncertain variables. In this section, we will present dependency

models known as copulas, which are mathematical tools used to represent the dependency between multiple random variables. Copulas can represent many types of dependency, ranging from complete monotonicity to complete counter-monotonicity, including independence between variables. Section 2.4.1 will present the mathematical definition of a copula, as well as practical families of copulas and how they can model different dependencies. Finally, we present how to generate multivariate samples from a copula, which will be used in Section 4.3.2.

2.4.1 Core Definitions and Examples

In the following, let $n \in \mathbb{N}^*$ be the number of uncertainty variables considered (either represented by random variables or random sets). We first introduce copulas, which are mapping from $[0,1]^n \to [0,1]$ verifying a number of properties, and that can model dependencies when considering Sklar's Theorem, which will be presented in this section.

Definition 16

A copula is a multivariate cumulative distribution function $C : [0,1]^n \to [0,1]$ whose marginals follow uniform distributions on $[0,1]$. It can be interpreted as a joint cumulative distribution of n random variables. For all $i \in [\![1, n]\!]$, we will refer to $u_i \in [0,1]$ as its i-th variable (or marginal). A copula satisfies a number of properties:

$$\text{if } \exists j \in [\![1,n]\!] \ s.t. \ u_j = 0, \ \text{then } C(u_1, \ \ldots, \ u_j, \ \ldots, \ u_n) = 0 \tag{2.21}$$

$$\forall i \in [\![1,n]\!], \ C(1, 1, \ \ldots, \ 1, u_i, 1, \ \ldots, \ 1) = u_i \tag{2.22}$$

$$\forall (v_1, \ \ldots, \ v_n) \in [0,1]^n \ s.t. \ \forall i \in [\![1,n]\!], \ v_i \geqslant u_i$$
$$\sum_{(w_1, \ \ldots, \ w_n) \in \prod_{i=1}^{n} \{u_i, v_i\}} (-1)^{|\{i \ | \ w_i = u_i\}|} C(w_1, \ \ldots, \ w_n) \geqslant 0 \tag{2.23}$$

where $\prod_{i=1}^{n}$ is the Cartesian product of n elements, meaning that $(w_1, \ \ldots, \ w_n) \in \prod_{i=1}^{n} \{u_i, v_i\}$ is a tuple of n elements, where each element is either u_i or v_i. Additionally, $|\{i \mid w_i = u_i\}|$ refers to the cardinal of the set $\{i \mid w_i = u_i\}$. An interpretation of the value of Equation (2.23) is presented in the next remark.

The first term in Equation (2.23) is also called H-volume or hyper-volume. It is used to compute joint probability mass assignments in the precise case (and also in the imprecise case, see Section 3.1.2). We will now use the following notation to refer to the H-volume:

$$\forall i \in [\![1,n]\!], \ \forall \ 0 \leqslant u_i \leqslant v_i \leqslant 1,$$
$$H_{u_1, \ \ldots, \ u_n}^{v_1, \ \ldots, \ v_n} = \sum_{(w_1, \ \ldots, \ w_n) \in \prod_{i=1}^{n} \{u_i, v_i\}} (-1)^{|\{i \ | \ w_i = u_i\}|} C(w_1, \ \ldots, \ w_n) \tag{2.24}$$

Remark: The formula of the H-volume actually represents the probability that n-uniform random variables are in the hyper rectangle $[u_1, v_1] \times \cdots \times [u_n, v_n]$. However, it is difficult to see this interpretation in the general case just by looking at the formula. For simplicity, consider the two-dimensional case. Using the interpretation of a copula C as a CDF, we can image two random uniform variables U_1 and U_2 on $[0, 1]$ for which C is their CDFs. We thus have for all $(u_1, u_2) \in [0, 1]^2$:

$$P(U_1 \leqslant u_1, U_2 \leqslant u_2) = C(u_1, u_2)$$

Let $(u_1, u_2) \in [0, 1]^2$ and $(v_1, v_2) \in [0, 1]^2$ *s.t.* $u_1 \leqslant v_1$ and $u_2 \leqslant v_2$. Computing the H-volume of C between (v_1, v_2) and (u_1, u_2) yields:

$$\begin{aligned} H_{u_1,u_2}^{v_1,v_2} &= C(v_1, v_2) - C(v_1, u_2) - C(u_1, v_2) + C(u_1, u_2) \\ &= P(U_1 \leqslant v_1,\ U_2 \leqslant v_2) - P(U_1 \leqslant v_1,\ U_2 \leqslant u_2) \\ &\quad - P(U_1 \leqslant u_1,\ U_2 \leqslant v_2) + P(U_1 \leqslant u_1,\ U_2 \leqslant u_2) \\ &= P(U_1 \leqslant v_1,\ u_2 < U_2 \leqslant v_2) - P(U_1 \leqslant u_1,\ u_2 < U_2 \leqslant v_2) \\ &= P(u_1 < U_1 \leqslant v_1,\ u_2 < U_2 \leqslant v_2) \end{aligned} \tag{2.25}$$

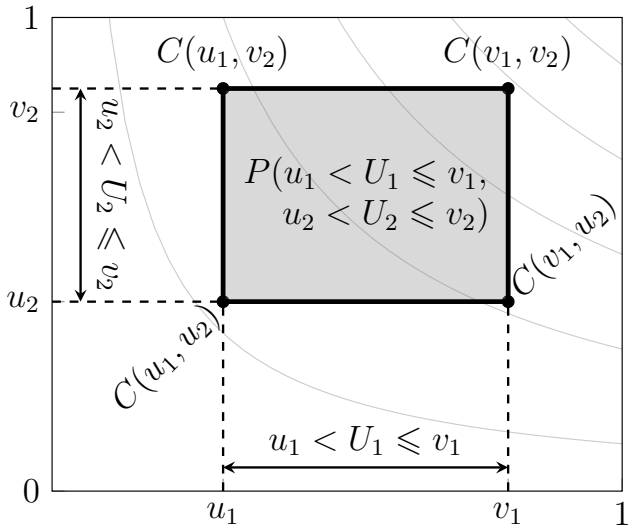


Figure 2.4: Schematic representation of the H-volume

Equation (2.25) means that the H-volume represent the probability of the event

$$u_1 < U_1 \leqslant v_1,\ u_2 < U_2 \leqslant v_2$$

or in other words, the probability that (U_1, U_2) is in the hyper rectangle $[u_1, v_1] \times [u_2, v_2]$ (the intervals can be open or closed in the continuous case, the probability remains the same). Verifying this result for the n-dimensional case can be done

similarly. Example 7 illustrates how the H-volume can be used to compute the discrete joint mass distribution function in the two-dimensional case.

A central theorem regarding copulas is Sklar's Theorem [Sklar, 1959]:

Theorem 1: Sklar's Theorem

Let $F : \mathcal{X}_1 \times \cdots \times \mathcal{X}_n \to [0, 1]$ be a multivariate cumulative distribution function, where $\mathcal{X}_i \subseteq \overline{\mathbb{R}}$. The marginals F_i of F are defined as $\forall i \in [\![1, n]\!], \forall x \in \mathcal{X}_i, F_i(x) = F(+\infty, \ \ldots, \ +\infty, x, +\infty, \ \ldots, \ +\infty)$ where x is the i-th component of F. If all F_i are continuous, then a unique copula C exists:

$$\forall (x_1, \ \ldots, \ x_n) \in \overline{\mathbb{R}}^n, F(x_1, \ \ldots, \ x_n) = C(F_1(x_1), \ \ldots, \ F_n(x_n)) \tag{2.26}$$

If some F_i are not continuous, then C is unique on the product of the ranges of all F_i.

The reverse is also true: any copula applied to univariate cumulative distribution functions yields a multivariate cumulative distribution function whose marginals are the univariate CDFs.

Sklar's Theorem thus allow to express any multivariate CDF by means of its marginal CDFs. Conversely, we can join multiple CDFs with a copula to create a multivariate CDF. A copula thus expresses the dependency between a multivariate CDF and its marginals.

Remark: For marginals F_i that are not continuous, then there can exist multiple copula C verifying Equation (2.26). However, if we note $F_i(\mathcal{X}_i)$ the image of X_i through F_i, then there exists a unique copula C on the ranges of images $F_1(\mathcal{X}_1) \times \cdots \times F_n(\mathcal{X}_n)$. The restriction of a copula to a subset of I^n containing 0 and 1 is called a sub-copula. Because we work in discrete spaces, we will mostly work with sub-copulas, but the difference will be mostly transparent.

Copulas are very useful to represent the dependencies between multiple uncertain variables. As such, they can play a key role in uncertainty propagation problems, explored in Chapter 4.

Example 7: *Usefulness of the H-Volume*

This example will illustrate how the H-volume of a copula is used to compute the

probability mass function of a multivariate probability. Let us imagine a game where a dealer throws two coins, and we are interested in the joint result of the throws. We consider the two random variables X_1 and X_2 indicating the results of each throw:

$X_1 = 0$ if the first coin is heads, otherwise $X_1 = 1$
$X_2 = 0$ if the second coin is heads, otherwise $X_2 = 1$

Let P_1 and P_2 be the probability distributions of X_1 and X_2 respectively, and assume that both heads and tails are possible outcomes for both coins. We now consider the joint probability distribution P associated with the random variable $(X_1,\ X_2)$, and we want to compute the probability mass distribution of P. We denote F_1, F_2 and F the respective CDFs of P_1, P_2 and P. With this definition, F_1 and F_2 are the marginals of F, and Sklar's theorem states that there exists a copula C such that $F = C(F_1,\ F_2)$.

The probability mass distribution of P can be computed by using the H-volume. For instance, let us start by computing $P(1, 1)$. By noticing the fact that:

$$\{x \in \mathcal{X} \mid X_1(x) = 1\} = \{x \in \mathcal{X} \mid F_1(X_1(x)) = F_1(1)\}$$

we can write that:

$$\begin{aligned} P(X_1 = 1, X_2 = 1) &= P(F_1(X_1) = 1, F_2(X_2) = 1) \\ &= P(F_1(0) < F_1(X_1) \leqslant F_1(1),\ F_2(0) < F_2(X_2) \leqslant F_2(1)) \end{aligned}$$

A common result in statistics states that random variables of the form $U_1 = F_1(X_1)$ and $U_2 = F_2(X_2)$ are uniform on $[0, 1]$. We can thus apply the result from Equation (2.25), which yields

$$\begin{aligned} P(X_1 = 1, X_2 = 1) &= P(F_1(0) < U_1 \leqslant F_1(1),\ F_2(0) < U_2 \leqslant F_2(1)) \\ &= H_{F_1(0),\ F_2(0)}^{F_1(1),\ F_2(1)} \end{aligned}$$

The probability of the atom $(1, 1)$ is therefore equal to the H-volume computed between CDFs (F_1, F_2) at $(1, 1)$ and at $(0, 0)$. Following a similar reasoning, we can compute the probability of every atom and express them as H-volumes:

$$\begin{aligned} P(X_1 = 0, X_2 = 1) &= H_{0,F_2(0)}^{F_1(0),F_2(1)} \\ P(X_1 = 1, X_2 = 0) &= H_{F_1(0),0}^{F_1(1),F_2(0)} \\ P(X_1 = 0, X_2 = 0) &= H_{0,0}^{F_1(0),F_2(0)} \end{aligned}$$

It is possible to generalize our observation: the probability of an atom $(x_1, \ldots, x_n)$ is the H-volume computed between marginals CDFs $(F_1, \ldots, F_n)$ at $(x_1, \ldots, x_n)$ and at the marginal atoms that precedes them. If x_1 is the smallest number, then the marginal CDF F_1 before it equals 0, *etc.* Example 8 presents numerical applications of this example with different copulas.

It follows from (2.23) that a copula is a component-wise increasing mapping. All copulas are actually dominating and dominated by two bounds (called lower and upper Fréchet–Hoeffding bounds):

$$\forall u_i \in [0,1]^n,$$
$$\max(0, 1 - n + \sum_{i=1}^{n} u_i) \leqslant C(u_1, \ldots, u_n) \leqslant \min(u_1, \ldots, u_n) \tag{2.27}$$

The upper bound is a copula, usually called the Minimum copula C_M. It is used to model co-monotonic variables, *i.e.* variables for which high values occur at the same time (or similarly, where low values tend to occur simultaneously). Co-monotonicity implies a maximal covariance between variables.

The lower bound is a copula only in the case $n = 2$, called the Łukasiewicz copula $C_L(u_1, u_2) = \max(0, u_1 + u_2 - 1)$. It is used to model counter-monotonic variables, *i.e.* variables with a perfect negative dependence between them. This explains why the lower bound is not a copula in dimensions higher than 2. Indeed, if X has a perfect negative dependence with Y and Z, then Y and Z cannot share a perfect negative dependence. However, for every $u_1, \ldots, u_n$, there always exists a copula C attaining the lower bound (which can differ for different $u_1, \ldots, u_n$):

$$\forall (u_1, \ldots, u_n) \in [0,1]^n, \ \exists C \ s.t. \ C(u_1, \ldots, u_n) = \max(0, 1 - n + \sum_{i=1}^{n} u_i)$$

Independence between variables is modeled by the product copula C_Π:

$$C_\Pi(u_1, \ldots, u_n) = \prod_{i=1}^{n} u_i = u_1 \cdot \ldots \cdot u_n$$

where "$\cdot$" refers to the product between two scalars, as the symbol $\times$ is already used for the Cartesian product. The product copula will be used later in Section 3.2.1. Graphical representations of the Łukasiewicz, product and Min copulas are displayed in Figure 2.5. Example 8 presents a setting where the Product, Minimum and Łukasiewicz copulas are used to model dependency between random variables.

Example 8: *Different copulas for different dependencies*

Let us try to illustrate how different copulas can represent different dependencies. Consider the same setting as Example 7 with two coins being thrown. For the purpose of the example, assume that the dealer throws the coins in a separate room, and comes back to tell the result. We thus never see if he is cheating or not. He only provides us this piece of information: coins seems fair when looked at separately. We therefore have the following marginals:

$$\begin{cases} P_1(\text{heads}) = P_1(0) = 0.5 \\ P_1(\text{tails}) = P_1(1) = 0.5 \end{cases} \quad \text{and} \quad \begin{cases} P_2(\text{heads}) = P_2(0) = 0.5 \\ P_2(\text{tails}) = P_2(1) = 0.5 \end{cases}$$

- Assume that the dealer is not cheating and that the two coin throws are independent. In that case, the product copula $C_\Pi(u, v) = u \cdot v$ must be used to represent the independence between variables. Using results from the previous example, it holds that:

$$\begin{aligned} P(1,1) =& H_{F_1(1),F_2(1)}^{F_1(0),F_2(0)} = 1 \cdot 1 - 0.5 \cdot 1 - 1 \cdot 0.5 + 0.5 \cdot 0.5 \\ =& 0.25 \\ P(1,0) =& H_{F_1(0),0}^{F_1(1),F_2(0)} = 0.25 \\ P(0,1) =& H_{0,F_2(0)}^{F_1(0),F_2(1)} = 0.25 \\ P(0,0) =& H_{0,0}^{F_1(0),F_2(0)} = 0.25 \end{aligned}$$

  Remark that we indeed find the same results as if we directly multiplied the marginal probability mass distributions: $P(1,1) = P_1(1) \cdot P_2(1)$, *etc.* We thus observe the famous result: if P_1 and P_2 are independent, then $P = P_1 \cdot P_2$.

- Imagine now that the dealer is not being fair, and actually forces the second throw to land on the same side as the first one (the coins will still seem fair when looked at separately). This kind of dependency is modeled by the Minimum copula $C_M(u, v) = \min(u, v)$. In this case, the joint probability is computed as

follows:

$$\begin{aligned}
P(1,1) =& H^{F_1(0),F_2(0)}_{F_1(1),F_2(1)} = \min(1,1) - \min(0.5,1) - \min(1,0.5) + \min(0.5,0.5) \\
=& 0.5 \\
P(1,0) =& H^{F_1(1),F_2(0)}_{F_1(0),0} = \min(1,0.5) - \min(1,0) - \min(0.5,0.5) + \min(0,0.5) \\
=& 0 \\
P(0,1) =& H^{F_1(0),F_2(1)}_{0,F_2(0)} = 0 \\
P(0,0) =& H^{F_1(0),F_2(0)}_{0,0} = \min(0.5,0.5) = 0.5
\end{aligned}$$

The values taken by the joint probability are now completely different from the independence case. We see that values $(0,1)$ and $(1,0)$ are indeed impossible to obtain, while $(1,1)$ and $(0,0)$ are equiprobable.

- Imagine now that the dealer is still not being fair, but this time forces the second coin to land on the first coin's opposite side. In other words, if the first coin lands on heads, then the dealer puts the second coin on tails, and inversely. Looking at marginal distributions separately will still suggest that the coins are fair. However, they appear fully counter-monotone when looked at jointly. In this case, the dependency is modeled by the Łukasiewicz copula $C_L(u,v) = \max(0, u+v-1)$, and the joint probability equals:

$$\begin{aligned}
P(1,1) =& H^{F_1(0),F_2(0)}_{F_1(1),F_2(1)} = \max(0,1+1-1) - \max(0,0.5+1-1) \\
& - \max(0,1+0.5-1) + \max(0,0.5+0.5-1) \\
=& 0 \\
P(1,0) =& H^{F_1(1),F_2(0)}_{F_1(0),0} = \max(0,1+0.5-1) - \max(0,1+0-1) \\
& - \max(0,0.5+0.5-1) + \max(0,0+0.5-1) \\
=& 0.5 \\
P(0,1) =& H^{F_1(0),F_2(1)}_{0,F_2(0)} = 0.5 \\
P(0,0) =& H^{F_1(0),F_2(0)}_{0,0} = \max(0,0.5+0.5-1) = 0
\end{aligned}$$

The values taken by the joint probability is now completely different than in the other cases. We see that values $(1,1)$ and $(0,0)$ are indeed impossible to obtain, while $(0,1)$ and $(1,0)$ are equiprobable.

Those three cases indicate how copulas can represent very different dependency structures from the same marginals. It also makes it intuitive that the Łukasiewicz copula only allows values that are "opposite", whereas the Minimum copula only allows values that are similar. In those examples, the dependency is so important that knowing

the result of one coin throw determines the result of the second, which is therefore modeled by extreme copulas, *i.e.* the upper and lower Fréchet-Hoeffding bounds. We will see in the following that there are other families of copula that allow less "extreme" dependencies.

To complete this overview of copulas, let us present other copulas that can be generated using a single parameter θ in the case $n = 2$. Some famous families of copulas are presented in Table 2.1. Those families are quite common in the literature, but this list is not exhaustive.

Another important family of copulas is the family of Gaussian copulas. Each Gaussian copula is generated with a correlation matrix $R \in [-1, 1]^{(n,n)}$:

$$C_R(u_1, \ \ldots, \ u_n) = \Phi_R(\Phi^{-1}(u_1), \ \ldots, \ \Phi^{-1}(u_n)) \tag{2.28}$$

where Φ_R is the joint multivariate cumulative distribution function of a Gaussian variable with correlation matrix R, and Φ^{-1} is the inverse cumulative distribution function of a univariate Gaussian variable. We do not know an exact form for Φ_R, but we can compute it by integrating its associated Gaussian PDF:

$$\Phi_R(\Phi^{-1}(u_1), \ \ldots, \ \Phi^{-1}(u_n)) = \int_{-\infty}^{\Phi^{-1}(u_1)} \ldots \int_{-\infty}^{\Phi^{-1}(u_n)} \frac{1}{\sqrt{(2\pi)^n |R|}} \exp\left(-\frac{1}{2} \begin{bmatrix} x_1 & \ldots & x_n \end{bmatrix} R^{-1} \begin{bmatrix} x_1 \\ \ldots \\ x_n \end{bmatrix} \right) dx_1 \ldots dx_n \tag{2.29}$$

Where $|R|$ is the determinant of R. This family of copulas will be used in Section 4.1 to model the dependency between the random intensities of pixels of stereo images, for instance.

Remark: The product copula is actually a Gaussian copula with the identity matrix

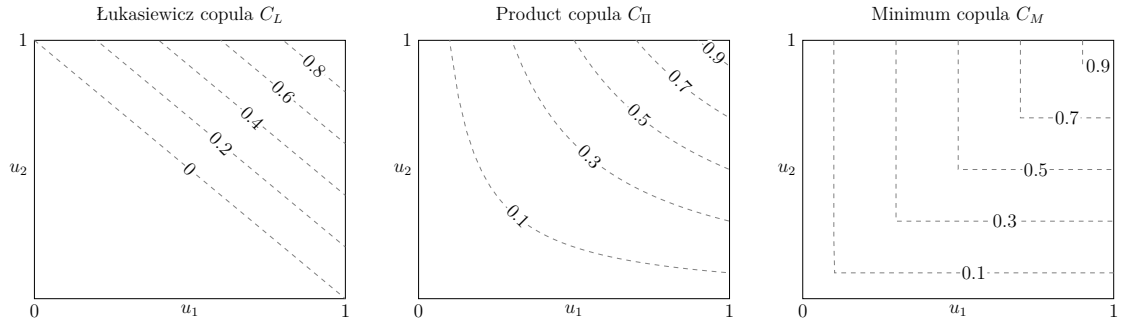


Figure 2.5: Bird view of the Łukasiewicz, product and Min copulas for $n = 2$. Dashed gray lines represent isolines of the copulas.

$\mathbb{I}_n$ as its covariance matrix:

$$
\begin{aligned}
&\Phi_R(\Phi^{-1}(u_1),\ \ldots,\ \Phi^{-1}(u_n)) = \\
&\quad \int_{-\infty}^{\Phi^{-1}(u_1)} \cdots \int_{-\infty}^{\Phi^{-1}(u_n)} \frac{1}{\sqrt{(2\pi)^n |\mathbb{I}_n|}} \exp\left(-\frac{1}{2} \begin{bmatrix} x_1 & \ldots & x_n \end{bmatrix} \mathbb{I}_n^{-1} \begin{bmatrix} x_1 \\ \ldots \\ x_n \end{bmatrix} \right) dx_1 \ldots dx_n \\
&= \int_{-\infty}^{\Phi^{-1}(u_1)} \cdots \int_{-\infty}^{\Phi^{-1}(u_n)} \frac{1}{\sqrt{(2\pi)^n}} \exp(-\frac{1}{2} \sum_{i=1}^{n} x_i^2) dx_1 \ldots dx_n \\
&= \int_{-\infty}^{\Phi^{-1}(u_1)} \frac{1}{\sqrt{(2\pi)}} \exp(-\frac{1}{2} x_1^2) dx_1 \ldots \int_{-\infty}^{\Phi^{-1}(u_n)} \frac{1}{\sqrt{(2\pi)}} \exp(-\frac{1}{2} x_n^2) dx_n \\
&= \int_{-\infty}^{\Phi^{-1}(u_1)} \Phi'(x_1) dx_1 \ldots \int_{-\infty}^{\Phi^{-1}(u_n)} \Phi'(x_n) dx_n \\
&= u_1 \ldots u_n
\end{aligned}
$$

Family	$C(u_1, u_2)$	$\theta \in$	D-convex	D-concave
Ali-Mikhail-Haq	$\frac{u_1 u_2}{1-\theta(1-u_1)(1-u_2)}$	$[-1, 1)$	$\theta \leqslant 0$	$\theta \geqslant 0$
Clayton	$\left[\max(u_1^{-\theta} + u_2^{-\theta} - 1, 0)\right]^{-1/\theta}$	$[-1, \infty)\backslash\{0\}$	$\theta < 0$	$\theta > 0$
Frank	$-\frac{1}{\theta}\ln(1 + \frac{(e^{-\theta u_1}-1)(e^{-\theta u_2}-1)}{e^{-\theta}-1})$	$\mathbb{R}\backslash\{0\}$	$\theta < 0$	$\theta > 0$
Gumbel	$u_1 u_2 \exp(-\theta \ln u_1 \ln u_2)$	$(0, 1]$	$\theta \in (0, 1]$	Never

Table 2.1: Examples of families of copulas in the case $n = 2$ which can be generated using a parameter θ. D-convexity/concavity is detailed in the Annex Section 7.1

Remark: As a copula is also a multivariate CDF, one can imagine an "imprecise" copula [Montes et al., 2015] similarly to what can be done with univariate probability

distribution (see Section 2.3.2). Imprecise copulas allow modeling partially known dependencies, but can be hard to manipulate at times, as for instance the lower and upper bounds of an imprecise copula are not necessarily copulas themselves. In this thesis, we will not consider imprecise copulas.

2.4.2 Sampling from a Copula

As a copula represents the CDF of a multivariate random variable, it is possible to sample from it. This section details a method for sampling from copulas in general, and a special method for sampling from Gaussian copulas. Chapter 4 will use those methods for Monte Carlo sampling. For simplicity, let us first present a method for sampling in the case $n = 2$. Given a copula C, and two CDFs F_X and F_Y, a method to generate a pair of observations (x, y) from a joint CDF $C(F_X, F_Y)$ is the following:

- Sample two independent samples u_1, u_2 from a uniform distribution on [0,1]
- Set $v = \partial C^{-1}(u_2)$ where ∂C^{-1} is the quasi-inverse of the partial derivative of C with respect to its first variable (which exists almost everywhere and is invertible).
- We now have a sample (u_1, v) from a multivariate random variable. Its marginals follow a uniform distribution on $[0, 1]$, and its associated copula is C.
- The desired pair is $(x, y) = (F_X^{-1}(u_1), F_Y^{-1}(v))$, with F_X^{-1}, F_Y^{-1} being the quasi-inverses of the marginals CDFs.

We do not present the n-dimensional general case here as it is a bit more complex, but it can be found in [Cherubini et al., 2004]. However, drawing samples from a Gaussian n-copula with a correlation matrix R are simpler to obtain:

- Compute the Cholesky decomposition A of the correlation matrix R
- Draw n independent random samples $u = (u_1, \ \ldots, \ u_n)'$ from $\mathcal{N}(0, 1)$, where $\mathcal{N}$ is the normal distribution.
- Set $v = Au$
- Set $w_k = \Phi(v_k)$ where Φ is the univariate normal cumulative distribution function
- The desired draw is $(x_1, \ \ldots, \ x_n) = (F_1^{-1}(w_1), \ \ldots, \ F_n^{-1}(w_n))$ with F_i^{-1}, being the quasi-inverse of the i-th marginal CDF.

Conclusion

In this chapter, we presented different uncertainty models, whose usage depends on the type of uncertainty and evidence available. Choosing between the different models is often a trade-off between expressivity, compatibility and "realism" of the hypotheses of the considered problem. We also presented a way to combine probabilities by taking into account the dependency between random variables, using copulas. Copulas are well suited for probabilities, but their usage with IP is more complex. Chapter 3 will investigate how IP models can be combined using a copula, and Chapter 4 will use copulas and imprecise models in a stereo matching problem.

Chapter 3

Using Copulas to Join Credal Sets

In the presence of multiple uncertain variables, the question of how to join them arises. Using probability distributions, the usual method for joining probability distributions is to use copulas (Section 2.4). Indeed, Sklar's Theorem provides a well-defined way of uniquely joining multiple continuous CDF with a copula. However, this is no longer true when adding imprecision to the models, as there are multiple ways of specifying the dependency between imprecise models using a copula. Sections 3.1.1 to 3.1.3 present three different ways of joining credal sets using a copula. Those three methods are not equivalent. We thus explore their similarities and differences in Section 3.2. Methods will be used to join possibility distributions to propagate the uncertainty in Chapter 4.

3.1 Methods for Joining Credal Sets with Copulas

This section will use copulas, introduced in Section 2.4, to join imprecise models in three different ways. The way we prove the existence of copulas and their usage with Sklar's Theorem was based on CDFs [Nelsen, 2006]. However, the IP model closest to CDFs, p-boxes from Section 2.3.5, does not allow representing every credal set as seen in Figure 2.1. When working with marginals modeled by Imprecise Probabilities, we will thus define different methods for using IP with copulas. The first approach in Section 3.1.1 maintains the classical interpretation of a copula found in Sklar's Theorem. The multivariate credal set obtained with this method is however hard to handle computationally. A similar approach using only the product copula can be already be found in [Couso et al., 2000]. The second approach in Section 3.1.2 takes some distances with the classical interpretation of a copula, but is easier to handle computationally. It is based on previous work detailed in [Ferson et al., 2004]. Finally, we introduce a third approach in Section 3.1.3, which completely abandons Sklar's Theorem interpretation, but is very easy to handle. Inclusion relationships between those three approaches are explored in Section 3.2, in order to

determine which is an approximation of the other.

We consider here $n \in \mathbb{N}^*$ uncertain variables $(X_i)_{1 \leqslant i \leqslant n}$ taking values respectively in a totally ordered finite space $\mathcal{X}_i$. The index i will usually refer to the i-th random variable (or random set). We denote $\mathfrak{M}_i$ as the credal set representing the uncertainty of X_i, and C a n-copula. We also suppose that $\mathfrak{M}_i$ the lower bound of $\mathfrak{M}_i$ is defined by a belief function. Focal sets of belief functions will be noted a_k^i, where k refers to the k-th focal set, if they are numbered. We also note $\bigsqcup$ the union of disjoint elements. Finally, we must introduce the concept of cylindrical sets, used to specify definition domains of various mappings.

Definition 17: Cylindrical sets

Let $\mathcal{X}_1, \ldots, \mathcal{X}_n$ be n sets and let $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$ be the Cartesian product of $\mathcal{X}_1, \ldots, \mathcal{X}_n$. We call a cylindrical (or cylinder) set X of $\mathcal{X}$ a set which can be written as a Cartesian product of elements of $\mathcal{X}_1, \ldots, \mathcal{X}_n$, *i.e.*:

$$X \subseteq \mathcal{X} \text{ is cylindrical } \Leftrightarrow \exists (X_1, \ldots, X_n), \ s.t. \ X = X_1 \times \cdots \times X_n \tag{3.1}$$

3.1.1 Point-wise Aggregation

A first way of creating a joint credal set is to first consider every precise marginal within each marginal credal sets $\mathfrak{M}_i$. We then use Sklar's Theorem with the copula C to create a precise multivariate CDF. The set of all resulting CDFs is as follows:

$$\mathcal{S}(C, \mathfrak{M}_i) = \{F \mid \forall x_i \in \mathcal{X}_i, \ F_i \in \mathfrak{M}_i, \ F(x_1, \ldots, x_n) = C(F_1(x_1), \ldots, F_n(x_n))\} \tag{3.2}$$

This set is not guaranteed to be convex [Schmelzer, 2023]. We thus define the joint credal set as the convex hull $\mathfrak{M}_{robust}$ of $\mathcal{S}$

Definition 18: Robust Credal Set

We define the robust credal set obtained by joining n marginal credal sets $\mathfrak{M}_i$ and a copula C as:

$$\mathfrak{M}_{robust}(C, \mathfrak{M}_i) = CH(\{C(F_1, \ldots, F_n), \ F_i \in \mathfrak{M}_i\}) \tag{3.3}$$

where CH is the convex hull presented in Definition 7.

We refer to this joint credal set as $\mathfrak{M}_{robust}(C, \mathfrak{M}_i)$ as it contains every element of the marginal credal sets with copula C as their dependency model. We will omit "$(C, \mathfrak{M}_i)$" when there is no confusion possible to avoid using heavy notations. As we take the convex hull of $\mathcal{S}$, it is interesting to notice that it can contain additional multivariate CDFs that do not possess the copula C as their dependency model, as seen in Example 9. This credal set is usually hard to compute for events that are not Cartesian products of marginal cumulative events. Indeed, copula being non-linear operators, there is no reason that their point-wise application would preserve convexity.

Example 9: Consider two coins, with their associated random variable X_1 and X_2 taking values in $\mathcal{X}_1 = \mathcal{X}_2 = \{heads,\ tails\}$. The uncertainty of each random variable is represented by the following mass distribution function m_1 and m_2:

$$m_1(heads) = 0.4 \qquad m_2(heads) = 0.4$$
$$m_1(tails) = 0.4 \qquad m_2(tails) = 0.5$$
$$m_1(\mathcal{X}_1) = 0.2 \qquad m_2(\mathcal{X}_2) = 0.1$$

The credal sets $\mathfrak{M}_1$ and $\mathfrak{M}_2$ associated with X_1 and X_2 are therefore composed of all the probabilities $P_1 \in \mathfrak{M}_1$ and $P_2 \in \mathfrak{M}_2$ verifying:

$$0.4 \leqslant P_1(heads) \leqslant 0.6 \qquad 0.4 \leqslant P_2(heads) \leqslant 0.5$$
$$0.4 \leqslant P_1(tails) \leqslant 0.6 \qquad 0.5 \leqslant P_2(tails) \leqslant 0.6$$

If we assume we throw the two coins independently, then their dependency can be modeled by the product copula $C_\Pi(u, v) = u \cdot v$.

The robust credal set $\mathfrak{M}_{robust}$ obtained by joining $\mathfrak{M}_1$ and $\mathfrak{M}_2$ with C_Π will therefore be the convex hull of the set:

$$\{P \mid \forall P_1 \in \mathfrak{M}_1, P_2 \in \mathfrak{M}_2, \forall (x_1, x_2) \in \mathcal{X}_1 \times \mathcal{X}_2, \\ P(x_1, x_2) = P_1(x_1) \cdot P_2(x_2)\ \}$$

In this specific case the bounds of $P \in \mathfrak{M}_{robust}$ on cylindrical sets are quite straightforward because we consider binary random variables and the product copula:

$$0.16 \leqslant P(heads, heads) \leqslant 0.3 \qquad 0.2 \leqslant P(tails, tails) \leqslant 0.36$$
$$0.2 \leqslant P(heads, tails) \leqslant 0.36 \qquad 0.16 \leqslant P(tails, heads) \leqslant 0.3$$

Now, let us exhibit a probability from $\mathfrak{M}_{robust}$ that does not have C_Π as its dependency

model. Consider the following marginal probabilities $P_1, P_1' \in \mathfrak{M}_1$ and $P_2, P_2' \in \mathfrak{M}_2$:

$$P_1(heads) = 0.4 \qquad P_2(heads) = 0.4$$
$$P_1'(heads) = 0.6 \qquad P_2'(heads) = 0.5$$

Then $P = P_1 \cdot P_2$ and $P' = P_1' \cdot P_2'$ both belong in $\mathfrak{M}_{robust}$, and the convex mixture $P_{0.5} = 0.5 \cdot P + 0.5 \cdot P'$ is also in $\mathfrak{M}_{robust}$. We will compute the value of $P_{0.5}(heads, heads)$ and show that it does not equal to the product of its marginals:

$$\begin{aligned} P_{0.5}(heads, heads) &= 0.5 \cdot P(heads, heads) + 0.5 \cdot P'(heads, heads) \\ &= 0.5 \cdot P_1(heads) \cdot P_2(heads) + 0.5 \cdot P_1'(heads) \cdot P_2'(heads) \\ &= 0.5 \cdot 0.4 \cdot 0.4 + 0.5 \cdot 0.6 \cdot 0.5 \\ &= 0.08 + 0.15 = 0.23 \end{aligned}$$

$$\begin{aligned} P_{0.5}(heads, \mathcal{X}_2) &= 0.5 \cdot P_1(heads) \cdot P_2(\mathcal{X}_2) + 0.5 \cdot P_1'(heads) \cdot P_2'(\mathcal{X}_2) \\ &= 0.5 \cdot 0.4 + 0.5 \cdot 0.6 \\ &= 0.2 + 0.3 = 0.5 \\ P_{0.5}(\mathcal{X}_1, heads) &= 0.5 \cdot P_1(\mathcal{X}_1) \cdot P_2(heads) + 0.5 \cdot P_1'(\mathcal{X}_1) \cdot P_2'(heads) \\ &= 0.5 \cdot 0.4 + 0.5 \cdot 0.5 \\ &= 0.2 + 0.25 = 0.45 \end{aligned}$$

The product of the marginals on heads is therefore equal to

$$\begin{aligned} P_{0.5}(heads, \mathcal{X}_2) \cdot P_{0.5}(\mathcal{X}_1, heads) &= 0.5 \cdot 0.45 \\ &= 0.225 \\ &\neq 0.23 = P_{0.5}(heads, heads) \end{aligned}$$

therefore $P_{0.5}$ belongs to $\mathfrak{M}_{robust}$ but does not have C_Π as its copula.

Remark: Due to the convex hull, we can both consider $\mathfrak{M}_{robust}$ as only containing probabilities on the set $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$ and containing probabilities on $2^{\mathcal{X}}$. This applies to the other credal sets as well. We will consider the latter, though the former has been considered in the literature (for instance [Schmelzer, 2012, Schmelzer, 2023]). To insist on which marginal events we consider, we will sometimes write $P(a^1, a^2)$ instead of $P(a^1 \times a^2)$, even though this notation suggests that P is defined on $\mathcal{X}$ and not $2^{\mathcal{X}}$.

Figure 3.1 represents a schematic of $\mathfrak{M}_{robust}$ is obtained. First, two samples F_1 and F_2 (represented by "+" signs) are sampled from the marginals credal sets $\mathfrak{M}_1$ and $\mathfrak{M}_2$. They are then joined using the copula C to create a multivariate CDF represented by a "x" sign. This process is then reproduced for every combination of samples in the marginal credal sets and $\mathfrak{M}_{robust}$ is the convex hull of the set of samples.

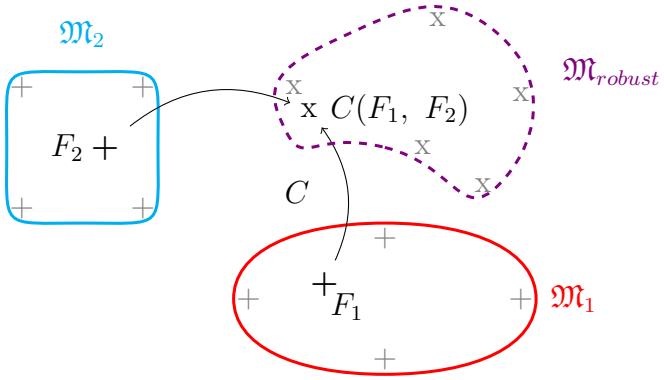


Figure 3.1: Schematic representation of $\mathfrak{M}_{robust}$

3.1.2 Copula Applied to Cumulative Mass Functions

In this section, we will present another way of creating a joint credal set from multiple marginal ones. Consider the same copula C as before and the same marginal credal sets $\mathfrak{M}_i$. Each credal set $\mathfrak{M}_i$ is fully determined by a mass distribution function m_i, which is strictly positive over its N_i focal sets $a_1^i,\ \ldots,\ a_{N_i}^i$. As described in [Ferson et al., 2004], it is possible to use the cumulative mass distribution functions as marginals of the copula to create a joint mass distribution function, granted that there is a complete ordering defined on the focal sets. Links between copulas and belief functions have been investigated in the continuous case in [Schmelzer, 2015a, Schmelzer, 2019], the special case of necessity functions in [Schmelzer, 2015b] and of p-boxes in [Schmelzer, 2023].

Let us assume, without loss of generality, that the marginal focal sets are numbered according to the ordering $\preceq_i$: $a_1^i \preceq_i a_2^i \preceq_i \ldots \preceq_i a_{N_i}^i$. The idea behind this method is to replace the precise marginal CDFs by cumulative masses, to keep the philosophy behind Sklar's Theorem. We thus first define the joint mass m_C on the product space of focal sets as follows:

Definition 19: Joint Mass

Let $m_1,\ \ldots,\ m_n$ be mass distribution functions over their respective power sets of $\mathcal{X}_1,\ \ldots,\ \mathcal{X}_n$. Assume that focal sets in each $\mathcal{X}_i$ are ordered and that $a_{k_i}^i$ is the k_i-th focal set of m_i according to the chosen order. We define m_C as the H-volume of

copula C computed over the cumulative marginal masses:

$$m_C(a^1_{k_1} \times \cdots \times a^n_{k_n}) = H^{\sum_{k=0}^{k_1} m_1(a^1_k),\ ...,\ \sum_{k=0}^{k_n} m_n(a^n_k)}_{\sum_{k=0}^{k_1-1} m_1(a^1_k),\ ...,\ \sum_{k=0}^{k_n-1} m_n(a^n_k)} \tag{3.4}$$

with the convention that $\forall i,\ a^i_0 = \emptyset$. It is not *strictly* a focal set but allows dealing with the case $k_i = 1$ as $m_i(a^i_0) = 0$. For sets that are not of the form $a^1_{k_1} \times \cdots \times a^n_{k_n}$, the mass m_C is null.

Proposition 1: The function m_C defined in Definition 19 is a correctly defined mass distribution function over $\mathcal{X}$.

Proof: To be a mass distribution function over $\mathcal{X}$, m_C must verify the 3 properties of Definition 8.

By construction, it holds that $m_C(\emptyset) = 0$, and the properties of the H-volume impose that $m_C \in [0, 1]$.

There are multiple ways of proving that $\sum_{A \subseteq \mathcal{X}} m_C(A) = 1$. A direct proof can be done in the case $n = 2$, but the notations become quite heavy for any $n > 2$. Instead, let us use the interpretation of a copula as a multivariate CDF. This method will also be used in future proofs.

For all $i \in [\![0,\ n]\!]$ let F_i be a CDF over $[0,\ N_i]$, with $F_i(j) = \sum_{k=0}^{j} m_i(a^i_k)$. By Sklar's Theorem, $F = C(F_1,\ \ldots,\ F_n)$ is a multivariate CDF over $[0,\ N_1] \times \cdots \times [0,\ N_n]$, and P its PDF. Thus, it holds that $P([0,\ N_1] \times \cdots \times [0,\ N_n]) = F(N_1, \ldots, N_n) = 1$

and:

$$
\begin{aligned}
P([0,N_1]\times\cdots\times[0,N_n]) &= P([0,\ \ldots,\ 0]) + \\
&\quad P\left(\bigsqcup_{k_1=0}^{N_1-1}\ldots\bigsqcup_{k_n=0}^{N_n-1}(]k_1,k_1+1]\times\cdots\times]k_n,k_n+1])\right) \\
&= 0+\sum_{k_1=0}^{N_1-1}\ldots\sum_{k_n=0}^{N_n-1}P(]k_1,k_1+1]\times\cdots\times]k_n,k_n+1]) \\
&\quad \text{(CDF of a union of disjoint elements, as in Figure 3.2)} \\
&= \sum_{k_1=0}^{N_1-1}\cdots\sum_{k_n=0}^{N_n-1}H_{F_1(k_1),\ \ldots,\ F_n(k_n)}^{F_1(k_1+1),\ \ldots,\ F_n(k_n+1)} \\
&= \sum_{k_1=1}^{N_1}\cdots\sum_{k_n=1}^{N_n}H_{\sum_{k=0}^{k_1-1}m_1(a_k^1),\ \ldots,\ \sum_{k=0}^{k_n-1}m_n(a_k^n)}^{\sum_{k=0}^{k_1}m_1(a_k^1),\ \ldots,\ \sum_{k=0}^{k_n}m_n(a_k^n)} \\
&= \sum_{(a_{k_1}^1\times\cdots\times a_{k_n}^n)\subseteq\mathcal{X}} m_C(a_{k_1}^1\times\cdots\times a_{k_n}^n)
\end{aligned}
$$

Therefore it holds that:

$$
\sum_{A\subseteq\mathcal{X}} m_C(A) = \sum_{(a_{k_1}^1\times\cdots\times a_{k_n}^n)\subseteq\mathcal{X}} m_C(a_{k_1}^1\times\cdots\times a_{k_n}^n) = 1 \tag{3.5}
$$

which proves that m_C is a mass distribution function.

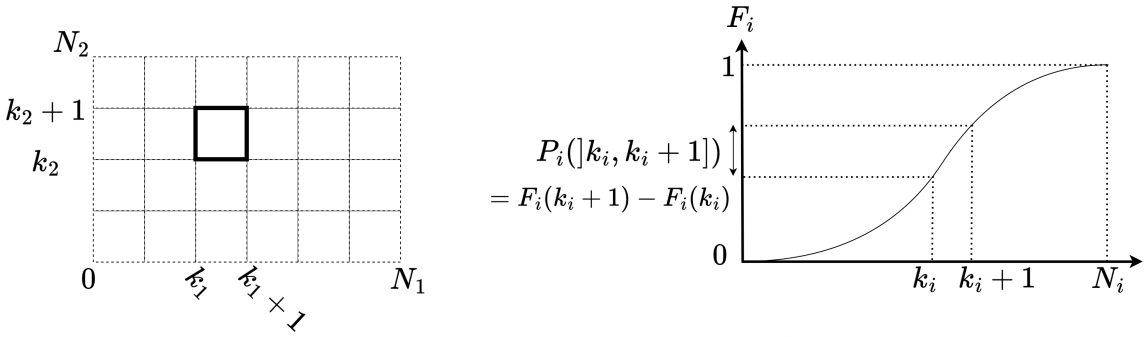


Figure 3.2: Splitting a CDF into a sum of disjoint events in dimension 1

□

Having defined a mass distribution function on the product space $\mathcal{X}$, we thus define the joint credal set $\mathfrak{M}_{mass}$ and its belief function Bel_C as:

Definition 20: Joint Belief Function and its Credal Set

Let m_C be the mass distribution function from Definition 19. We note Bel_C its joint

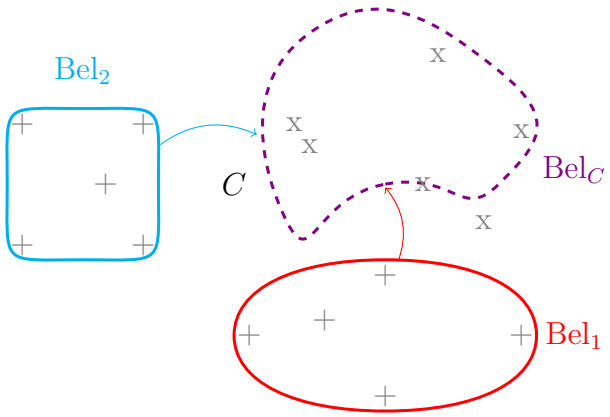


Figure 3.3: Schematic representation of $\mathfrak{M}_{mass}$

belief function over the power set of $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$, *i.e.*:

$$\forall A \subseteq \mathcal{X}, \text{Bel}_C(A) = \sum_{a \subseteq A} m_C(a) \tag{3.6}$$

Because Bel_C is a belief function, it defines a credal set $\mathfrak{M}_{mass}$:

$$\mathfrak{M}_{mass}(C, \mathfrak{M}_i) = \{\ P : 2^{\mathcal{X}} \to [0,1] \mid \forall A \subseteq \mathcal{X}, P(A) \geqslant \text{Bel}_C(A)\ \} \tag{3.7}$$

We only specify the lower bound Bel_C in the expression of the credal set as

$$\begin{aligned} &\text{Bel}_C(A^c) \leqslant P(A^c) \\ \Leftrightarrow\ &1 - \text{Pl}(A) \leqslant 1 - P(A) \\ &\text{Pl}(A) \geqslant P(A) \end{aligned}$$

Figure 3.3 presents a schematic of Bel_C similarly to what was presented with $\mathfrak{M}_{robust}$ in Figure 3.1: Bel_C is computed from Bel_1 and Bel_2. The gray "+" and "X" signs have the same position as in Figure 3.1, which shows that $\mathfrak{M}_{robust}$ and $\mathfrak{M}_{mass}$ are not the same, as the copula is applied to the cumulative masses instead of being applied point-wise to every CDF from marginal sets.

With this way of defining the multivariate mass, the choice of arbitrary orderings $\preceq_i$ can have a significant impact on the value of the multivariate mass function, as we will see in example Example 10. Those orderings will specifically be "natural" orderings in Sections 3.2.2 to 3.2.4, in the sense that there exists an intuitive total ordering (inspired by the ordering of reals). When no natural ordering exists, the arbitrary choice of the ordering can greatly impact the output mass or belief functions, as illustrated by the following example.

Example 10: Let us consider the same setting as Example 9 with two coins, and their mass defined this time as:

$$m_1(heads) = 0.4 \qquad m_2(tails) = 0.7$$
$$m_1(\mathcal{X}_1) = 0.6 \qquad m_2(\mathcal{X}_2) = 0.3$$

We consider the minimum copula $C(u, v) = \min(u, v)$ If there is no natural ordering on the focal sets of m_1, we have to choose an arbitrary one:

- If "$\{heads\} \preceq_1 \mathcal{X}_1$" and "$\{tails\} \preceq_2 \mathcal{X}_2$" are the arbitrary orderings, then

$$\begin{aligned} \text{Bel}_C(heads, tails) &= m_C(heads, tails) \\ &= C\left(m_1(heads),\ m_2(tails)\right) \\ &= \min(0.4, 0.7) \\ &= 0.4 \end{aligned}$$

- If "$\mathcal{X}_1 \preceq_1 \{heads\}$" is the arbitrary order, then

$$\begin{aligned} \text{Bel}_C(heads, tails) &= m_C(heads, tails) \\ &= C\left(1,\ m_2(tails)\right) - C\left(m_1(\mathcal{X}_1),\ m_2(tails)\right) \\ &= m_2(tails) - \min(0.6,\ 0.7) \\ &= 0.1 \end{aligned}$$

This illustrates that different orderings lead to different masses and thus to different credal sets.

Remark: One reason why $\mathfrak{M}_{robust}$ is usually different from $\mathfrak{M}_{mass}$ is mainly because the ordering on focal sets can greatly differ from the ordering on reals. Consider for instance the minimum copula already presented in Example 8:

- In the precise setting, the minimum copula associates the highest probabilities to events with similar values (high-high or low-low), and the lowest probabilities to events with opposite values (low-high).
- In the imprecise setting, the concept of high or low values for focal sets does not usually exist. We thus replace it by an ordering $\preceq$ on focal sets, determining which set is considered "low" and which is "high" (regardless of the real values actually contained in the set). The minimum copula then associates the highest mass to joint focal sets with similar "values" in the sense of the ordering $\preceq$, and

> the lowest mass to sets with opposite values in the sense of the ordering $\preceq$. For instance, in the bivariate case, with marginal focal sets $a_1^1 \preceq_1 a_2^1$ and $a_1^2 \preceq_2 a_2^2$, using the minimum copula will assign high masses to $a_1^1 \times a_1^2$ and $a_2^1 \times a_2^2$ and low masses to $a_1^1 \times a_2^2$ and $a_2^1 \times a_1^2$.

Assigning a high mass to sets containing both low and high values at the same time is something that would not occur in the precise case, but is possible in the imprecise case. This explains a source of the difference between credal sets $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{robust}$.

We saw that $\mathfrak{M}_{robust}$ and $\mathfrak{M}_{mass}$ can be different credal sets. However, because $\mathfrak{M}_{robust}$ is difficult to compute, it would be interesting to still use $\mathfrak{M}_{mass}$ to approximate it, *i.e.* to verify that $\mathfrak{M}_{robust} \subseteq \mathfrak{M}_{mass}$ (outer approximation) or $\mathfrak{M}_{mass} \subseteq \mathfrak{M}_{robust}$ (inner approximation). In Example 11, we show that there is in general no reason for such a relation to exist. Furthermore, if we found an ordering allowing this relationship, then this ordering is copula dependent, as changing the copula might break the inclusion.

Example 11: Consider the following setting:

- We consider two spaces $\mathcal{X}_1 = \mathcal{X}_2 = \{1,\ 2,\ 3\}$
- We consider two (identical) mass distribution functions m_1, m_2, each respectively possessing two focal sets $\{2\}$ and $\{1, 3\}$.
- $m_1(\{2\}) = m_1(\{1,\ 3\}) = m_2(\{2\}) = m_2(\{1,\ 3\}) = 0.5$
- We want to join the credal sets induced by the mass functions using the minimum copula.

We will compute the bounds of $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{robust}$ to compare them. The marginals masses being identical and the copula being symmetrical, many results can be obtained by symmetry.

Let us first compute the bounds of $\mathfrak{M}_{robust}$. Marginal masses m_1 and m_2 imposes that each marginal probability $P_1 \in \mathfrak{M}_1$ will verify:

$$\begin{aligned} &\mathrm{Bel}_1(\{2\}) \leqslant P_1(\{2\}) \leqslant \mathrm{Pl}_1(\{2\}) \\ 0.5 = \sum_{a \subseteq \{2\}} m_1(a) \leqslant &P_1(\{2\}) \leqslant \sum_{a \cap \{2\} \neq \emptyset} m_1(a) = 0.5 \end{aligned}$$

The same result holds for $\{1,\ 3\}$. We therefore have:

$$
\begin{aligned}
&P_1(2) = 0.5\\
&P_1(1) + P_1(3) = 0.5\\
&0 \leqslant P_1(1) \leqslant 0.5\\
&0 \leqslant P_1(3) \leqslant 0.5
\end{aligned}
$$

And we can compute the same for m_2. Looking at those equations, we can deduce that every $P \in \mathfrak{M}_{robust}$ with marginals P_1, P_2 verifies for events $\{1\}\times\{1\}$ and $\{1,2\}\times\{1,2\}$:

$$
\begin{aligned}
& & P(\{1\},\ \{1\}) &= \min(P_1(1), P_2(1))\\
&\implies 0 = \min(0,0) \leqslant & P(\{1\},\ \{1\}) &\ \leqslant \min(0.5, 0.5) = 0.5\\
& & P(\{1,\ 2\},\ \{1,\ 2\}) &= \min(P_1(\{1,\ 2\}), P_2(\{1,\ 2\}))\\
\implies 0.5 = \min(0+0.5, 0.5+0) \leqslant & P(\{1,\ 2\},\ \{1,\ 2\}) &\ \leqslant min(0.5+0.5, 0.5+0.5) = 1
\end{aligned}
$$

Let us now compute the bounds of $\mathfrak{M}_{mass}$. Choosing an ordering between $\{1,\ 3\}$ and $\{2\}$ is not intuitive. Assume that there is a reason which encourages us to choose different orderings for the focal sets of m_1 and for those of m_2, so that $\{1,\ 3\} \preceq_1 \{2\}$ and $\{2\} \preceq_2 \{1,\ 3\}$. In this case, using Definition 19 it holds that:

$$
\begin{aligned}
m_C(\{1,\ 3\}, \{2\}) =&\ \min(0.5,\ 0.5)\\
=&\ 0.5\\
m_C(\{2\}, \{1,\ 3\}) =&\ \min(1,\ 1) - \min(0.5,\ 1)\\
&- \min(1,\ 0.5) + \min(0.5,\ 0.5)\\
=&\ 0.5\\
m_C(\{2\}, \{2\}) =&\ 0\\
m_C(\{1,\ 3\}, \{1,\ 3\}) =&\ 0
\end{aligned}
$$

Thus every probability $P \in \mathfrak{M}_{mass}$ will verify:

$$
\begin{aligned}
&\mathrm{Bel}_C(\{1\},\ \{1\}) \leqslant P(\{1\},\ \{1\}) \leqslant \mathrm{Pl}(\{1\},\ \{1\})\\
\Leftrightarrow &\sum_{a \subseteq (\{1\}\times\{1\})} m_C(a) \leqslant P(\{1\},\ \{1\}) \leqslant \sum_{a \cap (\{1\}\times\{1\}) \neq \emptyset} m_C(a)\\
&\Leftrightarrow 0 \leqslant P(\{1\},\ \{1\}) \leqslant 0
\end{aligned}
$$

and

$$\begin{aligned} &\mathrm{Bel}_C(\{1,\ 2\},\ \{1,\ 2\}) \leqslant P(\{1,\ 2\},\ \{1,\ 2\}) \leqslant \mathrm{Pl}(\{1,\ 2\},\ \{1,\ 2\}) \\ \Leftrightarrow &\sum_{a \subseteq (\{1,\ 2\},\ \{1,\ 2\})} m_C(a) \leqslant P(\{1,\ 2\},\ \{1,\ 2\}) \leqslant \sum_{a \cap (\{1,\ 2\},\ \{1,\ 2\}) \neq \emptyset} m_C(a) \\ &\Leftrightarrow 0 \leqslant P(\{1,\ 2\},\ \{1,\ 2\}) \leqslant 1 \end{aligned}$$

Looking at the bounds of $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{robust}$ on cumulative event $\{1\} \times \{1\}$, we can see that $\overline{P}_{robust}(\{1\}, \{1\}) > \mathrm{Pl}(\{1\}, \{1\}))$ and thus $\mathfrak{M}_{robust} \not\subseteq \mathfrak{M}_{mass}$. Looking at the bounds on $\{1, 2\} \times \{1, 2\}$, we can see that $\underline{P}_{robust}(\{1,\ 2\},\ \{1,\ 2\}) > \mathrm{Bel}_C(\{1,\ 2\},\ \{1,\ 2\})$ and thus $\mathfrak{M}_{mass} \not\subseteq \mathfrak{M}_{robust}$.

> **Remark:** The bounds of $\mathfrak{M}_{mass}$ depend both on the ordering and the copula used. Therefore, if an ordering exists such that $\mathfrak{M}_{robust} \subseteq \mathfrak{M}_{mass}$ or $\mathfrak{M}_{mass} \subseteq \mathfrak{M}_{robust}$, then this ordering is not guaranteed to keep the inclusion relationship using a different copula.

We considered until now that an ordering has to be chosen arbitrarily. However, special cases of belief functions exhibit a natural ordering on their focal sets, for instance p-boxes and possibilities. Those special cases will be explored in Section 3.2.

3.1.3 Copulas Applied to Belief Functions

Another way of joining credal sets with a copula is by directly applying the copula to their lower envelope $\underline{P}_i$ for every event:

Definition 21: Aggregated Credal Set

Given a copula C and n marginal credal sets whose lower probabilities are $\underline{P}_1,\ \ldots,\ \underline{P}_n$, we define the credal set $\mathfrak{M}_{agg}$ over the power set of $\mathcal{X} = \mathcal{X}_1 \times \cdots \times X_n$ as:

$$\mathfrak{M}_{agg} = CH(\{P \mid \forall A_i \subseteq \mathcal{X}_i, P(A_1,\ \ldots,\ A_n) \geqslant C(\underline{P}_1(A_1),\ \ldots,\ \underline{P}_n(A_n))\}) \tag{3.8}$$

where CH is the convex hull from Definition 7.

Contrary to $\mathfrak{M}_{mass}$ or $\mathfrak{M}_{robust}$, constraints on this set only occur on Cartesian products in $\mathcal{X}$. We thus take the convex hull for extending its definition to every event in $2^{\mathcal{X}}$. We denote this credal set as $\mathfrak{M}_{agg}$ because it uses the copula solely as an aggregation operator, without conserving the meaning associated with copulas by Sklar's Theorem.

In this regard, $\mathfrak{M}_{agg}$ has less meaning than $\mathfrak{M}_{robust}$ or $\mathfrak{M}_{mass}$, but presents the advantage of being easier to compute on cylindrical sets. Figure 3.4 sums up the performances of the different methods in terms of computation cost and meaningfulness.

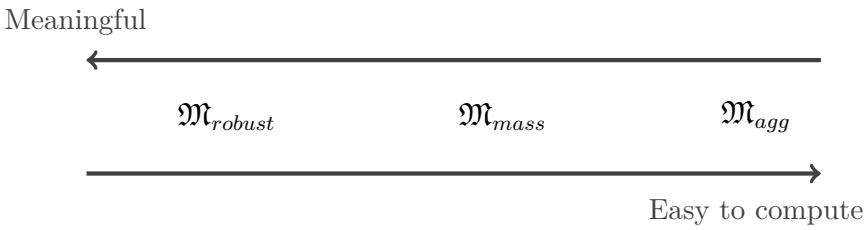


Figure 3.4: Comparing different methods of joining credal sets with a copula.

In general, applying the copula directly to the lower probabilities as in (3.8) does not produce a coherent lower probability inducing a non-empty credal set (see Definition 6). For instance, let us consider $\mathcal{X}_1 = \mathcal{X}_2 = \{1, 2\}$, two lower previsions $\underline{P}_1$ and $\underline{P}_2$ such that $\underline{P}_1(\{1\}) = \underline{P}_1(\{2\}) = \underline{P}_2(\{1\}) = \underline{P}_2(\{2\}) = 0.5$. Joining those two lower probabilities using the minimum copula $C(u, v) = \min(u, v)$ gives a mapping $\underline{P}$ which induces an empty credal set, as presented in Table 3.1. Indeed, no probabilities can satisfy all these constraints at once.

$\underline{P}$	$\{1\}$	$\{2\}$
$\{1\}$	0.5	0.5
$\{2\}$	0.5	0.5

Table 3.1: $\underline{P} = \min(\underline{P}_1, \underline{P}_2)$

Proposition 2: In the special case of the product copula C_Π, the credal set $\mathfrak{M}_{agg}$ induced by (3.8) is not empty. It follows that for all copulas C dominated by the product copula (*i.e.* $C_\Pi \geqslant C$), and for every non-empty marginal credal set $\mathfrak{M}_i$, $\mathfrak{M}_{agg}(C, \mathfrak{M}_i)$ is also non-empty credal set.

Proof: For $i \in \{1, \ldots, n\}$, let $\underline{P}_i$ be a lower probability avoiding sure loss, *i.e.* whose credal set $\mathfrak{M}_i$ contains at least one probability distribution P_i. Let us define a multivariate probability P on every $(A_1 \times \cdots \times A_n) \subseteq \mathcal{X}$ as:

$$P(A_1 \times \cdots \times A_n) = P_1(A_1) \times \cdots \times P_n(A_n)$$

Defining P on $(A_1 \times \cdots \times A_n) \subseteq \mathcal{X}$ is sufficient as those sets contain every atom of

$\mathcal{X}$. Because $\forall i, P_i \in \mathfrak{M}_i,\ P_i \geqslant \underline{P}_i$, then:

$$\begin{aligned} P(A_1 \times \cdots \times A_n) \geqslant \underline{P}_1(A_1) \times \cdots \times \underline{P}_n(A_n) &= C_\Pi(\underline{P}_1(A_1),\ \ldots,\ \underline{P}_n(A_n)) \\ &= \underline{P}_{C_\Pi}(A_1 \times \cdots \times A_n) \end{aligned}$$

which means $P \in \mathfrak{M}_{agg}(C_\Pi, \mathfrak{M}_i)$. Therefore, $\mathfrak{M}_{agg}(C_\Pi, \mathfrak{M}_i) \neq \emptyset$ if every $\mathfrak{M}_i \neq \emptyset$.

Let C be a copula dominated by C_Π (*i.e.* $C_\Pi \geqslant C$), and $\underline{P}_C$ the lower probability associated with $\mathfrak{M}_{agg}(C, \mathfrak{M}_i)$. Then it holds that for all $(A_1 \times \cdots \times A_n) \subseteq \mathcal{X}$:

$$\begin{aligned} \underline{P}_{C_\Pi}(A_1 \times \cdots \times A_n) &= C_\Pi(\underline{P}_1(A_1),\ \ldots,\ \underline{P}_n(A_n)) \\ &\geqslant C(\underline{P}_1(A_1),\ \ldots,\ \underline{P}_n(A_n)) = \underline{P}_C(A_1 \times \cdots \times A_n) \end{aligned}$$

which implies that $\mathfrak{M}_{agg}(C_\Pi, \mathfrak{M}_i) \subseteq \mathfrak{M}_{agg}(C, \mathfrak{M}_i)$. Therefore, if every $\mathfrak{M}_i$ is a non-empty credal set, then $\mathfrak{M}_{agg}(C, \mathfrak{M}_i)$ is also non-empty. □

Proposition 3: Conversely, no lower probability $\underline{P}_C$ obtained using (3.8) with a copula C strictly superior to the product copula is guaranteed to induce a non-empty credal set $\mathfrak{M}_{agg}$. It all depends on the marginal credal sets $\underline{P}_i$.

Proof: Let C be a copula strictly superior to the product. Then there exists $(u_1,\ \ldots,\ u_n) \in [0,1]^2$ such that:

$$C(u_1,\ \ldots,\ u_n)\ >\ \prod_{i=1}^{n} u_i$$

Let $\mathfrak{M}_i$ be marginals credal sets such that $\underline{P}_i$ are *precise* probabilities, and that:

$$\forall i,\ \exists A_i \in \mathcal{X}_i,\ \underline{P}_i(A_i) = u_i$$

We will prove the proposition by contradiction. Assume that $\mathfrak{M}_{agg}(C, \mathfrak{M}_i)$ avoids sure loss, *i.e.* there is a probability P such that $P \geqslant \underline{P}_C$. Let S be a collection of disjoint cylindrical sets of $\mathcal{X}$ (defined in Equation (3.1)) covering the complementary event $(A_1 \times \cdots \times A_n)^c$ of $(A_1 \times \cdots \times A_n)$. S is defined so that $(A_1 \times \cdots \times A_n)^c = \bigsqcup_{s \in S} s$.

Then,

$$\begin{aligned}
P(\mathcal{X}) &= P\left((A_1 \times \cdots \times A_n) \bigsqcup (A_1 \times \cdots \times A_n)^c\right) \\
&= P\left(A_1 \times \cdots \times A_n\right) + P\left((A_1 \times \cdots \times A_n)^c\right) \\
&= P\left(A_1 \times \cdots \times A_n\right) + \sum_{(s_1 \times \cdots \times s_n) \in S} P(s_1 \times \cdots \times s_n) \\
&\geqslant \underline{P}_C\left(A_1 \times \cdots \times A_n\right) + \sum_{(s_1 \times \cdots \times s_n) \in S} \underline{P}_C(s_1 \times \cdots \times s_n) \\
&> \underline{P}_{C_\Pi}\left(A_1 \times \cdots \times A_n\right) + \sum_{(s_1 \times \cdots \times s_n) \in S} \underline{P}_{C_\Pi}(s_1 \times \cdots \times s_n)
\end{aligned}$$

Because we chose $\underline{P}_i$ so that they are precise probabilities, their product is also a precise probability. Using the fact that summing probabilities of disjoint events is equal to the probability of their union:

$$\begin{aligned}
\underline{P}_{C_\Pi}\left(A_1 \times \cdots \times A_n\right) + \sum_{(s_1 \times \cdots \times s_n) \in S} \underline{P}_{C_\Pi}(s_1 \times \cdots \times s_n) &= \underline{P}_{C_\Pi}\left(A_1 \times \cdots \times A_n\right) + \\
&\quad \underline{P}_{C_\Pi}\left((A_1 \times \cdots \times A_n)^c\right) \\
&= 1
\end{aligned}$$

This means that $P(\mathcal{X}) > 1$ which is impossible. Thus, $\mathfrak{M}_{agg}(C, \mathfrak{M}_i) = \emptyset$ and $\mathfrak{M}_{agg}(C, \mathfrak{M}_i)$ does not avoid sure loss. □

We have now presented three methods for joining imprecise models using a copula. The following sections will explore special cases where some interesting relationships between those copulas exist.

3.2 Inclusions Between Joint Credal Sets

Section 3.1 presented three methods for joining marginal credal sets using a copula. In general, there is no reason for the three methods to lead to the same multivariate credal sets. However, for some specific cases on the copulas or on the marginal credal sets, it is possible to find inclusion relationships between the methods. This section explores some of these specific cases. Because each method has a different computational complexity, knowing those relationships allows us to use a simpler method to approximate another. For instance, if we know that $\mathfrak{M}_{robust} \subseteq \mathfrak{M}_{mass}$, then we can determine conservative bounds on $\mathfrak{M}_{robust}$ by computing those of $\mathfrak{M}_{mass}$, which are simpler to determine. This will specifically be used in Chapter 4.

3.2.1 Using the Product Copula

In this section, we will consider the case of the product copula C_Π, representing independence between variables. Using this copula in the robust approach defined by Definition 18 is referred to as the strong product in [de Cooman et al., 2010]. Let us denote $\underline{P}_{robust}$ the infimum of $\mathfrak{M}_{robust}(C_\Pi, \mathfrak{M}_i)$ and $\mathcal{S}$ the set from which $\mathfrak{M}_{robust}$ is computed (Equation (3.2)). For cylindrical sets $(A_1, \ \ldots, \ A_n)$ of $\mathcal{X}$, it holds that:

$$\begin{aligned}
\underline{P}_{robust}(A_1 \times \cdots \times A_n) &= \inf\{P(A_1 \times \cdots \times A_n) \mid P \in \mathcal{S}\} \\
&= \inf\{P_1(A_1) \ldots P_n(A_n) \mid P_i \in \mathfrak{M}_i\} \\
&= \inf\{P_1(A_1) \mid P_1 \in \mathfrak{M}_1\} \ldots \inf\{P_n(A_n) \mid P_n \in \mathfrak{M}_n\} \\
&= \underline{P}_1(A_1) \ldots \underline{P}_n(A_n)
\end{aligned}$$

We can split the infimum of a product as a product of infima because we consider mappings with positive values. As this is equivalent to applying the copula directly to the marginals, $\mathfrak{M}_{robust}$ and $\mathfrak{M}_{agg}$ have the same bounds on cylindrical events. On other events, the lower probabilities are defined as the infimum of the credal sets, thus all bounds are the same. $\mathfrak{M}_{robust}$ is defined as the convex hull of the set of probabilities whose marginals are in $\mathfrak{M}_i$, which is a constraint that $\mathfrak{M}_{agg}$ do not have. Therefore, the sets are not necessarily the same, although they share the same bounds. The way $\mathfrak{M}_{agg}$, is defined, it is the largest set with those bounds. On the other hand, as $\mathfrak{M}_{robust}$ is the convex hull of the set S from Equation (3.2) now defined specifically as:

$$\begin{aligned}
S = \{ \ F \mid \forall (x_1, \ \ldots, \ x_n) \in \mathcal{X}, \ \forall F_i \in \mathfrak{M}_i, \\
F(x_1, \ \ldots, \ x_n) = F_1(x_1) \cdot \ldots \cdot F_n(x_n) \ \}
\end{aligned}$$

This means that $\mathfrak{M}_{robust}$ is the convex hull of a set of specific probabilities verifying those bounds; it is therefore a smaller set than $\mathfrak{M}_{agg}$. It results that:

$$\mathfrak{M}_{robust}(C_\Pi, \mathfrak{M}_i) \subseteq \mathfrak{M}_{agg}(C_\Pi, \mathfrak{M}_i) \tag{3.9}$$

This result can also be found in [Couso et al., 2000].

Let us now consider a property on the mass m_C from Definition 19, which will later allow us to prove an inclusion with Bel_C.

Proposition 4: In the case of the product copula C_Π, the arbitrary orderings on marginal focal sets have no impact on the value of the joint mass m_C defined in (3.4).

Indeed, if $a^1_{k_1},\ \ldots,\ a^n_{k_n}$ is a focal set of $m_1,\ \ldots,\ m_n$, then m_C is given by:

$$m_C(a^1_{k_1} \times \cdots \times a^n_{k_n}) = m_1(a^1_{k_1}) \ldots m_n(a^n_{k_n}) \tag{3.10}$$

Proof: For simplicity and coherence with the notations of Section 2.4.1, we will note for all $i \in [0, n]$, $u_{k_i} = \sum_{l=0}^{k_i-1} m_i(a^i_l)$, $v_{k_i} = \sum_{l=0}^{k_i-1} m_i(a^i_l)$. $\prod_{i=1}^n \{u_{k_i}, v_{k_i}\}$ will refer to the Cartesian product $\{u_{k_1}, v_{k_1}\} \times \{u_{k_2}, v_{k_2}\} \times \cdots \times \{u_{k_n}, v_{k_n}\}$ and we will note C_Π and H as the product copula and its H-volume regardless of their number of marginals. Those notations established, it holds that:

$$\begin{aligned} m_C(a^1_{k_1} \times \cdots \times a^n_{k_n}) &= H^{v_{k_1},\ \ldots,\ v_{k_n}}_{u_{k_1},\ \ldots,\ u_{k_n}} \\ &= \sum_{\substack{(w_{k_1},\ \ldots,\ w_{k_n}) \in \\ \prod_{i=1}^n \{u_{k_i}, v_{k_i}\}}} (-1)^{|\{k \ |\ w_{k_i} = u_{k_i}\}|} C_\Pi(w_{k_1},\ \ldots,\ w_{k_n}) \\ &= \sum_{\substack{(w_{k_1},\ \ldots,\ w_{k_n}) \in \\ \prod_{i=1}^n \{u_{k_i}, v_{k_i}\}}} (-1)^{|\{k \ |\ w_{k_i} = u_{k_i}\}|} (w_{k_1} \cdot \ldots \cdot w_{k_n}) \end{aligned}$$

and by explicitly writing the terms for $w_{k_n} = v_{k_n}$ and $w_{k_n} = u_{k_n}$:

$$\begin{aligned} m_C(a^1_{k_1} \times \cdots \times a^n_{k_n}) &= \sum_{\substack{(w_{k_1},\ \ldots,\ w_{k_{n-1}}) \in \\ \prod_{i=1}^{n-1} \{u_{k_i}, v_{k_i}\}}} (-1)^{|\{k \ |\ w_{k_i} = u_{k_i}\}|} (w_{k_1} \cdot \ldots \cdot w_{k_{n-1}} \cdot v_{k_n}) \\ &\quad + \sum_{\substack{(w_{k_1},\ \ldots,\ w_{k_{n-1}}) \in \\ \prod_{i=1}^{n-1} \{u_{k_i}, v_{k_i}\}}} (-1)^{|\{k \ |\ w_{k_i} = u_{k_i}\}|+1} (w_{k_1} \cdot \ldots \cdot w_{k_{n-1}} \cdot u_{k_n}) \\ &= v_{k_n} H^{v_{k_1},\ \ldots,\ v_{k_{n-1}}}_{u_{k_1},\ \ldots,\ u_{k_{n-1}}} - u_{k_n} H^{v_{k_1},\ \ldots,\ v_{k_{n-1}}}_{u_{k_1},\ \ldots,\ u_{k_{n-1}}} \\ &= m_n(a^n_{k_n}) H^{v_{k_1},\ \ldots,\ v_{k_{n-1}}}_{u_{k_1},\ \ldots,\ u_{k_{n-1}}} \end{aligned}$$

Doing the same procedure for every variable leads to:

$$m_C(a^1_{k_1} \times \cdots \times a^n_{k_n}) = m_1(a^1_{k_1}) \ldots m_n(a^n_{k_n})$$

which concludes the proof. □

The mass m_C corresponds to the notion of random set independence presented in [Dempster, 1967, Couso et al., 2000]. Let Bel_C be the belief function associated with m_C, and $\forall i \in [1, n], \mathrm{Bel}_i$ the mass function associated with m_i. Then for cylindrical sets

$(A_1,\ \ldots,\ A_n)$ of $\mathcal{X}$, it holds that:

$$\begin{aligned}
\mathrm{Bel}_C(A_1 \times \cdots \times A_n) &= \sum_{(a^1 \times \cdots \times a^n) \subseteq (A_1,\ \ldots,\ A_n)} m_C(a^1 \times \cdots \times a^n) \\
&= \sum_{(a^1 \times \cdots \times a^n) \subseteq (A_1,\ \ldots,\ A_n)} m_1(a_1) \cdot \ldots \cdot m_n(a^n) \\
&= (\sum_{a^1 \subseteq A_1} m_1(a^1)) \cdot \ldots \cdot (\sum_{a^n \subseteq A_n} m_n(a^n)) \\
&= \mathrm{Bel}_1(A_1) \ldots \mathrm{Bel}_n(A_n) \qquad (3.11)
\end{aligned}$$

This means that in the case of the product copula C_Π with marginals being belief functions, $\mathfrak{M}_{robust}$, $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$ all coincide on cylindrical sets. Because $\mathfrak{M}_{agg}$ has no specific constraints on other sets, it is the largest credal set with these bounds. Because m_{mass} is defined on other bounds, $\mathfrak{M}_{mass}$ also has constraints on other bounds. It is therefore a smaller set than $\mathfrak{M}_{agg}$ and:

$$\mathfrak{M}_{mass} \subseteq \mathfrak{M}_{agg}$$

Finally, it is straightforward to verify that $\mathfrak{M}_{mass}$ contains every probability from the set S

$$\begin{aligned}
S = \{\ F \mid \forall (x_1,\ \ldots,\ x_n) \in \mathcal{X},\ \forall F_i \in \mathfrak{M}_i, \\
F(x_1,\ \ldots,\ x_n) = F_1(x_1) \cdot \ldots \cdot F_n(x_n)\ \}
\end{aligned}$$

of which $\mathfrak{M}_{robust}$ is the convex hull. As $\mathfrak{M}_{mass}$ is defined by a belief function, it is therefore also convex, therefore:

$$\mathfrak{M}_{robust} \subseteq \mathfrak{M}_{mass}$$

In the case of the product copula, the following inclusion ordering holds:

$$\mathfrak{M}_{robust} \subseteq \mathfrak{M}_{mass} \subseteq \mathfrak{M}_{agg} \qquad (3.12)$$

Regardless of the marginal belief functions used. This means that computing the bounds of $\mathfrak{M}_{agg}$, which is straightforward, allow us to obtain a set containing $\mathfrak{M}_{robust}$. We also have seen in this section that on Cartesian products, the multivariate belief function could simply be evaluated without computing its joint mass. The next sections will investigate the relationship between $\mathfrak{M}_{robust}$, $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$ for other copulas, but with specific types of marginal imprecise models.

3.2.2 Using the Natural Ordering of Necessity Functions

We will now investigate the specific case where we use any copula C to join multiple marginal necessity functions. This setting will be considered in Chapter 4. We saw in Section 2.3.4 that focal sets $(a_1, \ldots, a_n)$ of necessity functions are included into one another as follows:

$$a_1 \subset a_2 \subset \ldots \subset a_n$$

Here, we used a specific ordering for focal sets (the natural order) but any other ordering could have been used. A family of events verifying this inclusion property is called an *increasing* family of events in the following.

In [Schmelzer, 2015a], the author showed that in order to describe the relation between a multivariate belief function and its marginals in the bivariate case, it is necessary to consider a family of sub-copulas: one copula for each tuple of increasing family of events. We remind that a sub-copula is a restriction of a copula to a subset of the unit hyper-cube $[0, 1]$ as presented in Section 2.4.1.

Theorem 2: Sklar's Theorem for Belief Functions [Schmelzer, 2015a]

Let Bel : $2^{\mathcal{X}_1} \times 2^{\mathcal{X}_2} \to [0, 1]$ be a bivariate belief function and let Bel_1 and Bel_2 denote its marginals over $2^{\mathcal{X}_1}$ and $2^{\mathcal{X}_2}$ respectively. Furthermore, let $\mathcal{I}_1$ and $\mathcal{I}_2$ denote increasing families of subsets of $\mathcal{X}_1$ and $\mathcal{X}_2$. Then there exists a unique sub-copula $C^{\mathcal{I}_1,\mathcal{I}_2}$ on $\text{Bel}_1(\mathcal{I}_1) \times \text{Bel}_2(\mathcal{I}_2)$ such that:

$$\text{Bel}(L_1, L_2) = C^{\mathcal{I}_1,\mathcal{I}_2}(\text{Bel}_1(L_1), \text{Bel}_2(L_2)) \tag{3.13}$$

for all $L_1 \in \mathcal{I}_1, L_2 \in \mathcal{I}_2$.

For the reverse to be true, it is necessary that $\mathcal{X}_1 \in \mathcal{I}_1, \mathcal{X}_2 \in \mathcal{I}_2$. Example 1 of [Schmelzer, 2015a] illustrate the need of a copula for each increasing family of events.

> **Remark:** In [Lesniewska-Choquet, 2020], it has been proposed to directly apply the copula to the marginal possibility distributions π_i, *i.e.* $\pi(x, y) = C(\pi_1(x), \pi_2(y))$. It is however shown that this method does not work in general with copulas; however, the author presents more specific aggregation models (called *t-conorms*) as a solution. This work is very interesting, and is also used on satellite images (although for a different application as ours, *i.e.* for detecting land changes). Although it may seem very similar to our subject, we sadly cannot compare our results to it as the consid-

ered settings differ too much. The objective of their thesis can be briefly summed as follows: given a multivariate probability P, how to obtain a multivariate possibility distribution π consistent with P, *i.e.* such that $P \in \mathfrak{M}(\pi)$. They also restrict the marginals of P to single probability distributions (and not credal sets), which must be either Gaussian, Cauchy or Student probability distributions. They show that determining such a multivariate possibility is possible for the upper and lower Fréchet–Hoeffding bounds, but can only determine bounds on the possibility for the product copula, for instance. Other copulas are not considered. Linking our work to theirs seemed a big stretch and has thus not be considered.

Necessity functions are completely determined by their focal sets, which form an increasing family of events. Thus, by applying Sklar's Theorem for Belief Functions, it holds that joining two necessity functions with a copula C as in (3.8) yields a bivariate belief function (which is not necessarily a necessity function):

$$\text{Bel} = C(\text{Nec}_1, \text{Nec}_2) \tag{3.14}$$

where Nec_1 and Nec_2 are the marginal necessity functions. The proof of those results were shown in [Schmelzer, 2015a, Schmelzer, 2015b]. This way of applying the copula directly on necessities is the same approach as in $\mathfrak{M}_{agg}$. In the following, we will consider that the focal sets a^i of a necessity functions Nec_i are already ranked using the natural ordering $\preceq_i$, which is convenient when manipulating those representations:

$$\forall (k,j) \in [1, N_i]^2,\ k \leqslant j \ \Leftrightarrow\ a^i_k \preceq_i a^i_j \ \Leftrightarrow\ a^i_k \subseteq a^i_j \tag{3.15}$$

The method for joining necessity functions from Sklar's Theorem is similar to the one for creating a multivariate belief function as in (3.6), as presented in the following proposition.

Proposition 5: Joining two marginal necessity functions $\text{Nec}_1, \text{Nec}_2$ with a copula C as in (3.14) or using the bivariate mass function as in (3.4) with the natural inclusion ordering yields the same bivariate belief function.

Proof: If we denote by Bel_C the belief function defined in (3.4) where the ordering is the inclusion ordering $\preceq_i$ for $i \in [1, 2]$. For convenience and with respect to the notations of Section 2.4.1, we note: $u^i_k = \sum_{j=0}^{k} m_i(a^i_j)$ and consider that $a^i_0 = \emptyset$. For

all focal elements a_k^1 of Nec_1 and a_j^2 of Nec_2, it holds that:

$$\begin{aligned}
\text{Bel}_C(a_k^1, a_j^2) &= \sum_{a_p^1 \subseteq a_k^1} \sum_{a_q^2 \subseteq a_j^2} m_C(a_p^1, a_q^2) = \sum_{p=1}^{k} \sum_{q=1}^{j} m_C(a_p^1, a_q^2) \\
&= \sum_{p=1}^{k} \sum_{q=1}^{j} (\ C(u_p^1, u_q^2) + C(u_{p-1}^1, u_{q-1}^2) \\
&\quad -C(u_{p-1}^1, u_q^2) - C(u_p^1, u_{q-1}^2)\ ) \\
&= \sum_{p=1}^{k} \sum_{q=1}^{j} C(u_p^1, u_q^2) + \sum_{p=0}^{k-1} \sum_{q=0}^{j-1} C(u_p^1, u_q^2) \\
&\quad - \sum_{p=0}^{k-1} \sum_{q=1}^{j} C(u_p^1, u_q^2) - \sum_{p=1}^{k} \sum_{q=0}^{j-1} C(u_p^1, u_q^2) \\
&= C(u_k^1, u_j^2) = C\left(\text{Nec}_1(a_k^1), \text{Nec}_2(a_j^2)\right)
\end{aligned}$$

This proof only works in the special case of necessity functions because:

$$\sum_{a_p^1 \subseteq a_k^1} \sum_{a_q^2 \subseteq a_j^2} m_C(a_p^1, a_q^2) = \sum_{p=1}^{k} \sum_{q=1}^{j} m_C(a_p^1, a_q^2)$$

is only true for marginal necessity functions. □

Proposition 5 considers two marginals. However, we will see in the next proposition that it still holds for n marginals, not covered in [Schmelzer, 2015b].

Proposition 6: Joining n marginal necessity functions $\text{Nec}_1, \ \ldots, \ \text{Nec}_n$ with a n-copula C as in (3.14) or using the multivariate variate mass function as in (3.4) with the natural inclusion ordering yields the same multivariate belief function. In other words, for every cylindrical set $(A_1, \ \ldots, \ A_n) \subseteq \mathcal{X}$, it holds that:

$$\text{Bel}_C(A_1 \times \cdots \times A_n) = C\left(\text{Nec}_1(A_1), \ \ldots, \ \text{Nec}_n(A_n)\right) \tag{3.16}$$

In other words, $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$ have the same bounds on cylindrical sets when marginals are necessity functions.

Proof: The proof is similar to the one of Definition 19, but this time computing the

mass of $(a^1_{k_1} \times \cdots \times a^n_{k_n})$ using $F([0,k_1] \times \cdots \times [0,k_n])$ and noticing that

$$\sum_{(a^1_{p_1} \times \cdots \times a^n_{p_n}) \subseteq (a^1_{k_1} \times \cdots \times a^n_{k_n})} m_C(a^1_{p_1} \times \cdots \times a^n_{p_n}) = \sum_{p_1=1}^{k_1} \ldots \sum_{p_n=1}^{k_n} m_C(a^1_{p_1} \times \cdots \times a^n_{p_n})$$

because all marginals are necessity functions, and the natural inclusion ordered is used for ranking their focal sets. □

As $\mathfrak{M}_{mass}$ is defined by a mass distribution function on $2^{\mathcal{X}}$, it also possesses constraints on events that are not cylindrical sets. On the other hand, $\mathfrak{M}_{agg}$ is the largest credal set with bounds specified by (3.16) on cylindrical sets. For marginal sets $\mathfrak{M}_i$ defined by necessity functions, it therefore holds that:

$$\mathfrak{M}_{mass}(C, \mathfrak{M}_i) \subseteq \mathfrak{M}_{agg}(C, \mathfrak{M}_i) \tag{3.17}$$

regardless of the copula C used.

When considering Bel_C whose marginals necessity functions equipped with the natural ordering for their focal sets, it is straightforward that m_C verifies:

$$\sum_{a^1_p \subseteq a^1_k} \sum_{a^2_q \subseteq a^2_j} m_C(a^1_p, a^2_q) = \sum_{p=1}^{k} \sum_{q=1}^{j} m_C(a^1_p, a^2_q)$$

We can wonder if this equality is only verified by marginals that are necessity functions or not. The following proposition shows that this equality is a sufficient and necessary condition to characterize multivariate belief functions whose marginals are necessities.

Proposition 7: Let m_C be a joint mass obtained using (3.4). m_C verifies

$$\sum_{a^1_p \subseteq a^1_k} \sum_{a^2_q \subseteq a^2_j} m_C(a^1_p, a^2_q) = \sum_{p=1}^{k} \sum_{q=1}^{j} m_C(a^1_p, a^2_q)$$

for all marginal focal sets (a^1_k) ,(a^2_j) if and only if its marginals masses correspond to necessity functions, equipped with the natural ordering.

Proof:

$\Longleftarrow$ By using the natural inclusion ordering on marginal focal set, it is immediate

that

$$\sum_{a_p^1 \subseteq a_k^1} \sum_{a_q^2 \subseteq a_j^2} m_C(a_p^1, a_q^2) = \sum_{p=1}^{k} \sum_{q=1}^{j} m_C(a_p^1, a_q^2)$$

$\implies$ Let m_C be a joint mass obtained using (3.4), with marginal focal sets $(a_k^1)_{1 \leqslant k \leqslant N_1}$, $(a_j^2)_{1 \leqslant j \leqslant N_2}$ and marginal masses m_1, m_2. Let Bel_C be its associated belief function verifying:

$$Bel_C(a_k^1, a_j^2) = \sum_{a_p^1 \subseteq a_k^1} \sum_{a_q^2 \subseteq a_j^2} m_C(a_p^1, a_q^2) = \sum_{p=1}^{k} \sum_{q=1}^{j} m_C(a_p^1, a_q^2)$$

for all marginal focal sets a_k^1, a_j^2. By summing the H-volume over a complete partition of $[0, 1]$, it is easy to check that:

$$m_1(a_p^1) = \sum_{a_q^2 \subseteq \mathcal{X}_2} m_C(a_p^1, a_q^2) = \sum_{q=1}^{N_2} m_C(a_p^1, a_q^2)$$

Thus it holds that:

$$\sum_{p=1}^{k} m_1(a_p^1) = \text{Bel}_C(a_k^1, \mathcal{X}_2) = \sum_{a_p^1 \subseteq a_k^1} m_1(a_p^1)$$

This result is not sufficient to prove the inclusion of focal sets (there could be a set $a_p^1 \subseteq a_k^1,\ p > k$ with the same mass value than another set $a_{p'}^1 \not\subseteq a_k^1,\ p' < k$). Let us show by induction that for all $(k, p) \in [\![1, N_1]\!]^2$, $a_1^1 \subset \ldots \subset a_k^1$ and $a_p^1 \not\subseteq a_k^1$ if $p > k$. For the case $k = 1$, it holds that:

$$\begin{aligned}
&\sum_{p=1}^{1} m_1(a_p^1) = \sum_{a_p^1 \subseteq a_1} m_1(a_p^1) \\
&\Leftrightarrow m_1(a_1^1) = m_1(a_1^1) + \sum_{a_p^1 \subset a_1^1} m_1(a_p^1) \\
&\qquad \Leftrightarrow 0 = \sum_{a_p \subset a_1} m_1(a_p^1)
\end{aligned}$$

which means that no focal set is a strict subset of $a_1^1 = a_k^1$, so if $p > k$, $a_p^1 \not\subseteq a_k^1$.

For the induction step, suppose that $k \in [\![1, N_1]\!]$, $a_1 \subset \ldots \subset a_k$ and $\forall p > k,\ a_p \not\subseteq$

a_k. In particular, $a_{k+1} \not\subseteq a_k$. It holds that:

$$\begin{aligned}
\sum_{p=1}^{k+1} m_1(a_p^1) &= \sum_{a_p^1 \subseteq a_{k+1}^1} m_1(a_p^1) \\
\Leftrightarrow m_1(a_{k+1}^1) + \sum_{p=1}^{k} m_1(a_p^1) &= \sum_{a_p^1 \subseteq a_{k+1}^1} m_1(a_p^1) \\
\Leftrightarrow m_1(a_{k+1}^1) + \sum_{a_p^1 \subseteq a_k^1} m_1(a_p^1) &= \sum_{a_p^1 \subseteq a_{k+1}^1} m_1(a_p^1) \\
\Longrightarrow m_1(a_{k+1}^1) &= \sum_{\substack{a_p^1 \subseteq a_{k+1}^1 \\ a_p^1 \not\subseteq a_k^1}} m_1(a_p^1) \\
\Longrightarrow m_1(a_{k+1}^1) &= m_1(a_{k+1}^1) + \sum_{\substack{a_p^1 \subset a_{k+1}^1 \\ a_p^1 \not\subseteq a_k^1}} m_1(a_p^1) \\
\Leftrightarrow 0 &= \sum_{\substack{a_p^1 \subset a_{k+1}^1 \\ a_p^1 \not\subseteq a_k^1}} m_1(a_p^1)
\end{aligned}$$

Which means that either there is no focal set that is a strict subset of a_{k+1}^1, or that they are all included in a_k^1. The first case is discarded as:

$$\begin{aligned}
\sum_{a_p^1 \subseteq a_{k+1}^1} m_1(a_p^1) &= \sum_{p=1}^{k+1} m_1(a_p^1) \\
\Leftrightarrow \sum_{a_p^1 \subset a_{k+1}^1} m_1(a_p^1) &= \sum_{p=1}^{k} m_1(a_p^1) > 0
\end{aligned}$$

thus $a_1 \subset \ldots \subset a_{k+1}$. Finally, it also follows that:

$$\begin{aligned}
\sum_{a_p^1 \subseteq a_{k+1}^1} m_1(a_p^1) &= \sum_{p=1}^{k+1} m_1(a_p^1) \\
\Longrightarrow \sum_{\substack{a_p^1 \subseteq a_{k+1}^1 \\ p > k+1}} m_1(a_p^1) + \sum_{\substack{a_p^1 \subseteq a_{k+1}^1 \\ p \leqslant k+1}} m_1(a_p^1) &= \sum_{p=1}^{k+1} m_1(a_p^1) \\
\Leftrightarrow \sum_{\substack{a_p^1 \subseteq a_{k+1}^1 \\ p > k+1}} m_1(a_p^1) &= 0
\end{aligned}$$

meaning that for all $p > k+1$, $a_p^1 \not\subseteq a_{k+1}^1$, ending the proof by induction. Because all focal sets form a nested family of sets, Bel_1 is a necessity function. The proof for Bel_2 is identical. □

Without further assumptions, there is no inclusion relations between $\mathfrak{M}_{robust}$ and $\mathfrak{M}_{agg}$ or $\mathfrak{M}_{mass}$. The following examples present cases where $\inf \mathfrak{M}_{robust} < \inf \mathfrak{M}_{agg}$ or $\inf \mathfrak{M}_{mass} < \inf \mathfrak{M}_{robust}$, proving that it is not always possible to get an (inner $\subseteq$ or outer $\supseteq$) approximation of $\mathfrak{M}_{robust}$ using $\mathfrak{M}_{mass}$ or $\mathfrak{M}_{agg}$.

Example 12: Let $n = 2$. Consider $\mathcal{X}_1 = \{x_1^1, x_2^1\}$ and $\mathcal{X}_2 = \{x_1^2, x_2^2\}$. Let us define two possibility distribution π_1 and π_2 over $\mathcal{X}_1$ and $\mathcal{X}_2$ respectively, such that:

$$\begin{cases} \pi_1(x_1^1) = 0.1 \\ \pi_1(x_2^1) = 1 \end{cases} \quad \text{and} \quad \begin{cases} \pi_2(x_1^2) = 1 \\ \pi_2(x_2^2) = 0.1 \end{cases}$$

For $i \in \{1, 2\}$, π_i generates a necessity measure Nec_i, a possibility measure Π_i and a credal set $\mathfrak{M}_i$. Let P_1 and P_2 be two probabilities respectively included in $\mathfrak{M}_1$ and $\mathfrak{M}_2$, whose values are indicated in Table 3.2.

$\mathcal{X}_1$	x_1^1	x_2^1
Nec_1	0	0.9
P_1	0.1	0.9
Π_1	0.1	1

$\mathcal{X}_2$	x_1^2	x_2^2
Nec_2	0.9	0
P_2	0.9	0.1
Π_2	1	0.1

Table 3.2: Probability distributions over $\mathcal{X}_1$ and $\mathcal{X}_2$

We first consider the Minimum copula $C_M(u, v) = \min(u, v)$. We construct a joint probability $P \in \mathfrak{M}_{robust}$ by joining P_1 and P_2 with C_M. Let us compare its value with the value of the bivariate necessity function $C_M(\text{Nec}_1, \text{Nec}_2)$ on the same event $\{x_2^1\} \times \{x_1^2\}$:

$$\begin{aligned} \text{Bel}_C(\{x_2^1\} \times \{x_1^2\}) &= C_M\left(\text{Nec}_1(x_2^1), \text{Nec}_2(x_1^2)\right) \\ &= \min(0.9,\ 0.9) = 0.9 \\ P(\{x_2^1\} \times \{x_1^2\}) &= F(x_2^1,\ x_1^2) - F(x_2^1,\ x_1^2) \\ &= C_M\left(P_1(\mathcal{X}_1), P_2(x_1^2)\right) - C_M\left(P_1(x_1^1), P_2(x_1^2)\right) \\ &= \min(1,\ 0.9) - \min(0.1,\ 0.9) = 0.8 \end{aligned}$$

Here $P < \text{Bel}_C$ on $\{x_2^1\} \times \{x_1^2\}$. Therefore $P \notin \mathfrak{M}_{mass}$. Because $P \in \mathfrak{M}_{robust}$, this proves that $\mathfrak{M}_{robust} \not\subseteq \mathfrak{M}_{mass}$.

Let us now compare the lower bound $\underline{P}$ of $\mathfrak{M}_{robust}$ with that of $\mathfrak{M}_{agg}$ (or $\mathfrak{M}_{mass}$ as they coincide on cylindrical sets). This time, we will be using the Łukasiewicz copula

$C_L(u,v) = \max(u+v-1, 0)$ as our dependency model. It holds that:

$$\begin{aligned}
\mathrm{Bel}_C(\{x_2^1\} \times \{x_1^2\}) &= C_L(\mathrm{Nec}_1(x_2^1), \mathrm{Nec}_2(x_1^2)) \\
&= \max(0,\ 0.9 + 0.9 - 1) = 0.8 \\
\underline{P}(\{x_2^1\} \times \{x_1^2\}) &= \inf_{P \in \mathfrak{M}_{robust}} P(\{x_2^1\} \times \{x_1^2\}) \\
&= \inf_{P \in \mathfrak{M}_{robust}} \left(F(x_2^1, x_1^2) - F(x_1^1, x_1^2)\right) \\
&= \inf_{P_1 \in \mathfrak{M}_1, P_2 \in \mathfrak{M}_2} \left(C_L\left(P_1(\mathcal{X}_1),\ P_2(x_1^2)\right) - C_L\left(P_1(x_1^1),\ P_2(x_1^2)\right)\right) \\
&= \inf_{P_1 \in \mathfrak{M}_1, P_2 \in \mathfrak{M}_2} \left(P_2(x_1^2) - \max(0,\ P_1(x_1^1) + P_2(x_1^2) - 1)\right) \\
&= \inf_{P_1 \in \mathfrak{M}_1, P_2 \in \mathfrak{M}_2} \max\left(P_2(x_1^2),\ P_2(x_1^2) - P_1(x_1^1) - P_2(x_1^2) + 1\right) \\
&\geqslant \min\left(\inf_{P_2 \in \mathfrak{M}_2} P_2(x_1^2),\ \inf_{P_1 \in \mathfrak{M}_1} \left(1 - P_1(x_1^1)\right)\right) = 0.9
\end{aligned}$$

On this event $\underline{P} > \mathrm{Bel}_C$, therefore $\mathfrak{M}_{agg} \not\subseteq \mathfrak{M}_{robust}$ and $\mathfrak{M}_{mass} \not\subseteq \mathfrak{M}_{robust}$.

We investigated the relationships between $\mathfrak{M}_{robust}$, $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$ in the case of marginals modeled by necessity functions. Those results will notably be used in Chapter 4. The next sections will consider other models: p-boxes.

3.2.3 Using the Natural Ordering of P-boxes

We first start this section by reminding some properties of p-boxes. P-boxes are special cases of belief functions that resemble the most well-known CDFs. They are defined with two CDFs $\underline{F}$, $\overline{F}$ such that $\underline{F} \leqslant \overline{F}$. Their focal sets a_α are of the form $a_\alpha = [\overline{F}^{-1}(\alpha), \underline{F}^{-1}(\alpha)]$ with $\alpha \in [0,1]$ [Destercke et al., 2008], where F^{-1} is the inverse of a CDF (or pseudo-inverse if not properly defined). It is thus possible to define a natural ordering on the focal sets. Let a_α and a_β be two focal sets of a p-box $[\underline{F}, \overline{F}]$, with $(\alpha, \beta) \in [0,1]^2$. The natural ordering $\preceq$ on focal sets is defined as follows:

$$a_\alpha \preceq a_\beta \ \Leftrightarrow\ \overline{F}^{-1}(\alpha) \leqslant \overline{F}^{-1}(\beta) \text{ and } \underline{F}^{-1}(\alpha) \leqslant \underline{F}^{-1}(\beta) \ \Leftrightarrow\ \alpha \leqslant \beta \tag{3.18}$$

We will consider this ordering when investigating the relationships between $\mathfrak{M}_{mass}$ and the other multivariate credal sets.

As stated previously in Section 2.3.5, p-boxes are very closely related to CDFs which can motivate one to apply Sklar's Theorem to the lower CDF and upper CDF respectively. Given n p-boxes $[\underline{F}_1, \overline{F}_1], \ldots, [\underline{F}_n, \overline{F}_n]$ defined over $\mathcal{X}_1, \ldots, \mathcal{X}_n$ and a copula C, we

can define the lower and upper bounds of a n variate CDF as:

$$\underline{F}_{\times} = C(\underline{F}_1,\ \ldots,\ \underline{F}_n)$$
$$\overline{F}_{\times} = C(\overline{F}_1,\ \ldots,\ \overline{F}_n)$$

Sklar's Theorem states that $\underline{F}_{\times}$ and $\overline{F}_{\times}$ are both CDFs, which means that $[\underline{F}_{\times}, \overline{F}_{\times}]$ is a multivariate p-box [Pelessoni et al., 2016, Montes et al., 2015] over cylindrical sets, defining a credal set $\mathfrak{M}$. Clearly, the bounds of $\mathfrak{M}_{robust}$ on cumulative events are the same as those of the credal set $\mathfrak{M}$ induced by the multivariate p-box $[C(\underline{F}_1,\ \ldots,\ \underline{F}_n),\ C(\overline{F}_1,\ \ldots,\ \overline{F}_n)]$. The bounds of $\mathfrak{M}_{robust}$ on cumulative events are therefore easy to compute, contrary to the case where the marginals are not p-boxes.

There is no clear relationship between $\mathfrak{M}_{robust}$ and $\mathfrak{M}_{mass}$ or $\mathfrak{M}_{agg}$. As it is the case for necessity functions, it is possible to find cases where the sets $\mathfrak{M}_{robust} \not\subseteq \mathfrak{M}_{mass}$ or $\mathfrak{M}_{agg} \not\subseteq \mathfrak{M}_{robust}$, as shown in the following example.

Example 13: Consider the following p-boxes:

$\mathcal{X}_1$	x_1^1	x_2^1
$\underline{F}_1$	0	1
$\overline{F}_1$	0.1	1

$\mathcal{X}_2$	x_1^2	x_2^2
$\underline{F}_2$	0.9	1
$\overline{F}_2$	1	1

Table 3.3: P-boxes over $\mathcal{X}_1$ and $\mathcal{X}_2$

The p-boxes from Table 3.3 lead to the same belief functions as in Example 12. Therefore, the same conclusions as in Example 12 hold, *i.e.* there are cases where $\mathfrak{M}_{robust} \not\subseteq \mathfrak{M}_{mass}$, $\mathfrak{M}_{mass} \not\subseteq \mathfrak{M}_{robust}$ and $\mathfrak{M}_{agg} \not\subseteq \mathfrak{M}_{robust}$.

The relationships between $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$ cannot be expressed using inclusions, as in Section 3.2.2 where marginals are necessity functions. Instead, we must consider an additional property on copulas called directional-convexity (D-convexity) and directional-concavity (D-concavity). A copula is called D-convex if it behaves like a convex function when considering each variable separately. The formal definition of D-convexity and D-concavity can be found Annex, Section 7.1. In this thesis, we also prove some relevant properties concerning D-convexity. However, their proof requires some lengthy and detailed explanations. As those results will only serve us for the following property, we chose to only present them in the Section 7.1 of the Annex, as Chapter 2 and Chapter 3 are already hard to follow for non experts.

The relationships between $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$ can be detailed as follows:

Proposition 8: When joining marginals represented by p-boxes using the natural ordering from (3.18) with a copula C, it holds that:

- if C is D-convex, then $\mathfrak{M}_{mass} \subseteq \mathfrak{M}_{agg}$.
- if C is D-concave, then $\mathfrak{M}_{agg} \subseteq \mathfrak{M}_{mass}$

The proof of this property is demonstrated in Section 7.2 of the Annex.

We saw that using the natural ordering on p-box, it is not possible to establish an inclusion relationship between $\mathfrak{M}_{robust}$ and $\mathfrak{M}_{mass}$ or $\mathfrak{M}_{agg}$ without further assumptions. It is possible to find relationships between $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$ by supposing an additional property on the copula, *i.e.* D-convexity or D-concavity.

3.2.4 Joining Different Types of Models

In previous sections, we considered cases where every marginal has the same uncertainty model. In this section, we will consider multivariate uncertainty models for which some marginals are modeled by possibilities and other marginals are modeled by p-boxes. In this setting, it is possible to derive results similar to those of Sections 3.2.2 and 3.2.3, when considering natural orderings on marginal focal sets.

Consider $\mathfrak{M}_i$ marginal credal sets, either defined by possibility distributions or by p-boxes. If all $\mathfrak{M}_i$ are modeled by possibilities, then we are in the setting of Section 3.2.2, and if they are all modeled by p-boxes, then we are in the setting of Section 3.2.3. Therefore, we assume here that there is at least one credal set defined by a possibility distribution and one credal set defined by a p-box.

When considering $\mathfrak{M}_{robust}$, we can draw the same conclusions as before, *i.e.* there is no clear relationship between $\mathfrak{M}_{robust}$ and $\mathfrak{M}_{mass}$ or $\mathfrak{M}_{agg}$. This is straightforward as Example 12 and Example 13 also apply in the current setting.

We will now consider the relationships between $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$. When marginals are all possibilities, we saw that $\mathfrak{M}_{mass} \subseteq \mathfrak{M}_{agg}$ (Section 3.2.2). When marginals are p-boxes, the inclusion holds for D-convex copula, but the reverse is true for D-concave copulas (*i.e.* $\mathfrak{M}_{mass} \supseteq \mathfrak{M}_{agg}$). Therefore, it may not seem obvious at first if obtaining a similar inclusion is possible when mixing uncertainty models for the marginals. The next proposition shows that it is still possible to find an inclusion, depending on the properties of the considered copula.

Proposition 9: When joining marginal credal sets induced by possibility distributions and p-boxes, using a copula C, the following inclusion holds:

- if C is D-convex, then $\mathfrak{M}_{mass} \subseteq \mathfrak{M}_{agg}$.
- if C is D-concave then $\mathfrak{M}_{agg} \subseteq \mathfrak{M}_{mass}$

Proof: We first provide an intuitive idea as to why this proposition holds. When marginals are possibility distributions, bounds of $\mathfrak{M}_{agg}$ and $\mathfrak{M}_{mass}$ are equal on cylindrical sets. However, when marginals are p-boxes, the bounds on cylindrical sets of $\mathfrak{M}_{agg}$ are less than those of $\mathfrak{M}_{mass}$ if the copula is D-convex, and are greater than those of $\mathfrak{M}_{mass}$ if the copula is D-concave. Bounds for p-boxes are thus more constraining than those for possibility distributions, which explains why this property is similar to Proposition 8.

The exact proof is similar to the proof of Proposition 8 using the fact that for every focal set a_p^i of a possibility distribution, we can still define $\underline{p}_i$ and $\overline{p}_i$ as $\underline{p}_i = 1$ and $\overline{p}_i = p$. The rest of the proof is identical. □

In this section, we presented results similar to those of Sections 3.2.2 and 3.2.3, when considering natural orderings on marginal focal sets. The next section will consider other ordering than the natural orderings consider until now.

3.2.5 Joining Belief Functions Using Other orderings

When considering belief functions that are neither possibilities nor p-boxes, a natural ordering on the focal sets might not always exist. For instance, consider a mass distribution function with the following focal sets $\{1,3\}$, $\{2,4\}$ and $\{1,4\}$. Defining an ordering between $\{1,3\}$, $\{2,4\}$ and $\{1,4\}$ is not trivial as it was for possibilities or p-boxes. We can even consider other orderings than the natural ordering on possibilities or p-boxes. For instance, consider a necessity function whose focal sets are: $\{1,2,3\}$, $\{2,3\}$ and $\{3\}$. The natural ordering on focal sets would be: $\{3\} \preceq \{2,3\} \preceq \{1,2,3\}$, while another ordering more similar to the order on reels could be $\{1,2,3\} \preceq \{2,3\} \preceq \{3\}$. Those examples illustrate the fact that the natural ordering does not always exist, or is not necessarily the obvious choice.

One could therefore consider an arbitrary ordering between focal sets when defining $\mathfrak{M}_{mass}$ as in (3.4). In this setting, a few questions arise: is there always an arbitrary ordering allowing $\mathfrak{M}_{robust} \subseteq \mathfrak{M}_{mass}$? If such an ordering exists, is it possible to explicit

it in advance without computing lower bounds of credal sets? It appears that there may not always exist an ordering that allows for $\mathfrak{M}_{robust} \subseteq \mathfrak{M}_{mass}$. To prove it, let us present an example where no ordering allows for either inclusion.

Example 14: Consider the Clayton copula for $\theta = 2$ and $n = 2$. The expression of the copula given in Table 2.1 can be simplified as follows:

$$\forall (u_1, u_2) \in \mathbb{R}^2 \backslash (0,0),\ C(u_1, u_2) = \frac{u_1 u_2}{\sqrt{{u_1}^2 + {u_2}^2 - {u_2}^2 {u_2}^2}}$$

and $C(0,0) = 0$ by continuity. Let us consider $\mathcal{X}_1 = \mathcal{X}_2 = \{1, 2, 3\}$, and two possibility distributions π_1, π_2 over $\mathcal{X}_1$ and $\mathcal{X}_2$ respectively:

$$\pi_1(1) = \pi_2(1) = 0.2 \qquad \pi_1(2) = \pi_2(2) = 1 \qquad \pi_1(3) = \pi_2(3) = 0.7$$

and the marginal credal sets $\mathfrak{M}(\pi_1)$, $\mathfrak{M}(\pi_2)$ they induce. Because both possibilities have the same focal sets, we will note them as $a_1 = \{2\},\ a_2 = \{2, 3\},\ a_3 = \{1, 2, 3\}$. We will first compute the lower bounds of $\mathfrak{M}_{robust}$ on specific events, and then compare it to the different values of the lower bounds of $\mathfrak{M}_{mass}$ depending on the orderings used on focal sets.

By joining $\mathfrak{M}(\pi_1)$ and $\mathfrak{M}(\pi_2)$ using C, we can obtain the lower probability $\underline{P}$ of $\mathfrak{M}_{robust}$ using Definition 18. If we consider the two events $E_1 = \{2\} \times \{2,\ 3\}$ and $E_2 = \{2,\ 3\} \times \{2\}$, it is possible to show that:

$$\underline{P}(E_1) = \underline{P}(E_2) \approx 0.131$$

which can be obtained for:

$$\begin{array}{lll} P_1(1) = 0 & P_1(2) = 0.3 & P_1(3) = 0.7 \\ P_2(1) = 0.2 & P_2(2) = 0.3 & P_2(3) = 0.5 \end{array}$$

for E_1, and the same holds with P_1 and P_2 reversed for E_2. Those values were estimated by running simulations, but their exact value can be computed by solving an optimization problem as C is differentiable (although it is a bit tedious to compute).

Depending on the orderings $\preceq_1, \preceq_2$ used to join the marginal masses, we can create a total of 6 belief functions $\mathrm{Bel}_C^{\preceq_1, \preceq_2}$ using Definition 19. For instance, if $\preceq_1$ and $\preceq_2$ are such that:

- $a_3 \preceq_1 a_1 \preceq_1 a_2$

- $a_1 \preceq_2 a_2 \preceq_2 a_3$

then the bivariate mass m_C would be defined as follows:

$$\begin{aligned} m_C(a_3,\ a_1) &= C(m_1(a_3),\ m_2(a_1)) \\ m_C(a_1,\ a_1) &= C(m_1(a_3)+m_1(a_1),\ m_2(a_1)) - C(m_1(a_3),\ m_2(a_1)) \\ m_C(a_3,\ a_2) &= C(m_1(a_3),\ m_2(a_1)+m_2(a_2)) - C(m_2(a_3),\ m_2(a_1)) \\ &\textit{etc.} \end{aligned}$$

We can compute values of $\mathrm{Bel}_C^{\preceq_1,\preceq_2}$ on E_1 and E_2 for all orderings $\preceq_1,\ \preceq_2$ on the focal sets of π_1 and π_2. The results are presented in Table 3.4. Base on those values, we can deduce that for all combinations of orderings $\preceq_1$ and $\preceq_2$, it holds that:

$$\begin{cases} \mathrm{Bel}_C^{\preceq_1,\preceq_2}(E_1) \leqslant \underline{P}(E_1) \\ \mathrm{Bel}_C^{\preceq_1,\preceq_2}(E_2) > \underline{P}(E_2) \end{cases} \quad \text{or} \quad \begin{cases} \mathrm{Bel}_C^{\preceq_1,\preceq_2}(E_1) > \underline{P}(E_1) \\ \mathrm{Bel}_C^{\preceq_1,\preceq_2}(E_2) \leqslant \underline{P}(E_2) \end{cases}$$

This proves that it is not always possible to find orderings $\preceq_1$ and $\preceq_2$ allowing $\mathfrak{M}_{mass} \subseteq \mathfrak{M}_{agg}$ or $\mathfrak{M}_{mass} \supseteq \mathfrak{M}_{agg}$.

$\mathrm{Bel}_C^{\preceq_1,\preceq_2}(\mathbf{E_1})$	$a_3^2 \preceq_2 a_2^2 \preceq_2 a_1^2$	$a_3^2 \preceq_2 a_1^2 \preceq_2 a_2^2$	$a_2^2 \preceq_2 a_3^2 \preceq_2 a_1^2$
$a_3^1 \preceq_1 a_2^1 \preceq_1 a_1^1$	0.296	0.296	0.224
$a_3^1 \preceq_1 a_1^1 \preceq_1 a_2^1$	0.254	0.254	0.240
$a_2^1 \preceq_1 a_3^1 \preceq_1 a_1^1$	0.296	0.296	0.224
$a_1^1 \preceq_1 a_3^1 \preceq_1 a_2^1$	**0.131**	**0.131**	0.279
$a_2^1 \preceq_1 a_1^1 \preceq_1 a_3^1$	0.291	0.291	0.216
$a_1^1 \preceq_1 a_2^1 \preceq_1 a_3^1$	**0.131**	**0.131**	0.279

$\mathrm{Bel}_C^{\preceq_1,\preceq_2}(\mathbf{E_1})$	$a_1^2 \preceq_2 a_3^2 \preceq_2 a_2^2$	$a_2^2 \preceq_2 a_1^2 \preceq_2 a_3^2$	$a_1^2 \preceq_2 a_2^2 \preceq_2 a_3^2$
$a_3^1 \preceq_1 a_2^1 \preceq_1 a_1^1$	0.259	0.180	0.180
$a_3^1 \preceq_1 a_1^1 \preceq_1 a_2^1$	0.208	0.270	0.270
$a_2^1 \preceq_1 a_3^1 \preceq_1 a_1^1$	0.259	0.180	0.180
$a_1^1 \preceq_1 a_3^1 \preceq_1 a_2^1$	0.251	0.293	0.293
$a_2^1 \preceq_1 a_1^1 \preceq_1 a_3^1$	0.236	0.218	0.218
$a_1^1 \preceq_1 a_2^1 \preceq_1 a_3^1$	0.251	0.293	0.293

$\text{Bel}_C^{\preceq_1,\preceq_2}(\mathbf{E_2})$	$a_3^2 \preceq_2 a_2^2 \preceq_2 a_1^2$	$a_3^2 \preceq_2 a_1^2 \preceq_2 a_2^2$	$a_2^2 \preceq_2 a_3^2 \preceq_2 a_1^2$
$a_3^1 \preceq_2 a_2^1 \preceq_2 a_1^1$	0.296	0.254	0.296
$a_3^1 \preceq_1 a_1^1 \preceq_1 a_2^1$	0.296	0.254	0.296
$a_2^1 \preceq_1 a_3^1 \preceq_1 a_1^1$	0.224	0.240	0.224
$a_1^1 \preceq_1 a_3^1 \preceq_1 a_2^1$	0.259	0.208	0.259
$a_2^1 \preceq_1 a_1^1 \preceq_1 a_3^1$	0.180	0.270	0.180
$a_1^1 \preceq_1 a_2^1 \preceq_1 a_3^1$	0.180	0.270	0.180

$\text{Bel}_C^{\preceq_1,\preceq_2}(\mathbf{E_2})$	$a_1^2 \preceq_2 a_3^2 \preceq_2 a_2^2$	$a_2^2 \preceq_2 a_1^2 \preceq_2 a_3^2$	$a_1^2 \preceq_2 a_2^2 \preceq_2 a_3^2$
$a_3^1 \preceq_2 a_2^1 \preceq_2 a_1^1$	**0.131**	0.291	**0.131**
$a_3^1 \preceq_1 a_1^1 \preceq_1 a_2^1$	**0.131**	0.291	**0.131**
$a_2^1 \preceq_1 a_3^1 \preceq_1 a_1^1$	0.279	0.216	0.279
$a_1^1 \preceq_1 a_3^1 \preceq_1 a_2^1$	0.251	0.236	0.251
$a_2^1 \preceq_1 a_1^1 \preceq_1 a_3^1$	0.293	0.218	0.293
$a_1^1 \preceq_1 a_2^1 \preceq_1 a_3^1$	0.293	0.218	0.293

Table 3.4: Rounded value of $\text{Bel}_C^{\preceq_1,\preceq_2}$ for E_1 and E_2 depending on the arbitrary orderings $\preceq_1$, $\preceq_2$. Values in bald font represent the minimal value attained by the different belief functions, where $\text{Bel}_C^{\preceq_1,\preceq_2}(E) = \underline{P}(E)$.

Example 14 illustrates that it is not always possible to find orderings allowing to find an inclusion relationship between $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$. In the case where such orderings exist, answering the question "if an ordering allowing $\mathfrak{M}_{robust} \subseteq \mathfrak{M}_{mass}$ exists, is it possible to explicit it in advance?" is not as trivial as the orderings will be dependent of the copula.

Conclusion

In this chapter, we presented three methods for joining marginal credal sets with a copula, which we named $\mathfrak{M}_{robust}$, $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$. Our contributions consisted notably in the different propositions investigating inclusion relationships between those sets. In particular, we considered the special case of the product copula, and the special cases where marginal credal sets are defined by possibility distributions and p-boxes. The multivariate uncertainty models presented in this chapter are crucial in uncertainty propagation problems. In the following chapter, we will consider such a propagation problem in the context of stereo matching. The multivariate credal sets defined in this chapter will be used to model and propagate the uncertainty, and some results proved here will be used to simplify the problem.

Chapter 4

Propagating the Uncertainty from Stereo Images to the Cost Volume

In the previous chapter, different methods for joining credal sets using a copula have been presented. In this chapter, we present a first methodological attempt to apply these results to an uncertainty propagation problem in a stereo matching context. Usage of imprecise models for various engineering problems different from stereo matching have been investigated in related work [Oberkampf et al., 2001, Beer et al., 2013, Patelli et al., 2014]. We here specifically use possibility distributions from Chapter 2 as uncertainty models on the intensity values of epipolar images used in stereo matching. We will then propagate the uncertainty from those images to the cost volume using results from Section 3.2.2. We will also show that propagating the uncertainty has the potential to improve the disparity map derived from the cost volume. This chapter takes up work and data already published in [Malinowski and Destercke, 2022a, Malinowski et al., 2023].

It is important to note that in this disparity estimation problem, we only take into account the uncertainty in our input image intensities, without considering the uncertainty of the cost function's ability to correctly identify the true disparity as its minimum. In other words, we do not consider the uncertainty arising from the difference between "two patches are very similar" and "the pixels at the center of the patches are homologous". We refer to Figure 1.15 and more broadly to the different discussions in Section 1.3.3 for more details. This chapter focuses on the propagation of uncertainty. We will thus consider a stereo matching pipeline with a simple cost function and no SGM regularization. Indeed, propagating the uncertainty through a cost volume optimization is too complex and computationally expensive to be solved with this chapter's method, even though it is an interesting problem to consider. We will consider the different problem of uncertainty modeling with SGM regularization in Chapter 5.

Remark: It is interesting to see that in [Quinio and Matsuyama, 1991, Quinio, 1992], the author considered using closed random sets (a concept related to Imprecise Probabilities (IP) and belief functions studied in Chapter 2) to model the uncertainty of a stereoscopic setup. They mainly consider the uncertainty arising from the limited resolution of digital images (as we do in this chapter), and from the precision of the calibration setup: focal length of cameras, baseline distance, orientation, and vergence angle of the cameras *etc.* They consider that the uncertainty from dense matching is not of epistemic nature, but of aleatoric nature, and thus do not model it by imprecise models as we do in Chapter 5. This hypothesis is justified because they achieve the stereo matching step using a window based ZNCC cost function (without SGM, as it was not published at the time), which, by nature, possesses strong links with probabilistic models.

4.1 Context and Hypotheses for Uncertainty Propagation

4.1.1 Considered Stereo Matching Pipeline

We consider the Sum of Absolute Differences (SAD) as our cost function, introduced in Section 1.3.3, and reminded here. Given patches $W_L \subset I_L$ and $W_R \subset I_R$ of the same shape with n pixels (usually squares):

$$\mathrm{SAD}(W_L, W_R) = \sum_{(p_i,q_i)\in(W_L,W_R)} |I_L(p_i) - I_R(q_i)| \tag{4.1}$$

where p_i and q_i are pixels at the same position i in their patch. For convenience purposes, we will refer to the Absolute Difference between two pixels as AD (when there is no sum involved). An illustration of windows from stereo images to compare is displayed in Figure 4.1, and an illustration of the AD between pixels and SAD cost function can be found in Figure 4.2. The simplicity of the formulation of the SAD cost function allows for a more didactic presentation of the uncertainty propagation problem. For the same reason, we do not consider the SGM regularization here. The modeling of uncertainty for more advanced cost functions and SGM methods will be considered in Chapter 5.

Remark: Although the SAD is not the best performing cost function for dense matching, it is both fast and easily parallelizable. It is often used for comparison in cost-based stereo algorithms [Hirschmuller and Scharstein, 2007, Žbontar and LeCun, 2016], or in other applications. The limitations of the SAD cost function are well-

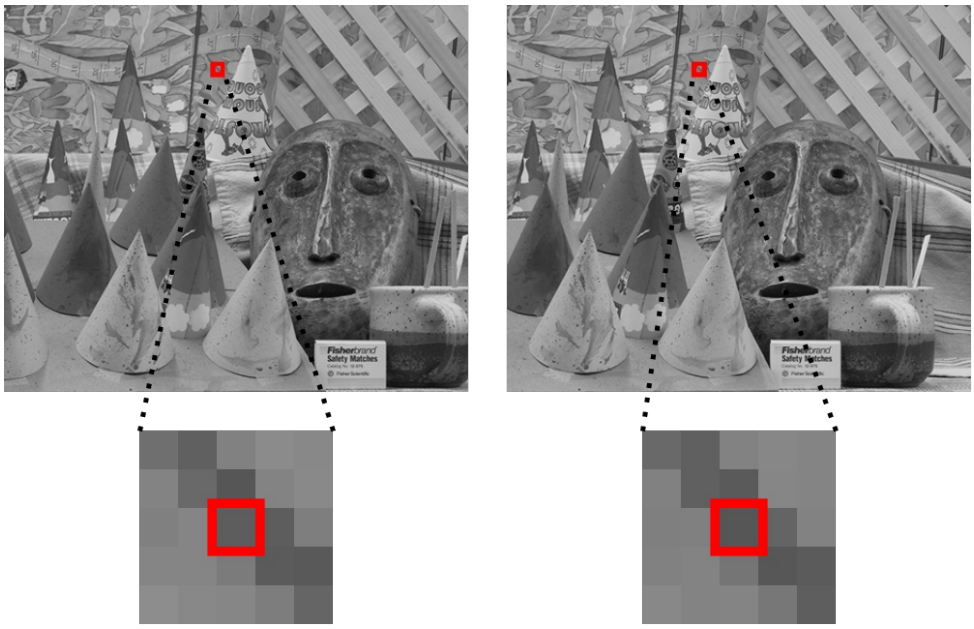


Figure 4.1: Homologous pixels in a pair of images. From [Malinowski et al., 2024a]

known, *i.e.* mostly a lack of robustness to gain and offset between patches. In the case of Middlebury images, illumination conditions and calibration of cameras are controlled, which are conditions well-suited for the SAD cost function. Considering a simple cost function such as SAD is relevant for different reasons:

- Simple stereo matching algorithms are often used as a quick and easy method for estimating the disparity.
- Simple cost functions (such as SAD, ZNCC, *etc.*) considered here are still used in other problems, such as video compression, for instance [Richardson, 2006].

In our experiments, we considered the "Cones" images from the 2003 Middlebury stereo dataset (`https://vision.middlebury.edu/stereo/data/scenes2003/`), as displayed in Figure 4.4(a). The two images have a of size 375×450, and the range of considered disparities is $[-60, 0]$.

4.1.2 Uncertainty Model for Epipolar Images Intensities

To maintain simplicity in this section, we will not consider panchromatic images, such as Pléiades products, encoding the reflectance values as positive integer, usually contained in $[0, 5000]$. Instead, we consider grayscale images that have intensity levels quantified within the range $[0, 255]$, which will represent our measurable space $\mathcal{X}$.

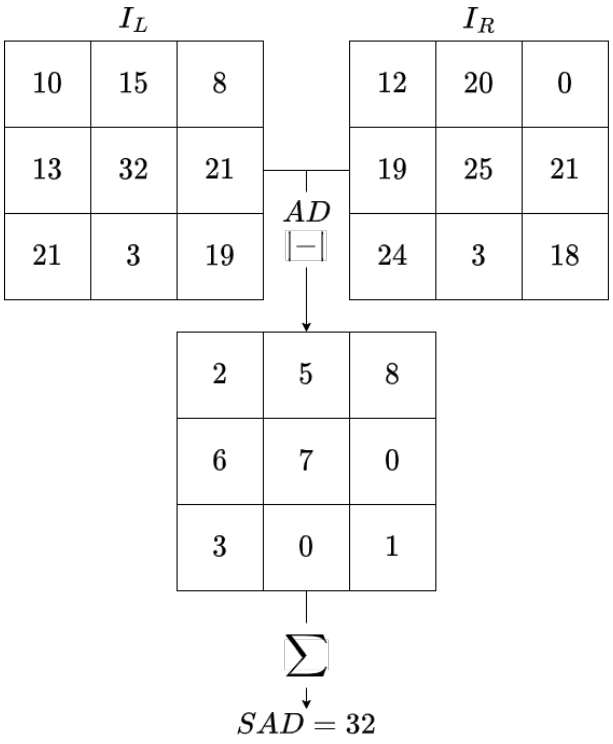


Figure 4.2: Diagram representing the SAD cost function between two 3×3 patches. From [Malinowski et al., 2024a]

> **Remark:** This hypothesis is not constraining, as we can easily normalize reflectances in order to encode them using 8-bit integers value, although doing so reduces the precision of the initial images. Moreover, this normalization step is often required to fit a given format, for instance if the images must be processed by a CNN trained on 8-bit integers.

We postulate that a pixel's intensity value can deviate around its observed value with a range of $\pm i_\sigma$, with the observed value being the most likely. This specific hypothesis remains simple and relatively plausible with regard to the processing leading to the epipolar images. We then assume that the uncertainty from the noise of the sensor capturing the image, from pre-processing steps such radiometric and geometric correction, or epipolar resampling (see Section 1.4.2) or from the quantification of observed radiometric values into 8-bit integers, are not exactly known, but can be model by a possibility distribution introduced in Section 2.3.4. Consequently, we model the uncertainty of each pixel $p \in I_L, I_R$ intensity with a possibility distribution π, centered around the observed intensity $i_p \in [0, 255]$:

$$\pi(i_p) = 1, \quad \pi(i_p \pm i_\sigma) = \alpha\,, \tag{4.2}$$

with $\alpha \in [0, 1]$. To remain simple, we chose $i_\sigma = 1$ in the following, which is relatively narrow but allows simplification without impacting our reasoning. Similarly, we do not consider multiple α values in order to limit the number of focal sets considered. Choosing

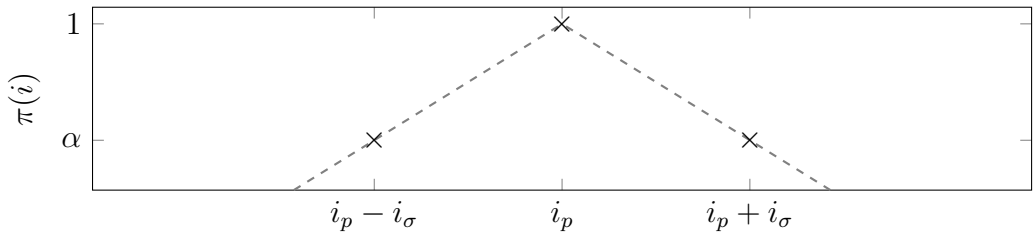


Figure 4.3: Possibility distribution for the intensity i of a pixel p

thus simple possibility distribution is convenient, but results obtained in this chapter can easily be extended to more complex possibility distributions. In our simulation, $\alpha = 0.3$ for pixels in the left image and $\alpha = 0.4$ for pixels in the right image. We use different values of α for the left and right images because the uncertainty model may vary between images due to differences in exposure, noise levels, or camera calibration. The values on themselves are chosen arbitrarily for the purpose of this example. From the credal set point of view of Definition 5, this model effectively states that we accept any probability distribution supported within $[i_p - 1, i_p + 1]$ where the probability measure P satisfies $\{P(A) \leq \sup_{i \in A} \pi(i)\}$ as an acceptable model for our uncertainty, as in Definition 13. The mass distribution function m_p from Definition 8 associated with this credal set possesses two focal sets a^p (Definition 9):

$$\begin{aligned} m_p(a_1^p = [\![i_p, i_p]\!]) &= 1 - \alpha \\ m_p(a_2^p = [\![i_p - 1, i_p + 1]\!]) &= \alpha \end{aligned} \tag{4.3}$$

with $[\![\cdot, \cdot]\!]$ referring to integer intervals. In particular, $[\![i_p, i_p]\!]$ corresponds to the singleton $\{i_p\}$.

> **Remark:** The hypothesis of modeling the uncertainty on image intensities by possibility distributions does not consider uncertainty from potentially bigger sources of errors, such as satellite vibrations during the acquisition, or errors in the computations of epipolar lines. Those type of errors have been encountered on some Pléiades acquisitions, and lead to significant biases and errors on the final DSM, that our simple model does not account for. We here suppose that the geometric models of each sensor as well as the epipolar geometry are perfectly known.

4.1.3 Dependency Model between Epipolar Images

We described the uncertainty models for pixels in both images, but we also need to define the dependency model between every pixel of both images. Indeed, as some pixels between images represent the light reflected by the same object, it seems natural that their (uncertain) values are correlated. In our case, we propose to model their dependency with the product copula if the pixels are not from the same physical object, meaning that the value of their intensities are independent. For pixels belonging to the same object, we model their dependency using a Gaussian copula with a covariance matrix R. Those copulas were introduced in Equation (4.4) from Chapter 2. We remind here the formulation of a Gaussian n-copula C_R:

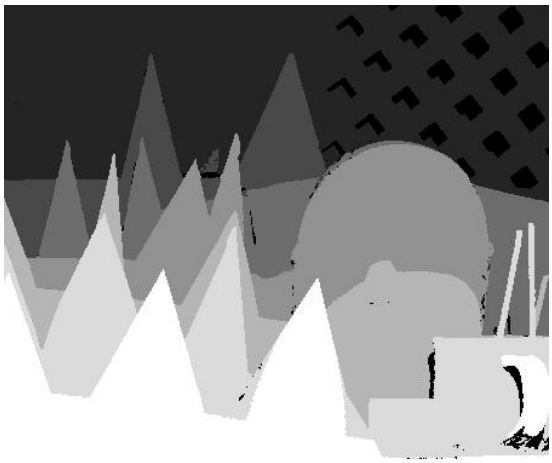
$$C_R(u_1, \ \ldots, \ u_n) = \Phi_R(\Phi^{-1}(u_1), \ \ldots, \ \Phi^{-1}(u_n)) \tag{4.4}$$

where Φ_R is the joint multivariate Cumulative Distribution Function (CDF) of a Gaussian variable with correlation matrix R, and Φ^{-1} is the inverse CDF of a univariate Gaussian variable. Gaussian copulas are popular and simple copulas used to represent dependencies between more than 2 variables. By comparison, the different 2-copulas presented in Section 2.4 cannot always be defined in more than 2 dimensions or possess a quite complex formulation. Another method for modeling the dependency for more than 2 variables is to express a n-copula as a combination of 2-copulas, which is called a vine copula ([Czado and Nagler, 2022]). However, this is a complex subject that is not adapted to our type of dependency, and is therefore not explored in this thesis.

In our experiments, the correlation values inside the covariance matrix are based on a segmentation $S : (I_L \cup I_R) \to [\![1, K]\!]$, $K \in \mathbb{N}$, of the images. This segmentation is the result of a *k-means* clustering performed on the ground truth disparity map. An example of such a clustering is presented in Figure 4.4, with $K = 8$. In real life scenarios, the correlation matrix must be estimated by other means, as the ground truth disparity is not available. We can for instance use a semantic segmentation algorithm [Hariharan et al., 2014, Ronneberger et al., 2015], or use other statistical estimators [Touloumis, 2015]. We suppose here that the correlation between pixels' intensities is well-known, as we also supposed we know the type of copula modeling their dependency. Given the segmentation S and two pixels $(p, q) \in (I_L \cup I_R)^2$, their covariance is here determined by:

$$\sigma(p, q) = \begin{cases} 1 & \text{if } p = q, \\ \rho_k, & \text{if } p \neq q \text{ and } S(p) = S(q)\,, \\ 0 & \text{otherwise}\,. \end{cases} \tag{4.5}$$

(a) Colored left image



(b) Proposed clustering of the image (*k-means* with $K = 8$)

Figure 4.4: Middlebury 2003 Cones left image, and a clustering computed from the disparity ground truth.

where $0 < \rho_k < 1$ is the correlation of pixels belonging to segment $k \in [\![1, N]\!]$. During our simulations, the segmentation contains $K = 8$ different clusters. For every k in $[\![1, K]\!]$, ρ_k is assigned a value between 0.9 and 1 in order to really emphasize their correlation. Given a set of pixels $\{p_1, \ldots, p_n\} \subseteq (I_L \cup I_R)^2$, their covariance matrix R is therefore:

$$R = \begin{bmatrix} 1 & \sigma(p_1, p_2) & \ldots & \sigma(p_1, p_{n-1}) & \sigma(p_1, p_n) \\ \sigma(p_2, p_1) & 1 & \ldots & \sigma(p_2, p_{n-1}) & \sigma(p_2, p_n) \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \sigma(p_{n-1}, p_1) & \sigma(p_{n-1}, p_2) & \ldots & 1 & \sigma(p_{n-1}, p_n) \\ \sigma(p_n, p_1) & \sigma(p_n, p_2) & \ldots & \sigma(p_n, p_{n-1}) & 1 \end{bmatrix} \tag{4.6}$$

In practice, we will only compute the correlation matrix between the two windows from the reference and secondary images that are compared. Both windows have a 3×3 shape, we thus consider Gaussian 18−copulas to model the dependency between pixels. This copula will be used for joining marginal masses in the uncertainty propagation step, as in Definitions 19 and 20 to compute the credal set $\mathfrak{M}_{mass}$. It will also be used in Section 4.3.2 to draw Monte Carlo samples, in order to estimate the credal set $\mathfrak{M}_{robust}$ from Definition 18.

> **Remark:** The segmentation is based on the ground truth disparity map. This means that two objects with similar disparities located at opposite sides of the image will be considered as belonging to the same object and thus correlated. In practice, those pixels are never compared, as we only measure the dissimilarity between small windows in a restricted disparity range. The clustering is thus only used at a *local*

scale.

Secondly, the segmentation allows computing the Gaussian copula, which will be used to propagate the uncertainty models. We will validate the uncertainty propagation in Section 4.3.2 with Monte Carlo samples using the same copula. As long as the copulas used are the same between the propagation and validation, it does not really matter which correlation matrix is used. We still tried to create a realistic but simple dependency model for the sake of the example, but it is not required from a theoretical point of view.

4.2 Propagating the Uncertainty with Belief Functions and a Copula

Having defined both marginal models for pixel intensities using possibility distributions and dependency models using copulas, we can now join them all to construct a multivariate uncertainty model, as seen in Section 3.1. We will see in this section how the multivariate models can then be used to compute the uncertainty regarding the cost curve.

4.2.1 From Multivariate Uncertainty Models to the Propagated Model

We first detail how multivariate models are used to propagate uncertainty in the precise case. We will then do the same in the imprecise setting by analogy.

Consider a mapping $f : \mathcal{X}_1 \times \mathcal{X}_2 \to \mathcal{Z}$ from a product space $\mathcal{X}_1 \times \mathcal{X}_2$ to a space $\mathcal{Z}$, which propagates multiple random variables X_1, X_2 to a new random variable $Z = f(X_1,\ X_2)$. In our case, f will be the SAD cost function propagating the intensities to the matching cost. When considering precise probabilities, the probability of Z on atoms z is obtained by summing the probabilities of all events $X_1 = x_1$, $X_2 = x_2$ whose image by f is z:

$$\forall z \in \mathcal{Z}, P_Z(z) = \sum_{\substack{x_1,x_2 \\ z=f(x_1,x_2)}} P(x_1, x_2). \tag{4.7}$$

where $P(x_1, x_2)$ is the joint probability, which is linked to its marginals by a copula C. P_Z is completely determined by evaluating every combination of atoms of $\mathcal{X}_1$ and $\mathcal{X}_2$.

Example 15: Consider the same setting as Example 8, where a dealer throws two coins in a separate room. The coins seem fair when looked at independently. In this example, a coin landing on heads rewards you with 1€ (or any currency of your choice), and a coin landing on tails rewards you with 0€. So you earn 2€ if both coins

land on heads, 1€ if only one coin land on heads, and 0€ if both coins land on tails. We are interested in the uncertainty regarding your earnings, noted Z.

In Example 8, we consider 3 different cases, each leading to a different copula C modeling the dependency between the probability P_1 of the first coin and P_2 of the second coin. For each copula, we then computed the joint probability P.

In the first case, where coin throws were independent, we saw that the joint probability P was:

$$P(\text{heads},\ \text{heads}) = 0.25 \qquad P(\text{tails},\ \text{tails}) = 0.25$$
$$P(\text{heads},\ \text{tails}) = 0.25 \qquad P(\text{tails},\ \text{heads}) = 0.25$$

In that case, it holds that the probability P_Z of our earnings is:

$$P_Z(Z = 2) = P(\text{heads},\ \text{heads}) = 0.25$$
$$P_Z(Z = 1) = P(\text{heads},\ \text{tails}) + P(\text{tails},\ \text{heads}) = 0.5$$
$$P_Z(Z = 0) = P(\text{tails},\ \text{tails}) = 0.25$$

In the second case, where coin throws were rigged to land on the same side, we saw that the joint probability P was:

$$P(\text{heads},\ \text{heads}) = 0.5 \qquad P(\text{tails},\ \text{tails}) = 0.5$$
$$P(\text{heads},\ \text{tails}) = 0 \qquad P(\text{tails},\ \text{heads}) = 0$$

Following the same methodology, we have:

$$P_Z(Z = 2) = 0.5 \qquad P_Z(Z = 1) = 0 \qquad P_Z(Z = 0) = 0.5$$

In the third case, where coin throws were rigged to land on opposite sides, we saw that the joint probability P was:

$$P(\text{heads},\ \text{heads}) = 0 \qquad P(\text{tails},\ \text{tails}) = 0$$
$$P(\text{heads},\ \text{tails}) = 0.5 \qquad P(\text{tails},\ \text{heads}) = 0.5$$

Therefore:

$$P_Z(Z = 2) = 0 \qquad P_Z(Z = 1) = 1 \qquad P_Z(Z = 0) = 0$$

Determining every (x_1, x_2), whose image by f equals z, is not always trivial. This becomes even more complex when considering $n > 2$ marginal variables. Similarly, the

joint probability $P(x_1, x_2)$ is computed using a H-volume, which is the sum of 2^n terms, thus also increasing exponentially with the dimension.

There are multiple ways of extending Equation (4.7) to the imprecise setting, as there are multiple ways of aggregating imprecise models using a copula. We described in Chapter 3 three methods for joining marginal credal sets, creating three different multivariate credal sets $\mathfrak{M}_{robust}$, $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$.

The robust approach of extending Equation (4.7) is based on the robust approach from Section 3.1.1. Given n marginal credal sets $\mathfrak{M}_i$, we can join them into a credal set $\mathfrak{M}_{robust}$ using a copula C, as in (3.3). The propagated uncertain model $\mathfrak{M}^Z_{robust}$ is then defined as:

$$\mathfrak{M}^Z_{robust} = \{P_Z \mid \forall z \in Z,\ P_Z(z) = \sum_{\substack{x_1,\ ...,\ x_n \\ z=f(x_1,\ ...,\ x_n)}} P(x_1,\ \ldots,\ x_n),\ P \in \mathfrak{M}_{robust}\} \quad (4.8)$$

Practically, this set is computed by sampling every probability P_i from each marginal credal set $\mathfrak{M}_i$ and joining them using Sklar's Theorem into a multivariate probability P. Then for each $z = P(x_1,\ \ldots,\ x_n)$, we can compute P_Z from P using (4.7). In a few words, each sample $(P_1,\ \ldots,\ P_n)$ leads to a new P, itself leading to a new P_Z. Sampling through every $(P_1,\ \ldots,\ P_n)$ thus leads to the estimation of the uncertain model of Z. This method is complicated to compute, but correctly propagates the uncertainty. In the case of the SAD cost function, the robust credal set is defined as:

$$\mathfrak{M}^{\text{SAD}}_{robust} = \{P_{\text{SAD}} \mid \forall z \in \text{SAD},\ P_{\text{SAD}}(z) = \sum_{\substack{p_1,\ ...,\ p_{18} \\ z=\text{SAD}(p_1,\ ...,\ p_{18})}} P(p_1,\ \ldots,\ p_{18}),\ P \in \mathfrak{M}_{robust}\} \quad (4.9)$$

We saw that it was easier to compute $\mathfrak{M}_{mass}$ than $\mathfrak{M}_{robust}$. Thus, another way of approximating the uncertainty model of Z is to compute it using $\mathfrak{M}_{mass}$. This is done by replacing the probability on atoms from Equation (4.7) with the joint mass m_C from Definition 19 [Gray et al., 2021]. Consider n uncertain variables X_i, each modeled by a mass distribution function whose j-th focal set is noted a^i_j. Given their multivariate mass m_C, it is possible to compute the mass distribution function m_Z of a random set $Z = f(X_1,\ \ldots,\ X_n)$ as:

$$\forall a^Z \subseteq \mathcal{Z}, m_Z(a^Z) = \sum_{\substack{a^1_i,\ ...,\ a^n_j \\ a^Z=f(a^1_i,\ ...,\ a^n_j)}} m_C(a^1_i,\ \ldots,\ a^n_j) \quad (4.10)$$

leading to a credal set $\mathfrak{M}^Z_{mass}$:

$$\mathfrak{M}^Z_{mass} = \{\ P_Z \mid \forall A \subseteq \mathcal{Z},\ P_Z(A) \geqslant \sum_{a^Z \subseteq A} m_Z(a^Z)\} \tag{4.11}$$

$$= \{\ P_Z \mid \forall A \subseteq \mathcal{Z},\ P_Z(A) \geqslant \sum_{\substack{a^1,\ ...,\ a^n \\ f(a_1,\ ...,\ a_n) \subseteq A}} m_C(a^1,\ \ldots,\ a^n)\ \} \tag{4.12}$$

In order to compute the propagated mass m_Z (and its associated belief function) from Equation (4.10), two difficulties arise. The first one is to determine what the focal sets a_Z of m_Z will be, which corresponds to the subscript $a^Z = f(a^n_1,\ \ldots,\ a^n_j)$ of the previous sum. Computing the image of f for every combination of focal sets $(a^1_i,\ \ldots,\ a^n_j)$ is even more difficult than in the precise case, as we are computing images of sets instead of real numbers. The second difficulty is to compute the joint mass m_C, as in the case of the SAD it requires computing the H-volume of a 18-copula. Those difficulties will be addressed in Sections 4.2.2 and 4.2.3

We saw in Chapter 3 that in the situation where marginals are possibility distributions, $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$ have the same bounds on Cartesian products of events (see Section 3.2.2). We will thus only compute the lower bounds Bel_C of $\mathfrak{M}_{mass}$ as it also provides the bounds of $\mathfrak{M}_{agg}$ on those events.

4.2.2 Determining the Bounds of the Propagated Focal Sets

In this section, we will detail how we compute the bounds of the SAD from marginal focal sets. Computing the image of sets should be approached with caution in the general case. However in our case, because we chose marginal focal sets with a simple expression, and because we are using a relatively regular cost function, computing the image is significantly easier.

Given a pixel p, we consider the mass distribution m_p of Section 4.1.2 and its two focal sets a^p_1 and a^p_2 from Section 4.1.2. For every pair of pixels $p \in I_L, q \in I_R$, we note $\mathrm{AD}_{pq} = |i_p - i_q|$, where i refers to a pixel's intensity. Given m_p, there exist 3 focal sets related to the absolute difference:

- a^{AD}_1 is the image of the AD of a^p_1 and a^q_1
- a^{AD}_2 is the image of the AD of a^p_2 and a^q_1 or a^p_1 and a^q_2
- a^{AD}_3 is the image of the AD of a^p_2 and a^q_2

The non-monotonicity of the absolute value around 0 needs to be taken into account to compute their exact image through the AD. Indeed, if a value x is in $[-1, 1]$, then its

absolute value will be in $[0, 1]$. Applying this remark to the AD yields the following focal sets:

$$
\begin{aligned}
a_1^{\mathrm{AD}} &= [\![\mathrm{AD}_{pq},\ \mathrm{AD}_{pq}]\!]\,, \\
a_2^{\mathrm{AD}} &= [\![\max(0, \mathrm{AD}_{pq} - 1),\ \mathrm{AD}_{pq} + 1]\!]\,, \\
a_3^{\mathrm{AD}} &= [\![\max(0, \mathrm{AD}_{pq} - 2),\ \mathrm{AD}_{pq} + 2]\!]\,,
\end{aligned}
$$

Example 16 provides numerical examples of the value of AD focal sets.

Example 16: Let us compute the focal sets of the absolute difference between two pixels. Let p and q be pixels of the left and right image, respectively.

- First, suppose that $i_p = 100$ and $i_q = 150$. Focal sets of i_p and i_q are then:

$$
\begin{aligned}
a_1^p &= [\![i_p,\ i_p]\!] = \{100\} & a_1^q &= [\![i_q,\ i_q]\!] = \{150\} \\
a_2^p &= [\![i_p - 1,\ i_p + 1]\!] & a_2^q &= [\![i_q - 1,\ i_q + 1]\!] \\
&= \{99, 100, 101\} & &= \{149, 150, 151\}
\end{aligned}
$$

Focal sets of the absolute difference between p and q will therefore be:

$$
\begin{aligned}
a_1^{\mathrm{AD}} &= \{|a_1^p - a_1^q|\} = \{50\} \\
a_2^{\mathrm{AD}} &= \{|a_2^p - a_1^q|\} = \{|a_1^p - a_2^q|\} = [\![49,\ 51]\!] \\
&= [\![50 - 1,\ 50 + 1]\!] \\
a_3^{\mathrm{AD}} &= \{|a_2^p - a_2^q|\} = [\![48,\ 52]\!] \\
&= [\![50 - 2,\ 50 + 2]\!]
\end{aligned}
$$

- Suppose now that $i_p = 100$ and $i_q = 100$. Focal sets of i_p and i_q are then the same:

$$
\begin{aligned}
a_1^p &= a_1^q = [\![i_p,\ i_p]\!] = \{100\} \\
a_2^p &= a_2^q = [\![i_p - 1,\ i_p + 1]\!] \\
&= \{99, 100, 101\}
\end{aligned}
$$

Focal sets of the absolute difference between p and q will therefore be:

$$
\begin{aligned}
a_1^{\mathrm{AD}} &= \{|a_1^p - a_1^q|\} = \{0\} \\
a_2^{\mathrm{AD}} &= \{|a_2^p - a_1^q|\} = \{|a_1^p - a_2^q|\} = [\![0,\ 1]\!] \\
a_3^{\mathrm{AD}} &= \{|a_2^p - a_2^q|\} = [\![0,\ 2]\!]
\end{aligned}
$$

Focal sets of the final SAD are then computed by simply summing the bounds of focal sets of the 9 AD (as in Figure 4.2), for every combination $(a_{k_1}^{\mathrm{AD}_1},\ \ldots,\ a_{k_9}^{\mathrm{AD}_9})_{k_i \in [\![1,3]\!]}$ of those focal sets:

$$a^{\mathrm{SAD}} = \sum_{i=1}^{9} a_{k_i}^{\mathrm{AD}_i} \tag{4.13}$$

In many cases, different combinations of AD focal sets will lead to the same SAD focal set. Actually, if every AD is greater than 2 so that each a_i^{AD} is symmetric with regard to its AD, as in the first bullet point of Example 16, there will only be 19 focal sets a^{SAD} for the SAD. They are of the following form:

$$a^{\mathrm{SAD}} = [\![\mathrm{SAD} - t, \mathrm{SAD} + t]\!],\ \text{with } t \in [\![0,\ 18]\!] \tag{4.14}$$

In comparison, there are $3^9 = 19\,683$ different combinations of AD focal sets.

When every AD is greater than 2, it holds that:

$$a^{\mathrm{SAD}} = [\![\mathrm{SAD} - t, \mathrm{SAD} + t]\!],\ \text{with } t \in [\![0,\ 18]\!]$$

This can easily be seen by considering AD that are similar to the first point of Example 16. This equation translates the fact that focal sets of the SAD form a nested family of sets, *i.e.* there is an inclusion relationship linking them all. This also means that $\mathrm{Bel}_{\mathrm{SAD}}$ is actually a necessity function introduced in Definition 13. We can thus compute a possibility distribution π_{SAD} to represent the uncertainty of the SAD. This can be useful to graphically represent the uncertainty of the SAD, if we later need to build joint uncertainty models using the SAD, or if we want to propagate the SAD uncertainty even further.

We proved in [Malinowski et al., 2024a] that in order to propagate marginal possibility distributions π_i into a possibility distribution π_Z using a copula and a propagating function f, a sufficient condition is that f and π_i verify:

- f is a monotone function applied to a linear combination $\alpha_1 X_1 + \ldots + \alpha_n X_n + \beta$ of marginal variables X_i
- Each π_i is symmetrical and uni-modal (meaning that all values x_i such that $\pi_i(x_i) = 1$ are adjacent). For instance, all triangular possibility distributions verify this condition

Figure 4.5 displays an example of a uni-modal symmetrical possibility distribution.

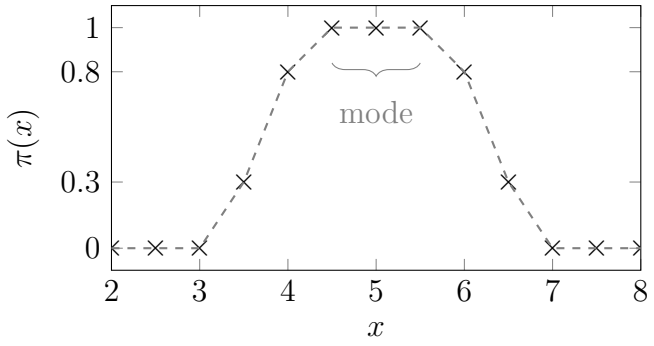


Figure 4.5: A uni-modal possibility distribution π

4.2.3 Computing the Mass of Propagated Focal Sets

Now that the bounds of the SAD have been computed, we need to compute the associated mass. Computing the joint mass over two 3×3 windows is significantly more complex. For each combination of marginal focal sets, the joint mass m_C is computed using the H-volume of a 18-copula, involving a sum of 2^{18} terms. Given that the uncertainty of each of the 18 pixels is represented by 2 focal sets, we need to evaluate 2^{18} combinations of these marginal masses in total. This computation can thus become quite costly in memory and computation time, especially when computing it over a whole image.

In the case of the family of Gaussian copulas, their expression given by Equation (4.4) show that we need to compute the multivariate CDF. A 18-variate Gaussian CDF does not possess a known analytic formula; it is thus computed by integrating its Probability Density Function (PDF) (so integrating a 18-variate function), as expressed below:

$$F(x_1, \ldots, x_{18}) = \int_{-\infty}^{x_1} \ldots \int_{-\infty}^{x_{18}} \frac{1}{\sqrt{(2\pi)^{18}|R|}} \exp(-\frac{1}{2} \begin{bmatrix} x_1 & \ldots & x_{18} \end{bmatrix} R^{-1} \begin{bmatrix} x_1 \\ \ldots \\ x_{18} \end{bmatrix}) dx_1 \ldots dx_{18} \quad (4.15)$$

Where $|R|$ is the determinant of R. Computing this CDF can quickly become time-consuming ($40ms$ on average[1]). For each pixel and each disparity, we need to compute the joint mass of 2^{18} focal sets in total, necessitating 2^{18} evaluations of its copula each time. Because there are around 170 000 pixels and 60 disparities to be evaluated, the processing time is too large to be computed as such. We will instead see that we can leverage specificities of our problem to drastically reduce the computation time.

The first idea is to notice that if we can divide our variables into multiple mutually independent sets of variables, then the evaluation of the 18-copula can be separated into the evaluation of multiple lower dimension copulas. To verify this statement, consider

[1]With an AMD EPYC7713 64-Core Processor at 2GHz, using Python and the SciPy library

the following independent sets of variables $\{X_1,\ \ldots,\ X_k\}$ and $\{X_{k+1},\ \ldots,\ X_n\}$ with $k \in [\![1, n-1]\!]$. Let $F_1,\ \ldots,\ F_n$ be their marginals CDFs. And let $F_{(1,\ \ldots,\ n)}$ be the joint CDF of all variables, $F_{(1,\ \ldots,\ k)}$ the joint CDF of the first set of variables and $F_{(k+1,\ \ldots,\ n)}$ the joint CDF of the second set. The independence between the two sets means that for all $(x_1,\ \ldots,\ x_n) \in \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$ it holds that:

$$F_{(1,\ \ldots,\ n)}(x_1,\ \ldots,\ x_n) = F_{(1,\ \ldots,\ k)}(x_1,\ \ldots,\ x_k) \cdot F_{(k+1,\ \ldots,\ n)}(x_{k+1},\ \ldots,\ x_n)$$

Using Sklar's theorem, there exist a n-copula C, a k-copula C' and a $n-k$-copula C'' respectively linking $F_{(1,\ \ldots,\ n)}$, $F_{(1,\ \ldots,\ k)}$, and $F_{(k+1,\ \ldots,\ n)}$ to their marginals:

$$\begin{aligned} &C(F_1(x_1),\ \ldots,\ F_n(x_n)) = \\ &\qquad C'(F_1(x_1),\ \ldots,\ F_k(x_k)) \cdot C''(F_{k+1}(x_{k+1}),\ \ldots,\ F_n(x_n)) \end{aligned} \tag{4.16}$$

> **Remark:** We stated earlier that we did not consider vine copulas [Czado and Nagler, 2022], which are a way of constructing multivariate copulas by composition of bivariate (conditional) copulas. The decomposition C into C' and C'' actually follows the same idea of decomposing a copula into smaller copulas. Similarly, this decomposition also reminds the concept of hierarchical copulas [Joe, 1997]. So even though we are not using vine or hierarchical copulas, we use a similar philosophy in our computations.

Establishing Equation (4.16) becomes interesting once we put it in relation with the following property:

Proposition 10: *H-Volume factorizing*

Let $1 < k < n$. If a n-copula C can be expressed as the product of a k-copula C' and a $(n-k)$-copula C'', then the H-volume of C is the product of the H-volume H' of C' and the H-volume H'' of C''. This means that for all $(u_1,\ \ldots,\ u_n) \in [0,1]^n$ and for all $(v_1,\ \ldots,\ v_n) \in [0,1]^n$ such that $u_i \leqslant v_i$, it holds that:

$$H_{u_1,\ \ldots,\ u_n}^{v_1,\ \ldots,\ v_n} = {H'}_{u_1,\ \ldots,\ u_k}^{v_1,\ \ldots,\ v_k} \cdot {H''}_{u_{k+1},\ \ldots,\ u_n}^{v_{k+1},\ \ldots,\ v_n} \tag{4.17}$$

> ***Proof:*** Let $1 < k < n$, C a n-copula, C' a k-copula and C'' a $n-k$ copula. H, H',

H'' are the respective H-volume of C, C', C''. Then:

$$
\begin{aligned}
H'^{v_1,\ \ldots,\ v_k}_{u_1,\ \ldots,\ u_k} \cdot H''^{v_{k+1},\ \ldots,\ v_n}_{u_{k+1},\ \ldots,\ u_n} &= \left(\sum_{w_i \in \prod_{i=1}^{k} \{u_i, v_i\}} (-1)^{|\{w_i \ |\ w_i = u_i\}|} C'(w_1,\ \ldots,\ w_k) \right) \\
&\quad \times \left(\sum_{w_j \in \prod_{j=k+1}^{n} \{u_j, v_j\}} (-1)^{|\{w_j \ |\ w_j = u_j\}|} C''(w_{k+1},\ \ldots,\ w_n) \right) \\
&= \sum_{w_i \in \prod_{i=1}^{k} \{u_i, v_i\}} \times \sum_{w_j \in \prod_{j=k+1}^{n} \{u_j, v_j\}} (-1)^{|\{w_i \ |\ w_i = u_i,\ i \leqslant k\}|} \\
&\quad \times (-1)^{|\{w_j \ |\ w_j = u_j,\ j > k\}|} C'(w_1,\ \ldots,\ w_k) C''(w_{k+1},\ \ldots,\ w_n) \\
&= \sum_{w_i \in \prod_{i=1}^{n} \{u_i, v_i\}} (-1)^{|\{w_i \ |\ w_i = u_i\}|} C'(w_1,\ \ldots,\ w_k) \\
&\quad \times C''(w_{k+1},\ \ldots,\ w_n) \\
&= H^{v_1,\ \ldots,\ v_n}_{u_1,\ \ldots,\ u_n}
\end{aligned}
$$

□

Using the result of Proposition 10, we can now compute the joint mass as a product of two lower dimension copulas. It is easier to compute as we only integrate a k-dimensional function and a $n-k$-dimensional function, instead of a n-dimensional one. Similarly, the H volume is not the sum of 2^{18} terms anymore, but the sum of 2^k and 2^{n-k} terms.

For comparison, consider that we can split the aforementioned Gaussian 18-copula into two Gaussian 9-copulas. Computing the value of a single mass naively takes around $10\,500s$, but now takes around $6s$, so around 1 700 times faster. This demonstrates the substantial time savings achieved by decomposing the problem into smaller, independent parts. These improvements apply directly to our application. Indeed, Equation (4.5) yields the following correlation between two pixels p, q given the segmentation S:

$$
\sigma(p, q) = \begin{cases} 1 & \text{if } p = q, \\ \rho_k, & \text{if } p \neq q \text{ and } S(p) = S(q)\,, \\ 0 & \text{otherwise}\,. \end{cases}
$$

From this, we can split the set $\{p_1,\ \ldots,\ p_{18}\}$ of pixels in at most $K = 8$ mutually independent sets $S_k = \{p_i \mid S(p_i) = k\}$, each set containing pixels from the same cluster. We can thus compute the H-volume of each cluster independently.

Another additional way of reducing the computation time is to avoid computing the same H-volume multiple times. Consider a set S_k with k_L pixels from the left image and

k_R pixels from the right image. The correlation matrix R_k for this cluster is:

$$R_k = \begin{pmatrix} 1 & & \rho_k & \dots & \rho_k \\ & & & & \vdots \\ \rho_k & & 1 & & \rho_k \\ \vdots & & & & \\ \rho_k & \dots & \rho_k & & 1 \end{pmatrix} \tag{4.18}$$

which implies that the Gaussian $(k_L + k_R)$-copula for this set is symmetrical. The joint mass $m_C^{S_k}$ of this set is then also symmetrical. Pixels from the left image will share the same mass m_L for their two focal sets and pixel from the right image will also share the same mass m_R, therefore computing $m_C^{S_k}$ on every possible combination of cumulative masses is redundant.

Example 17: Let us imagine a set S_k with $k_L = 2$ pixels p_1, p_2 from the left image and $k_R = 2$ pixels p_3, p_4 from the right image. Each pixel p_i has two focal sets a_1^i and a_2^i. We defined the mass distribution functions in Section 4.1.2 such that:

$$\begin{aligned} m_L(a_1^1) &= m_L(a_1^2) & m_L(a_2^1) &= m_L(a_2^2) \\ m_R(a_1^3) &= m_R(a_1^4) & m_R(a_2^3) &= m_R(a_2^4) \end{aligned}$$

Because of the symmetry of the Gaussian 4-copula of the set S_k, the joint mass $m_C^{S_k}$ computed as the H-volume on cumulative masses (Definition 19) verifies:

$$\begin{aligned} m_C^{S_k}(a_1^1, a_2^2, a_1^3, a_2^4) &= H^{\substack{m_L(a_1^1), & m_L(a_1^1)+m_L(a_2^2), & m_R(a_1^3), & m_R(a_1^4)+m_R(a_2^4) \\ 0, & m_L(a_1^2), & 0, & m_R(a_1^4)}} \\ &= H^{\substack{m_L(a_1^1)+m_L(a_1^1), & m_L(a_1^2), & m_R(a_1^3)+m_R(a_2^3), & m_R(a_1^4) \\ m_L(a_1^1), & 0, & m_R(a_1^3), & 0}} \\ &\quad \text{(by symmetry of the copula)} \\ &= m_C^{S_k}(a_2^1, a_1^2, a_2^3, a_1^4) \end{aligned}$$

We can see that we do not need to compute $m_C^{S_k}$ on every possible combination of focal sets as many combinations have the same joint mass $m_C^{S_k}$.

For each set S_k, we only have to compute $(k_L + 1) \cdot (k_R + 1)$ values of $m_C^{S_k}$ rather than $2^{k_L+k_R}$.

We saw earlier that it was complex to compute the joint mass of the SAD in general. However, by taking advantage of the potential factorization of the copula into smaller

copulas, and by leveraging some symmetries of the problem, we can greatly reduce the time and number of computations required.

4.3 Results and Discussions

In the previous section, we described how we can use $\mathfrak{M}_{robust}$ and $\mathfrak{M}_{mass}$ to propagate the uncertainty from the input images into the uncertainty of the SAD. Evaluating this uncertainty for every pixel and every considered disparity, results in the uncertainty models of every value of the cost volume (see Equation (1.7) from Section 1.3.3 for more details on the cost volume). We saw in Section 3.2.2 that $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{robust}$ are different credal sets, and that neither set is guaranteed to be included in the other. This section will then compare the various models and see if $\mathfrak{M}^Z_{mass}$ can be used to approximate $\mathfrak{M}^Z_{robust}$. Section 4.3.1 will present visualizations of $\mathfrak{M}^Z_{mass}$ using plausibility envelopes. Section 4.3.2 will present visualizations of $\mathfrak{M}^Z_{robust}$ using Monte Carlo samples and compare them to the plausibility envelopes. Finally, Section 4.3.3 will estimate the potential improvements unlocked by estimating the uncertainty.

4.3.1 Envelopes Defined by Plausibility Levels

Having defined efficient ways of computing the SAD focal sets in Section 4.2.3 and the joint mass in Section 4.2.2, we can determine the belief function Bel_{SAD} associated with every estimation of the SAD between two 3×3 windows. Bel_{SAD} is deducted from the mass m_C^{SAD} computed in Equation (4.10).

For each pixel $p_L = (row,\ col)$ in the left image and for each disparity d, we calculated the SAD cost between the window centered on p_L in the left image and the window centered on $p_R = (row,\ col + d)$ in the right image. We are usually interested in going through each considered disparity d to obtain the cost curve for pixel p_L. From this cost curve, a *winner-takes-all* strategy is applied to find the correct disparity. Because of its significance, we aim to visualize the uncertainty of the entire cost curve, not just a single value.

It can be hard to graphically represent focal sets, or similarly belief functions, especially when there are many sets to consider. We are usually more keen to represent uncertainty on singletons, as we do with probability densities, possibility distributions or, to a certain extent, p-boxes (although in that case singletons represent cumulative events). Following that logic, we will consider the plausibility of singletons as a way of representing the uncertainty graphically.

Remark: Considering the belief on singletons instead of the plausibility does not make much sense as the belief of singletons is very often null. The only exception is the precise SAD value, because it is the only singleton that is also a focal set.

We will first explain how we graphically represent the uncertainty in the special case of possibility distributions, and then extend this representation to all SAD belief functions.

As shown in Section 4.2.2, focal sets representing the SAD uncertainty are defined as intervals containing the "precise" SAD value. We saw in a remark from Section 4.2.2 that the uncertainty of the SAD can be represented by a possibility distribution π_{SAD} in the special case where all AD leading to the SAD are greater than 2. In this case, we can easily plot this possibility distribution, at least for a few degrees of possibility. Given a degree of possibility γ, we plot the bounds of the largest interval $I_\gamma = [\underline{I}_\gamma, \overline{I}_\gamma]$ whose possibility is greater than γ. It is the "γ-cut" defined in Definition 12, which we then called "α-cut". The possibility measure is computed as $\Pi_{\text{SAD}}(A) = \sup_{z \in A} \pi_{\text{SAD}}(z)$ or $\Pi_{\text{SAD}}(A) = \sum_{a \cap A \neq \emptyset} m_C^{\text{SAD}}(a)$ using Equation (2.12). This way, given $\gamma \in [0, 1]$, the bounds $\underline{I}_\gamma, \overline{I}_\gamma$ to plot are:

$$\underline{I}_\gamma = \arg\min_z \{\pi_{\text{SAD}}(z) \geqslant \gamma\}, \qquad \overline{I}_\gamma = \arg\max_z \{\pi_{\text{SAD}}(z) \geqslant \gamma\} \tag{4.19}$$

Because focal sets of π_{SAD} are increasing intervals, then every SAD value between $\underline{I}_\gamma$ and $\overline{I}_\gamma$ will have a possibility superior to γ. The bounds define the envelope of values with a degree of possibility greater than γ. Plotting envelopes for different values of γ allows visualizing a representation of the SAD uncertainty. We will define the envelopes in the following, but readers can already have a broad idea of what those envelopes look like by looking at Figure 4.6.

In Equation (4.19), $\underline{I}_\gamma$ and $\overline{I}_\gamma$ are the upper and lower bounds of the same focal set. If we relax this constraint, then we can extend this definition of $\underline{I}_\gamma, \overline{I}_\gamma$ to any type of SAD plausibility (or equivalently belief) function Pl_{SAD}. For all focal sets $a = [\underline{a}, \overline{a}]$ of the SAD plausibility function, the bounds $\underline{I}_\gamma, \overline{I}_\gamma$ are defined as:

$$\underline{I}_\gamma = \min\{\underline{a} \mid \text{Pl}_{\text{SAD}}([\underline{a}, \overline{a}]) \geqslant \gamma\}, \qquad \overline{I}_\gamma = \max\{\overline{a} \mid \text{Pl}_{\text{SAD}}([\underline{a}, \overline{a}]) \geqslant \gamma\} \tag{4.20}$$

Remark: Equation (4.20) details how we chose to represent a plausibility function (or equivalently a credal set) by some envelopes. One can wonder if we can reconstruct the plausibility function from the envelopes. It is straightforward to see that it is the case if and only if the plausibility function is a possibility measure, because it is then

fully determined by its values on singletons.

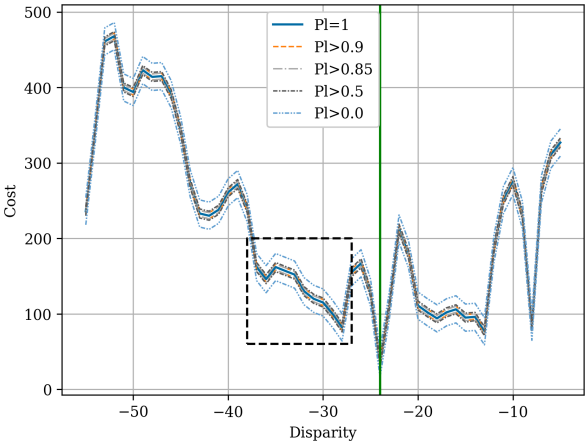
However, if the plausibility is not a possibility measure, then the possibility distribution π' defined by envelopes induces a credal set $\mathfrak{M}(\pi')$ containing $\mathfrak{M}(\mathrm{Pl}_{\mathrm{SAD}})$. In that regard, we chose to represent a plausibility by an outer approximating possibility.

Now that we decided on how to plot our uncertainty models, we can present some results. We arbitrarily considered different values for plausibility levels γ:

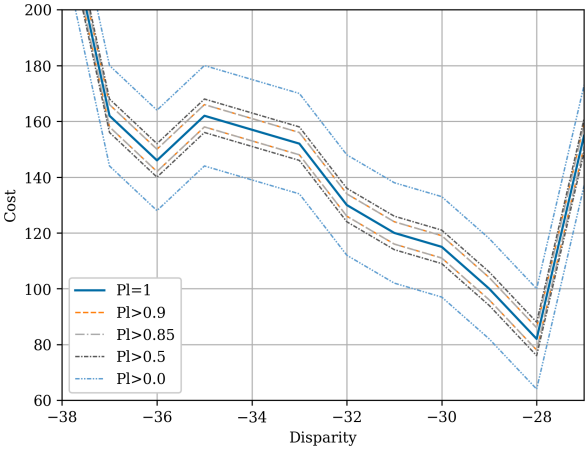
- The first value is $\gamma = 1$. It corresponds to the SAD value with the highest degree of possibility. This value is unique in our case, and corresponds to the SAD value that would have been computed without considering the uncertainty.
- We then consider values $\gamma = 0.9$ and $\gamma = 0.85$. They allow giving an estimation of the dispersion of envelopes with high plausibility. With them, we can estimate if highly plausible values are near the SAD values or not.
- $\gamma = 0.5$ gives a moderate plausibility estimation. From a credal set point of view, the probability that the SAD value lies in this set is at least 0.5.
- The last value is $\gamma = 0$. For this value, the inequalities in Equation (4.20) are actually strict (otherwise the inequality is verified by all imaginable values). These bounds represent the support of the plausibility function, or in other words, the range of values covered by focal sets.

A visualization of different plausibility levels of the SAD cost curve are displayed in Figure 4.6, computed with the product copula C_Π and a Gaussian copula C_R. The position of the considered pixel in the left image is displayed in Figure 4.7. Both copulas have the same support $\mathrm{Pl} > 0$, and the same value for $\mathrm{Pl} = 1$. Envelopes of the other plausibility levels are however different. Figure 4.6(b) and Figure 4.6(d) display the fact that the values covered by the plausibility levels vary with the copula used. Plausibility levels 0.85 and 0.5 are more concentrated around plausibility level 1 in the case of the product copula than in the case of the Gaussian copula. Conversely, plausibility level 0.9 is closer to plausibility level 1 in the case of the Gaussian copula. This is due to the fact that the Gaussian copula C_R is more co-monotone than the product copula given the correlation matrix R described in Equation (4.6).

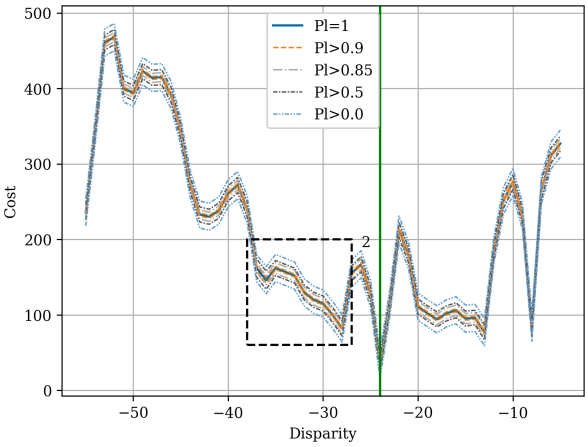
Remark: Both the Gaussian and the product copula have the same support. This is because when computing the joint mass of marginal focal sets, the copula is regular enough to assign a non-null mass to every focal set. This would not have been the case if we took a copula close to the lower or upper Fréchet-Hoeffding bound (the



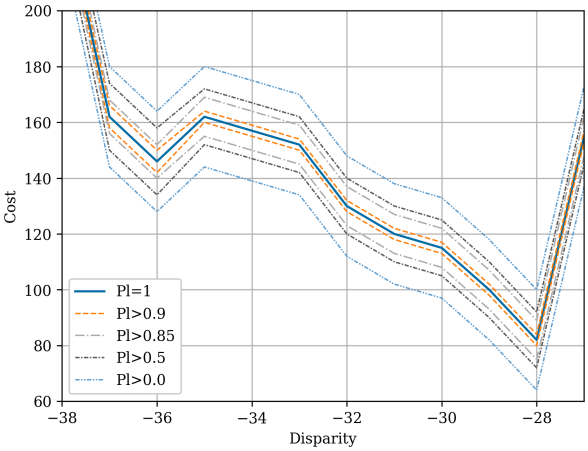
(a) SAD envelopes using the product Copula



(b) Detailed view of the rectangular section of (a)



(c) SAD envelopes using the Gaussian Copula



(d) Detailed view of the rectangular section of (c)

Figure 4.6: Plausibility levels of a cost curve for the product copula C_Π and the Gaussian copula C_R, for a pixel at coordinates $(100, 120)$. The green vertical line represents the true disparity. Rectangular sections from the left figures are detailed on the right. From [Malinowski et al., 2024a].

lower bound is not a copula for $n > 2$). Indeed, those copulas are less regular, and can assign a null mass to joint events, as we saw in the case of probabilities in Example 8 and in Example 15.

4.3.2 Estimating Propagated Credal Sets Using Monte Carlo Sampling

In Chapter 3, we defined 3 methods for creating multivariate credal sets: $\mathfrak{M}_{robust}$, $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$. In the case where marginals were possibility distributions, we saw that the bounds of $\mathfrak{M}_{agg}$ and $\mathfrak{M}_{mass}$ were the same on Cartesian products, so we only considered $\mathfrak{M}_{mass}$ in our application. In the previous sections, we propagated the uncertainty by using the approach from $\mathfrak{M}_{mass}$. In this section, we will aim to estimate the propagated uncertainty using $\mathfrak{M}_{robust}$, and compare it to previous results to evaluate whether we can use $\mathfrak{M}_{mass}$ to outer/inner approximate $\mathfrak{M}_{robust}$.

We remind here some definitions and results regarding $\mathfrak{M}_{robust}$. $\mathfrak{M}_{robust}$ is the convex hull CH of the set of every CDF from n marginal credal sets $\mathfrak{M}_i$ (in our case $n = 18$, one per pixel involved in the SAD computation) joined with a copula C (Definition 18):

$$\mathfrak{M}_{robust} = CH(\{F = C(F_1,\ \ldots,\ F_n),\ F_i \in \mathfrak{M}_i\})$$

We can estimate $\mathfrak{M}_{robust}$ using Monte Carlo samplings: we first generate marginal probability distributions $F_1,\ \ldots,\ F_n$ belonging in their respective marginal credal sets [Troffaes, 2017], then sample from the joint CDF F. More specifically, we sample a marginal CDF F_i belonging to the credal set defined in Equation (4.2), for each of the 18 considered pixels. Sampling from credal sets is not random: we generate probability distributions in such a way that for each marginal event A, the probability range $[\mathrm{Nec}(A), \Pi(A)]$ is sampled uniformly. With this method, we get a coverage of each marginal credal set. We are also ensuring that lower and upper bounds on events are reached at least once, in order to include "extreme" distributions in our simulations. Once we have a CDF F_i from each marginal credal set, we can sample from the joint CDF using the method detailed in Section 2.4.2. This yields a noised version of both left and right images, where the noised dependency is modeled by the provided copula, and its distribution is coherent with the possibility distributions from Equation (4.2). We can then compute a noised version of the cost volume from those images. For each cost curve, this provides a Monte Carlo samples of the SAD cost curves.

We compute 10 000 samples CDFs from each marginal credal set to construct the multivariate CDF. Each multivariate CDF is then itself sampled 10 000 times. The Central Limit Theorem states that N Monte Carlo samples provide an approximation which has

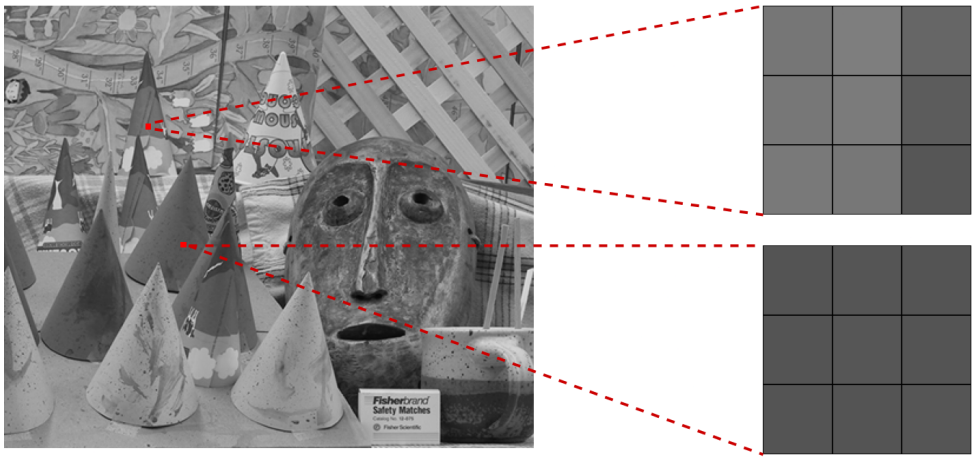


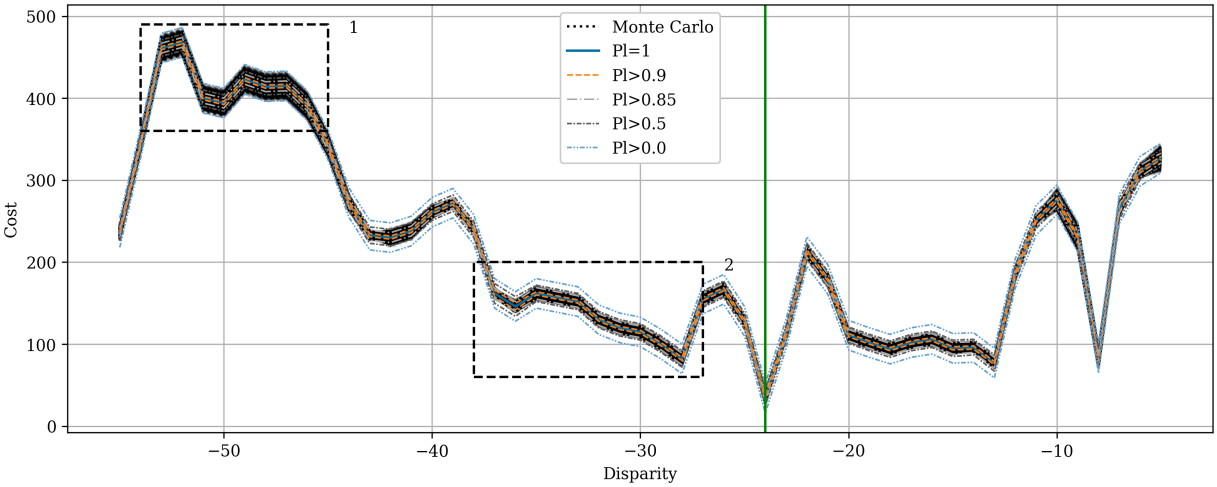
Figure 4.7: Position of the two 3×3 windows of Figures 4.8 and 4.9. Top right is $(100,\ 120)$ and bottom right is $(200, 150)$.

a precision of around $\frac{1}{\sqrt{n}}$ (it also depends on the standard deviation of the PDF you are estimating, but it is a good approximation), so $N = 10\,000$ provides a satisfying precision for our application.

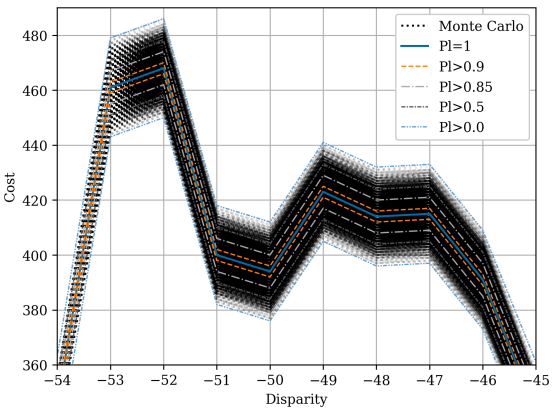
In practice, we do not simulate a full noised pair of images at once, as it would be required to sample from a copula of very large dimension (the number of pixels in both images). This is not realistically feasible, even though it would ensure that each time a pixel is considered in the cost volume, the same noise samples are used. We instead generate noise samples for each row separately: noised values of pixels will not change during the evaluation of a cost curve for different disparities, or between the cost curves of pixels of the same row. However, their values might change between cost curves of pixels belonging to different rows. For instance, let's consider a pixel $p = (row,\ col) \in I_L$ for which we computed a noised intensity i_p. We will use the same noised value i_p of intensity when computing the SAD of every pixel $q = (row,\ col') \in I_L$. But when computing the SAD of every pixel $q = (row - 1,\ col') \in I_L$, we will use a different Monte Carlo draw i'_p for its intensity, which will remain the same for every pixel of row $row - 1$. Because we draw 10^8 Monte Carlo draws for each row, proceeding as such should not be noticeable.

> **Remark:** As we construct the correlation matrices R based on the segmentation of the left and right images, the cost curves displayed in Figures 4.8 and 4.9 mostly use a different copula for each disparity. Providing the values of those matrices would necessitate to represent around 60 18×18 correlation matrices, and is thus not provided here.

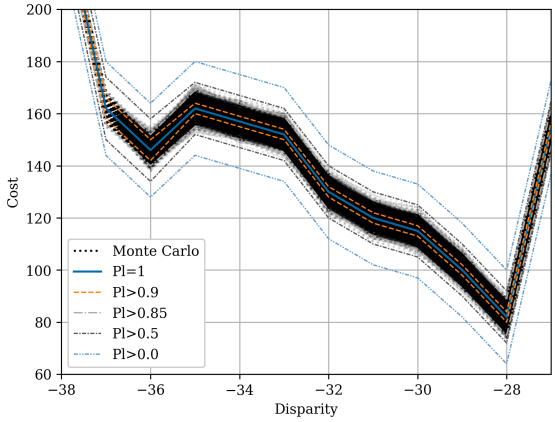
Monte Carlo draws using the Gaussian copula and marginals credal sets of Equation (4.2) are plotted in Figures 4.8 and 4.9. They respectively correspond to the cost



(a) Plausibility levels and Monte Carlo sampling using a Gaussian copula

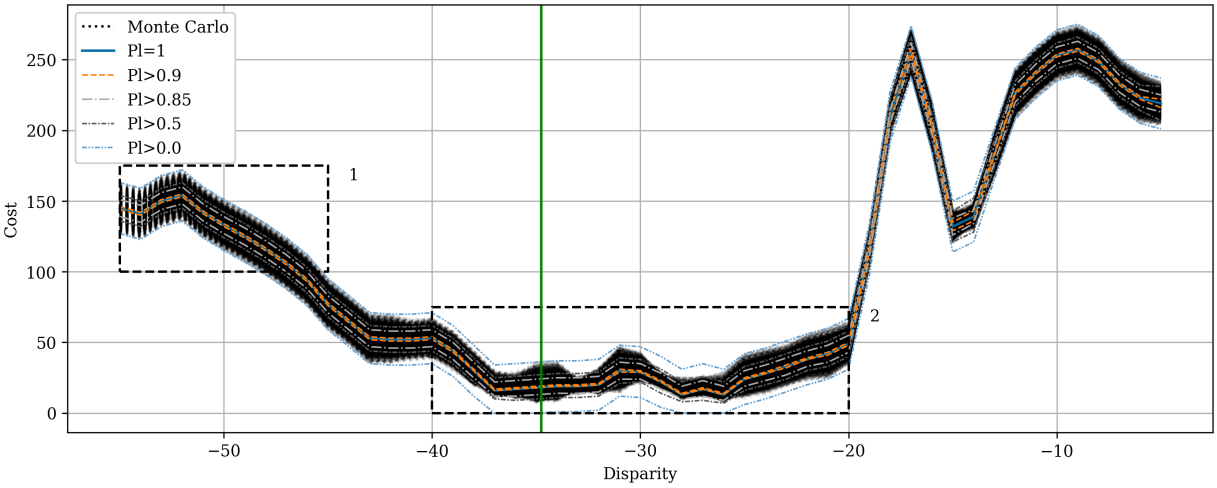


(b) Zoom over the first rectangle

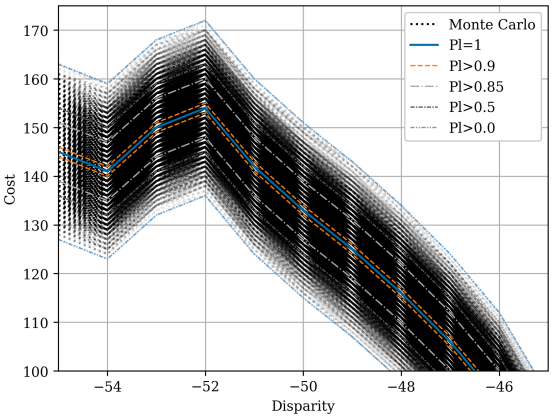


(c) Zoom over the second rectangle

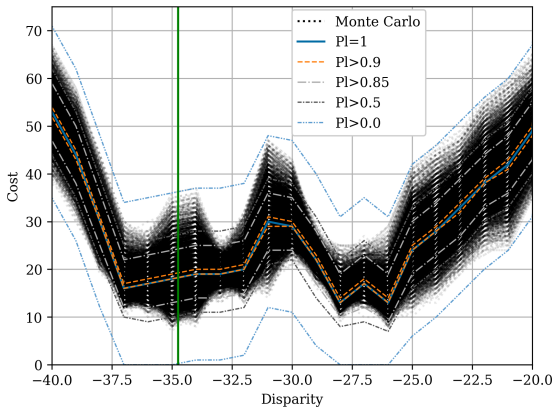
Figure 4.8: Plausibility levels and Monte Carlo sampling for a pixel at coordinates $(100, 120)$ in the left image. From [Malinowski et al., 2024a].



(a) Plausibility levels and Monte Carlo sampling using a Gaussian copula



(b) Zoom over the first rectangle



(c) Zoom over the second rectangle

Figure 4.9: Plausibility levels and Monte Carlo sampling for a pixel at coordinates $(200, 150)$ in the left image. From [Malinowski et al., 2024a].

curves of pixel located at positions (100, 120) and (200, 150) on the left image, as displayed in Figure 4.7. Plausibility levels computed using Equation (4.20) (displayed in Figure 4.6) also appear for comparison. The support envelopes from plausibility levels $\mathrm{Pl} > 0$ correctly contain all Monte Carlo samplings for all considered copulas. We can observe in Figure 4.9(c) that plausibility levels sometimes fail to correctly grasp the fluctuations of the dispersion of the samples, even though they correctly contain Monte Carlo samples. More specifically, Monte Carlo draws are first dense around disparity -37, then seem to spread around -35, and finally regather around disparity -32. The plausibility envelopes are more regular in this disparity range. This illustrates the fact that the "true" point-wise credal $\mathfrak{M}_{robust}$ set described in Section 3.1.1 is different from the joint credal set $\mathfrak{M}_{mass}$ from Section 3.1.2.

Although some differences persist between those sets, Figures 4.8(c) and 4.9(c) suggest that the point-wise credal set $\mathfrak{M}_{robust}$ can be outer approximated by $\mathfrak{M}_{mass}$ in our applications. To quantify this observation, we can compute the proportion of Monte Carlo samples contained inside the plausibility envelopes for different plausibility levels γ:

$$\text{coverage}_\gamma = \frac{\#\{ \text{ Monte Carlo samples } \in [\underline{I}_\gamma,\ \overline{I}_\gamma]\}}{\#\{ \text{ Monte Carlo samples } \}} \tag{4.21}$$

The coverage for the considered values of γ is presented in Table 4.1, where the first two rows represent the coverage of Figures 4.8 and 4.9. The global coverage over the whole left image is presented in the last row of the table. The coverage is always 100% for $\gamma = 0$, which shows that every sample is contained inside the support envelopes, and that $\mathfrak{M}_{robust}$ seems to be a subset of $\mathfrak{M}_{mass}$. On the other hand, plausibility level $\gamma = 0.5$ contain most Monte Carlo samples, which means that the bounds could be significantly reduced while still being a good estimation of $\mathfrak{M}_{robust}$. Finally, the variation of the coverage for plausibility levels 0.85 and 0.9 translates the previous observation that $\mathfrak{M}_{mass}$ does not capture the variations of $\mathfrak{M}_{robust}$ bounds and that those two set can substantially differ. After estimating the uncertainty of the cost volume, we will see in the following section how we can leverage the upper and lower bounds for different plausibility levels to improve the disparity map derived from the cost volume.

$p = (row, col)$	$\gamma = 0.9$	$\gamma = 0.85$	$\gamma = 0.5$	$\gamma = 0$
(100, 120)	$64,5\%$	$94,5\%$	$99,0\%$	100%
(200, 150)	$30,0\%$	$82,6\%$	$95,2\%$	100%
Global	$41,1\%$	$87,6\%$	$96,8\%$	100%

Table 4.1: Average coverage for various plausibility levels γ and for different pixels p of the left image.

4.3.3 Leveraging Confidence Envelopes for Potential Improvements

Knowing the uncertainty, represented as confidence envelopes in our case, can provide valuable insights into potential matches. This section outlines different observations suggesting that incorporating this information can enhance the performance of stereo matching algorithms.

From the cost volume C_V, we usually apply a *winner-takes-all* strategy to compute a disparity prediction. Given a pixel $(row,\ col)$, the predicted disparity $\tilde{d}$ is defined as:

$$\tilde{d}(row,\ col) = \arg\min_{d} C_V(row,\ col,\ d) \tag{4.22}$$

A common metric for evaluating stereo algorithm performance is the proportion of pixels for which the absolute difference between the true disparity d_{true} and the predicted disparity $\tilde{d}$ is less than one pixel. The score s is defined as:

$$s = \frac{\#\{(row,\ col) \text{ such that } |d_{\text{true}}(row,\ col) - \tilde{d}(row,\ col)| < 1\}}{\#\{(row,\ col)\}}\,. \tag{4.23}$$

The ground truth disparity can be any real number in the disparity range, but as we only consider integer disparities, we consider a disparity to be "correct" if it is less than one pixel away from the true disparity.

Having computed envelopes $\underline{I}_\gamma$, $\overline{I}_\gamma$ on the cost volume for different plausibility levels γ, the cost curve is not unique anymore, and we can instead consider all cost curves $C_V^{\underline{I}_\gamma, \overline{I}_\gamma}$ contained within the plausibility envelopes. Instead of a single predicted disparity, we can now compute the set of all potential disparities, defined as the set of predicted disparities derived from every cost curve within the plausibility envelopes. Given a pixel $(row,\ col)$, a confidence level $\gamma \in [0, 1]$, and plausibility envelopes $\underline{I}_\gamma(row,\ col,\ d)$, $\overline{I}_\gamma(row,\ col,\ d)$, the set of potential disparities $D_\gamma^{row,\ col}$ is defined as:

$$\begin{aligned} D_\gamma^{row,\ col} = \{d \ \mid\ & d = \arg\min_{\delta} C_V^{\underline{I}_\gamma, \overline{I}_\gamma}(row,\ col,\ \delta), \\ & \forall\ C_V^{\underline{I}_\gamma, \overline{I}_\gamma}(row,\ col,\ \delta) \in [\underline{I}_\gamma(row,\ col,\ \delta),\ \overline{I}_\gamma(row,\ col,\ \delta)]\} \end{aligned} \tag{4.24}$$

There is actually a simpler way of defining and computing D_γ, which is to notice that a disparity can be the minimum of a cost curve $C_V^{\underline{I}_\gamma, \overline{I}_\gamma}$ if and only if the lower bound $\underline{I}_\gamma$ for this disparity is less than the minimal value of the upper envelope $\overline{I}_\gamma$:

$$D_\gamma^{row,\ col} = \{d \mid \underline{I}_\gamma(row,\ col,\ d) \leq \min_{\delta} \overline{I}_\gamma(row,\ col, \delta)\} \tag{4.25}$$

Figure 4.10 provides a schematic example of $D_\gamma(row,\ col)$.

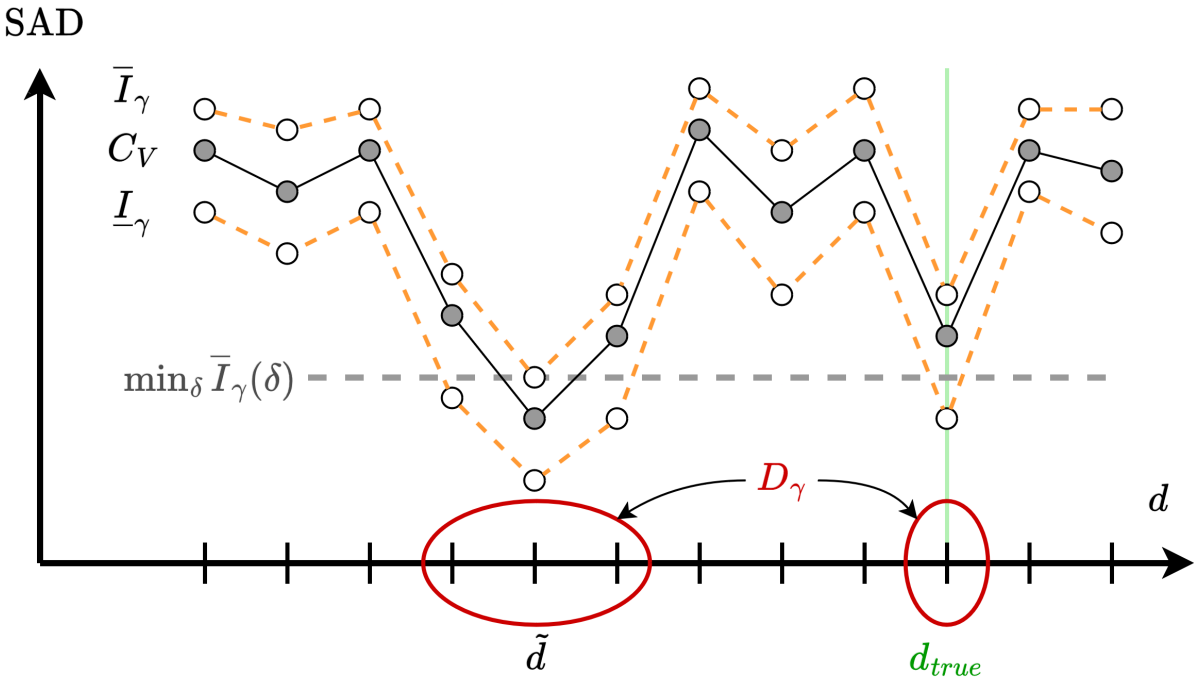


Figure 4.10: Example of a set of potential disparities D_γ. The minimum of the upper envelope $\min \overline{I}_\gamma$ in dashed gray line. The vertical green line represents the true disparity.

When the cost volume is considered without its uncertainty, many disparities are not correctly estimated by Equation (4.22). We thus want to quantify the potential score improvements contained in the set of potential disparities D_γ. To do so, we consider that D_γ is a potential improvement if it contains the true disparity d_{true}. From this we can compute the proportion of potential improvements s_γ^{opt} over the whole image which is computed as follows:

$$s_\gamma^{\text{opt}} = \frac{\#\{(row,\ col) \mid \min_{d \in D_\gamma^{row,\ col}} |d_{\text{true}}(row,\ col) - d| < 1\}}{\#\{(row,\ col)\}} \tag{4.26}$$

In Figure 4.10, we can see that the predicted disparity $\tilde{d}$ is far away from the true disparity d_{true}, but the set of possible disparity D_γ does indeed contain, the true disparity. This would have been counted as a potential improvement.

> **Remark:** Equation (4.26) defines the optimal score that *could* have been obtained if we used an ideal cost volume contained in the envelopes. We do not provide a method for determining this ideal cost volume. In reality, even with a good strategy to leverage uncertainty information to obtain a better disparity map, the new score s from Equation (4.23) would be lower than s_γ^{opt}.

The optimal score s_γ^{opt} can then be compared to the score s computed without uncertainty from Equation (4.23). We therefore define the potential gain as $\Delta s_\gamma = s_\gamma^{\text{opt}} - s$. Δs_γ measures the proportion of pixel that benefit from the method, *i.e.* pixels $(row,\ col)$ verifying:

$$|d_{\text{true}}(row,\ col) - \tilde{d}(row,\ col)| \geq 1 \quad \text{and} \quad \min_{d \in D_\gamma^{row,\ col}} |d_{\text{true}}(row,\ col) - d| < 1 \quad (4.27)$$

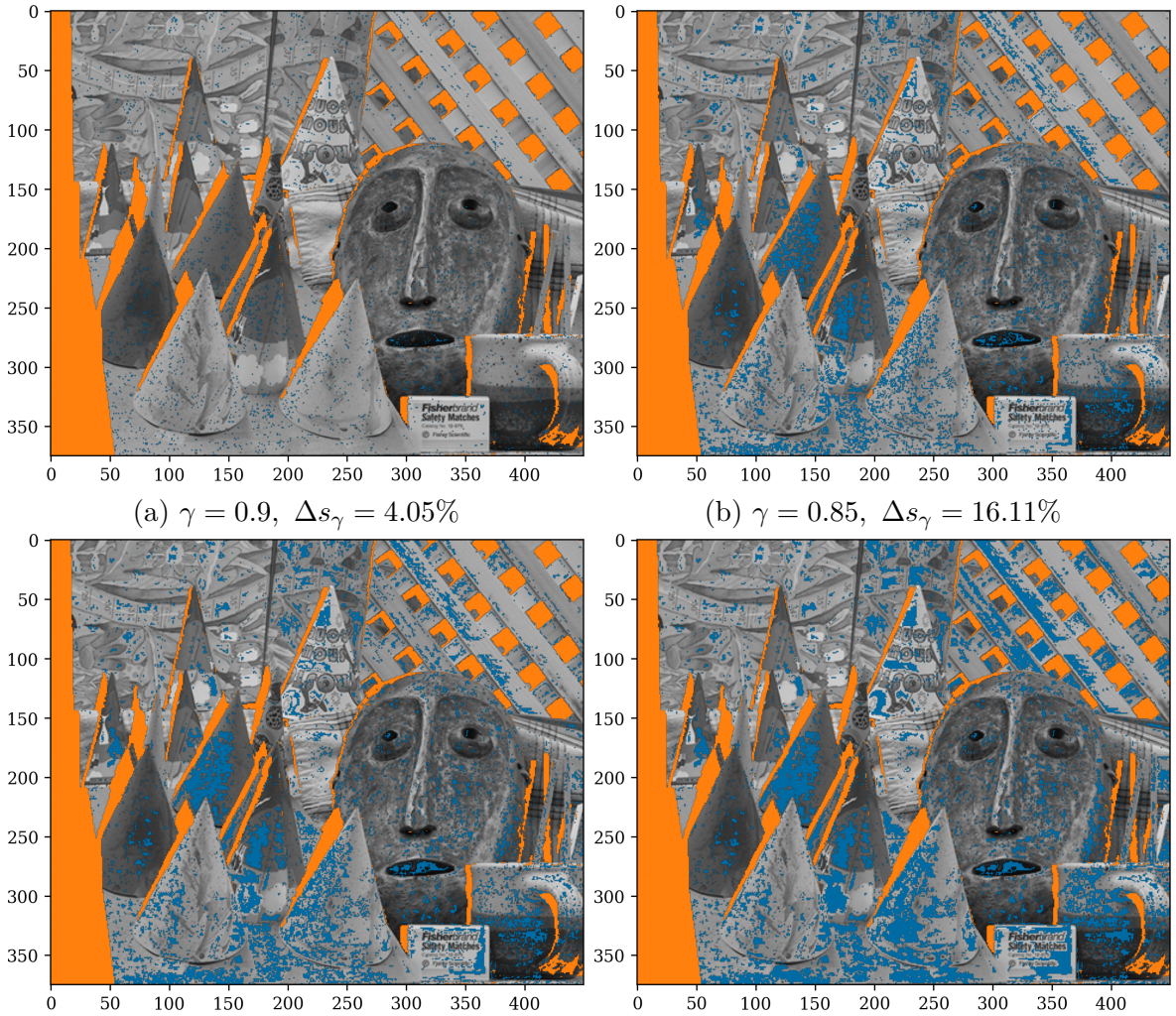
Instances of computed optimal scores and potential gains for various γ values are provided in Table 4.2. We can see that while the potential gain for $\gamma = 0.9$ is low, it increases significantly for lower values of γ. The base score is around 53%, which is a relatively low score for dense matching, but that was expected as we are using the SAD without SGM regularization or additional post-processing. With $\gamma = 0.85$, we could reach an optimal score of 67%, and using $\gamma = 0.5$ or $\gamma = 0$, the score could at best lie between 74% and 82%. This is a quite significant improvement for a cost volume computed with the SAD cost function. For comparison, a SAD cost volume regularized with SGM algorithm would have a score s of 68%.

Figure 4.11 displays the spatial distribution of pixels that can benefit from this method. Pixels in occluded regions, *i.e.* pixels present in one of the two images, are highlighted in orange. As occluded pixels cannot be improved as no true disparity exists, we do not consider them when computing the different statistics introduced in this section. Pixels where potential improvement can occur appear in blue.

We can see that pixels with potential improvements are not randomly distributed. These pixels are typically found in homogeneous areas where multiple disparities have low matching costs, similarly to the one represented in Figure 4.9(a).

$s = 52.87\%$	$\gamma = 0.9$	$\gamma = 0.85$	$\gamma = 0.5$	$\gamma = 0$
s_γ^{opt}	56.92%	66.99%	74.28%	81.75%
Δs_γ	4.05%	16.11%	21.41%	28.87%

Table 4.2: Optimal score and potential gain for different plausibility γ. The potential gain is computed with regard to the usual score $s = 52,87\%$.



(a) $\gamma = 0.9,\ \Delta s_\gamma = 4.05\%$ (b) $\gamma = 0.85,\ \Delta s_\gamma = 16.11\%$

(c) $\gamma = 0.5,\ \Delta s_\gamma = 21.41\%$ (d) $\gamma = 0,\ \Delta s_\gamma = 28.87\%$

Figure 4.11: Spatial disposition of potential improvements for different values of γ. Pixels with potential improvements appear in blue. Occluded pixels, for which a correct disparity does not exist, appear in orange. Grayscale left image is displayed on the background. From [Malinowski et al., 2024a].

Conclusion

In this Chapter, we modeled the uncertainty on stereo images and propagated it until the cost volume. We compared different methods from Chapter 3 for propagating the uncertainty, and evaluated the potential improvements unlocked by this uncertainty estimation. However, the stereo algorithm considered for computing the cost volume was intentionally simple to reduce the complexity of the problem. For the CO3D mission, different cost functions (CENSUS, MC-CNN) and SGM regularization would instead be used. Furthermore, although the uncertainty of input images have an influence on the final uncertainty, the major part of the disparity map uncertainty comes from the processing uncertainty of the stereo algorithm itself. Quantifying and propagating this processing uncertainty will thus be the subject of Chapter 5.

Chapter 5

Computing Disparity Confidence Intervals

The previous chapter detailed the propagation of uncertainty in a stereo matching problem, using the SAD function to compute the cost volume. However, real cases of stereo matching usually do not use such simple cost functions, but rather more complex ones. In particular, the CARS pipeline that will process the CO3D data will use the CENSUS and MC-CNN cost functions [Zabih and Woodfill, 1994, Žbontar and LeCun, 2016] to compute the cost volume, followed by a SGM regularization. Those methods produce far better results and are thus favored for stereo pipelines [Hirschmuller and Scharstein, 2009]. Unfortunately, propagating the uncertainty to those cost functions and their regularization as we did in the previous chapter would currently be too complex and computationally heavy for a practical use. The same observation can be made for any deep learning method [Laga et al., 2022] as the data is processed through many consecutive layers. Furthermore, the previous chapter was restricted to the propagation of the uncertainty from input images to the cost volume, but did not attempt to quantify the uncertainty of the stereo matching process itself, *i.e.* the algorithm's ability to identify the correct disparity. A bad performing stereo matching algorithm can produce errors of great magnitude regardless of the uncertainty on input images, while a more advanced algorithm may perform well despite noisy input images. Using the semantics of this thesis, we will refer to the uncertainty of the stereo algorithm itself as its epistemic uncertainty. Indeed, it does not result from any aleatoric process, but rather a lack of knowledge on how to automatically identify the correct disparity.

This epistemic uncertainty in stereo matching has been the subject of many studies in the literature [Hu and Mordohai, 2012, Poggi et al., 2021, Wang et al., 2022], designed for so-called "classical methods" using cost volumes obtained from cost functions, or for

learning-based methods. This uncertainty is quantified using "confidence measures", associating a value between 0 and 1 to each predicted disparity; 0 meaning that the prediction should be questioned, and 1 meaning that the prediction is most certainly correct. We refer to Section 1.4.3 for more details on confidence measures in stereo matching pipelines. In this chapter, we will study how possibility distributions are able to model the epistemic uncertainty associated with a cost volume, and then deduce disparity confidence intervals from the possibility distribution. This approach is complementary to classical confidence estimations, as it is not meant to indicate whether we trust a prediction, but rather to provide information on *where* the correct disparity is likely to be. We then propagate those disparity confidence intervals in the rest of the stereo pipeline to obtain elevation confidence intervals, and evaluate their performance. Following discussions with users and experts working at CNES, IGN and more generally in the AI4GEO consortium (`https://www.ai4geo.eu/`), we decided to aim for an objective of 90% of correct confidence intervals. This chapter takes up work and data published in [Malinowski et al., 2024c].

5.1 Producing Confidence Intervals

This section will detail the method developed to create disparity confidence intervals in the stereo matching step of the pipeline. We will use possibility distributions to model the uncertainty associated with the available information and deduce confidence intervals from it.

It is important to keep in mind that we want intervals as accurate as possible, while staying relatively small. Indeed, it would be easy to reach a 100% accuracy by simply extending the intervals to the whole range of considered disparities. However, the intervals would not contain any relevant information. We therefore must maintain a trade-off between the accuracy of the intervals and their size. As stated previously, we fixed ourselves a 90% accuracy objective. As for the criteria of maintaining small intervals, we will introduce in Section 5.2.1 different metrics to quantify their size for further evaluation. Different parameters will be introduced in our method, and we chose their values accordingly with the accuracy/size trade-off mentioned above. We will also study different configurations of those parameters in the Annex.

5.1.1 Possibility Distributions as Uncertain Models for Cost Curves

In this section, we will detail how possibility distributions can be used to model the epistemic uncertainty associated with cost curves. We first present a quick reminder

of concepts and notations regarding cost volumes presented in Section 1.3.3, as we will base our model on them. Cost volume based approaches, considered here, compare every pixel from the left image I_L to pixels from the same row in the right image I_R, in a given disparity range. The comparison is done using a cost function f, measuring the dissimilarity between two windows centered around pixels p and q. All evaluations using this cost function are stored in a cost volume C_V:

$$C_V(row,\ col,\ d) = f(I_L(row,\ col), I_R(row,\ col + d)) \tag{5.1}$$

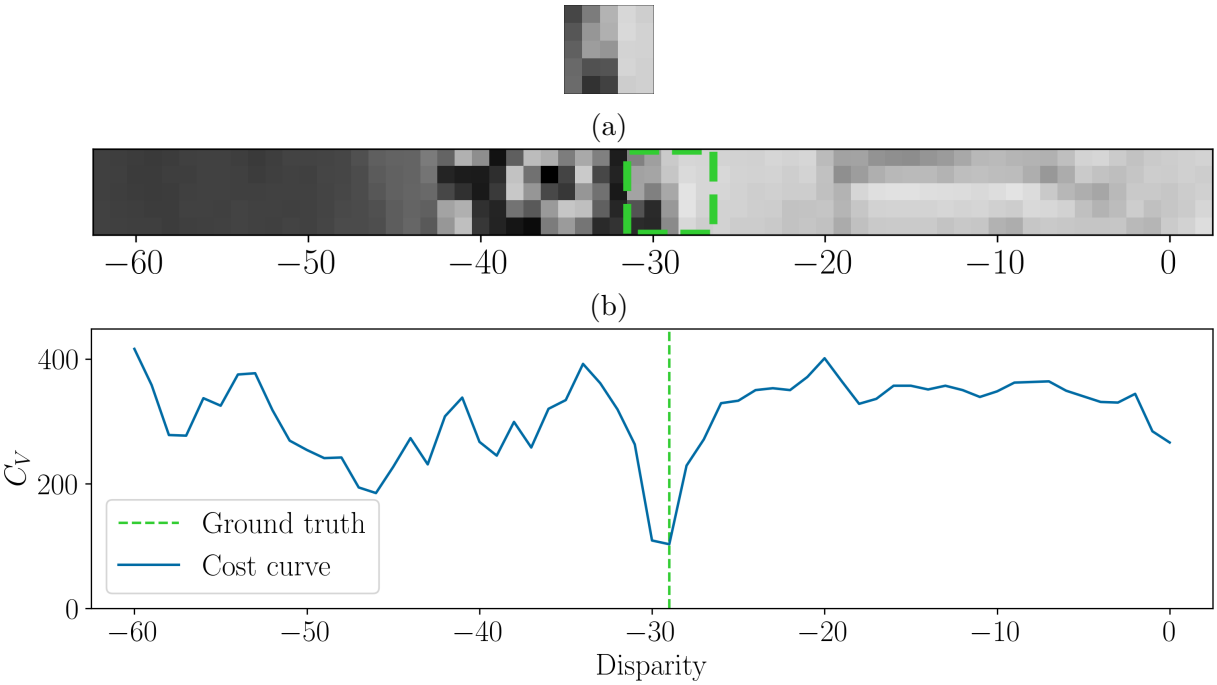
where d is the considered disparity. In this chapter, we consider that the cost volume undergoes a SGM regularization step, which modifies its values to take into account more global information as presented in Section 1.3.3. Based on the observation that the disparity map is usually piece-wise regular in a scene, SGM regularization has been designed to increase the cost of disparities for which no consensus exists among neighboring pixels. This way, only disparities that seem plausible and relatively regular compared to neighboring disparities are favored in the cost volume.

For every pixel $(row,\ col)$ from the left image, we refer to its cost curve as the cost volume at coordinates $(row,\ col)$ for every considered disparity d, *i.e.* the cost volume for which we fixed the first two variables. Cost curves are of great importance, as we estimate the disparity of a pixel solely based on its cost curve. Indeed, we define the predicted disparity $\tilde{d}$ of a pixel $(row,\ col)$ as:

$$\tilde{d} = \arg\min_{d} C_V(row,\ col,\ d) \tag{5.2}$$

In the next figures, we use the same configuration for computing the cost volume, *i.e.* CENSUS cost function with SGM regularization. Figure 5.1 presents a cost curve and its true disparity. We can see that the true disparity can be correctly estimated by looking at the minimum of the cost curve.

In the following, we propose to consider possibility distributions to model the uncertainty associated with the choice of the predicted disparity from a cost curve. The values taken by possibility distributions will be based on the available information, *i.e.* the values of the cost volume. Before getting into details, let us justify this model. Possibility distributions are relatively simple models to use in comparison with Imprecise Probabilities or belief functions for instance, as we only need to specify a constraint on atoms, and not on every event. As such, they have been used to model the uncertainty associated with an expert's opinion in applications such as groundwater contamination [Bárdossy et al., 1995], soil contamination and radioactive risk assessment [Baudrit and Dubois, 2005, Baudrit, 2005, Baudrit et al., 2007] or weather forecasting [Le Carrer and



(c) Cost curve

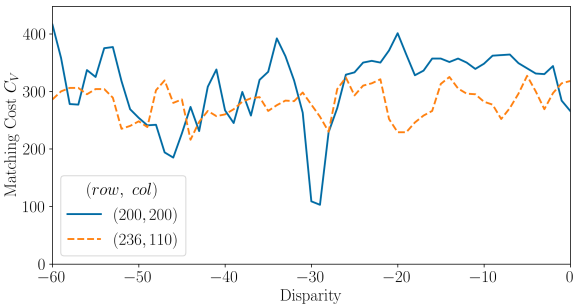
Figure 5.1: (c): cost curve obtained from comparing a patch of the left image from (a) to patches of the right image from (b). The matching patch and its corresponding true disparity are indicated using dashed green lines.

Ferson, 2021]. Since cost curves result in both:

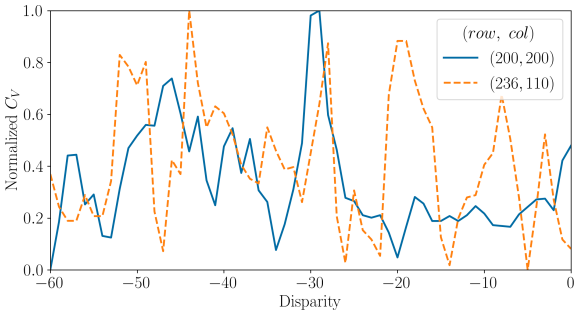
- dissimilarity measures between patches
- a semi-global fusion of the information contained in the cost volume due to SGM regularization

it does not seem far-stretched to consider them equivalent to an expert stating his opinion on how likely two pixels should be matched. For this reason, possibility distributions are appropriate to model the epistemic uncertainty of the cost volume.

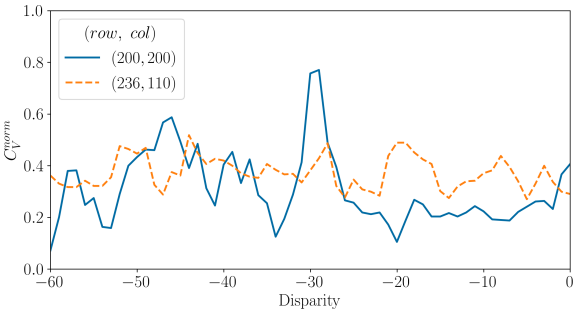
In order to use possibility distributions, we first need to transform the available information, in our case the values contained in the cost curves, into degrees of possibility. The definition of possibility distribution, Definition 11, imposes that the values must lie between 0 and 1, and that the value 1 must be attained at least once. We therefore propose to normalize each cost curve so that its minimal dissimilarity value equals a possibility degree of 1, and that greater dissimilarity values are closer to 0 in possibility. However, simply normalizing the values of each cost curve between 0 and 1 would artificially stretch the cost curve as seen in Figure 5.2(b). It is especially blatant in the case of the orange dashed curve from Figure 5.2(a) as the range of its values is quite narrow compared to the blue curve, but they are both stretched to $[0,\ 1]$ in Figure 5.2(b). In order to avoid



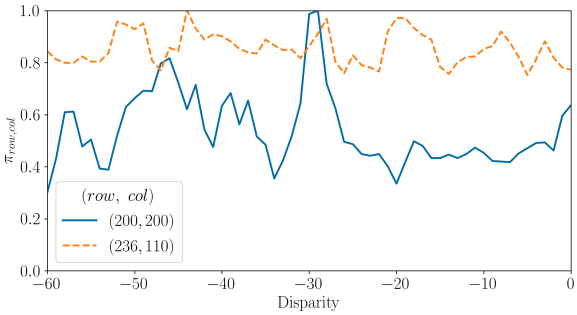
(a) Two cost curves



(b) Normalized cost curves with local extrema



(c) Normalized cost curves C_V^{norm} with global extrema



(d) Possibility distributions resulting from the cost curves

Figure 5.2: Transformation of cost curves (CENSUS + SGM on Middlebury Cones) into possibility distributions. (a) represents two cost curves that are normalized differently in (b) and (c). (d) uses the normalization of (c) to create possibility distributions.

this effect, we instead normalize every cost curve using the global minimum and global maximum of the cost volume, as:

$$C_V^{norm}(row,\ col,\ d) = \frac{C_V(row,\ col,\ d) - \max_{r,c,\delta} C_V(r,\ c,\ \delta)}{\min_{r,c,\delta} C_V(r,\ c,\ \delta) - \max_{r,c,\delta} C_V(r,\ c,\ \delta)} \tag{5.3}$$

Minima of the cost curve become maxima with this normalization. One problem remains, it is that unless the global maximum of the cost volume is attained in a cost curve, the normalized cost curve will never reach 1. Therefore, it will not be a possibility distribution. This problem can be observed in Figure 5.2(c). We thus add a constant to the normalized cost curve to obtain a possibility distribution $\pi_{row,\ col}(d)$:

$$\pi_{row,\ col}(d) = C_V^{norm}(row,\ col,\ d) + 1 - \max_{\delta} C_V^{norm}(row,\ col,\ \delta) \tag{5.4}$$

Figure 5.2(d) displays the possibility distributions obtained from the cost curves of Figure 5.2(a).

As stated previously, global extrema in Equation (5.3) are employed to minimize the stretching effect when converting cost curves into possibility distributions. Alternatively, we also could have used the theoretical extrema of a cost curve instead. For instance,

the CENSUS cost function on a 5×5 window provides values between 0 and $C_{max} = 24$. Adding SGM regularization with penalty P_2 on 8 directions yields cost volumes values between 0 and $8 \times (C_{max} + P_2)$ [Hirschmüller, 2005]. However, this maximal cost is rarely attained in real case scenarios. Therefore, using theoretical extrema of the cost volume is too pessimistic and tends to over-compress the normalized cost curves. It is instead preferred to use global extrema of the cost volume for the normalization, as we assume the best and worst match should have similar cost values across different scenes. This hypothesis is not restrictive for the images we consider in our stereo matching problem, as the size and diversity inside each scene lead to similar extrema. When processing very large images, the CARS stereo pipeline divides the image into small tiles that are processed in parallel. There is no guarantee that the cost volume extrema of each tile would be the same. In practice, cost volume extrema are similar for large images. We therefore assume differences in extrema are negligible.

5.1.2 From Possibilities to Disparity Confidence Intervals

With the possibility distributions defined, our next objective is to establish a set of most possible disparities. As presented previously, we decided to aim for sets containing the true disparity 90% of the time. To define this set of most possible disparities, we compute the α-cut from Definition 12, or, in other words, the set of all disparities D_α whose possibility is greater than α:

$$D_\alpha = \{\ d \mid \pi_{row,\ col}(d) \geqslant \alpha\} \tag{5.5}$$

By looking at possibility distributions obtained from different cost curves for which we know the true disparity, we first fixed the value α at 0.9. In depth study of this parameter will be conducted in Annex, in order to see if it depends on the cost function, the type of scene considered, and to provide general guidelines on its optimal value. In the following, when the value of α is not specified, it will always be set at 0.9. Figures 5.3(a) and 5.3(b) graphically represent D_α for the cost curves of Figure 5.2.

> **Remark:** The fact that its value is the same as the 90% confidence objective is a coincidence, and one should not assume that α and the confidence objective should be the same. Indeed, raising the α value would decrease the size of the set D_α and therefore decrease the proportion of sets containing the true disparity, *i.e.* the global confidence rate of intervals.

> **Remark:** We are modeling the epistemic uncertainty of the cost curves using possibility distributions. In the rest of this chapter, we use a possibility distribution for

each cost curve because we think it is a correct model in itself for the uncertainty we encounter. It is however possible to have a probabilistic interpretation of possibilities.

We saw in Equation (2.14) from Chapter 2 that one way of interpreting $\pi_{row,\ col}$ is that it defines a set of probability distributions, *i.e.* a credal set $\mathfrak{M}$. We can also define D_α using this set $\mathfrak{M}$, as:

$$D_\alpha = \{\ d \mid \exists P \in \mathfrak{M}\ s.t.\ P(d) \geqslant \alpha\} \tag{5.6}$$

or in plain words, D_α is the set of all disparities d for which there exists a probability in $\mathfrak{M}$ whose value is greater than α for d. We will not rely on this interpretation in the rest of this chapter, and instead only reason in terms of possibilities.

Equation (5.5) defines a set of disparities D_α that is not necessarily convex. We will rather consider disparity confidence intervals I_α deduced from D_α in the rest of this chapter:

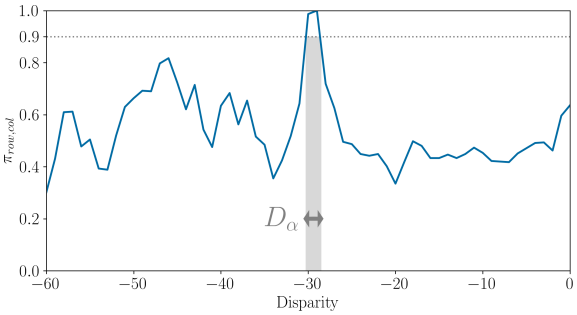
$$I_\alpha = [\underline{I}_\alpha,\ \overline{I}_\alpha] = [\min D_\alpha,\ \max D_\alpha] \tag{5.7}$$

Figures 5.3(c) and 5.3(d) graphically represent how I_α is determined for the two possibility distributions of the cost curves in Figure 5.2. In the rest of this section, we will not make any distinction between disparity confidence intervals, disparity intervals, confidence intervals or simply intervals. A disparity interval is the convex envelope of its disparity set D_α, which is a conservative approach as observed in Figures 5.3(b) and 5.3(d).

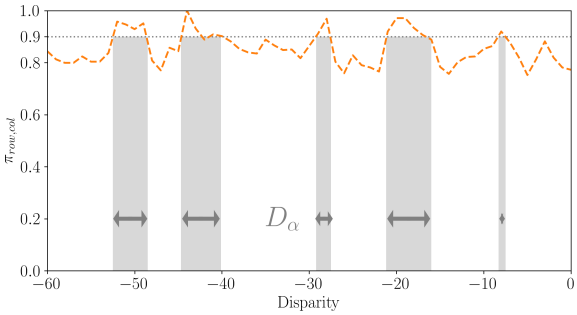
Considering intervals thus presents the advantage of working with convex sets and only requiring two scalars to describe the set. Users of DSMs produced by stereophotogrammetry are familiar with confidence intervals [Oksanen, 2006, Wang et al., 2015, Panagiotakis et al., 2018, Deschamps-Berger, 2021, Hugonnet et al., 2022]. It will also facilitate further processing in the rest of the stereo pipeline, as we only need to take into account 2 bounds to characterize possible disparities, instead of sets of arbitrary shape.

To qualitatively evaluate the behavior of disparity confidence intervals I_α, we will look at their values for consecutive pixels of the same row. Rows selected for this analysis are presented in Figure 5.4: the upper row (80) is smaller than the others, so more details can be observed, while other rows allows for a broader view of the intervals. We selected rows containing different disparity configurations in different part of the image, as adjacent rows tend to look relatively similar.

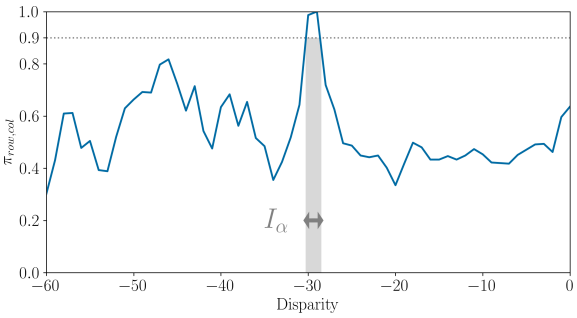
By construction, I_α will always contain the predicted disparity $\tilde{d}$ as the maximum of each possibility curve will also be selected as the predicted disparity by winner-takes-all



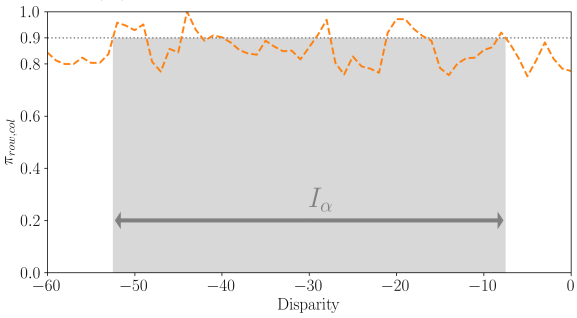
(a) D_α for the blue possibility of Figure 5.2(d)



(b) D_α for the orange possibility of Figure 5.2(d)



(c) I_α for the blue possibility of Figure 5.2(d)



(d) I_α for the orange possibility of Figure 5.2(d)

Figure 5.3: Set of possible disparities D_α and disparity intervals I_α with the same cost curves as in Figure 5.2, with $\alpha = 0.9$. (a) and (b) represent the set of possible disparities D_α from Equation (5.5) in gray. (c) and (d) represent disparity intervals I_α from Equation (5.7) in gray. There is no difference between D_α and I_α for the blue curve, contrary to the orange dashed curve.

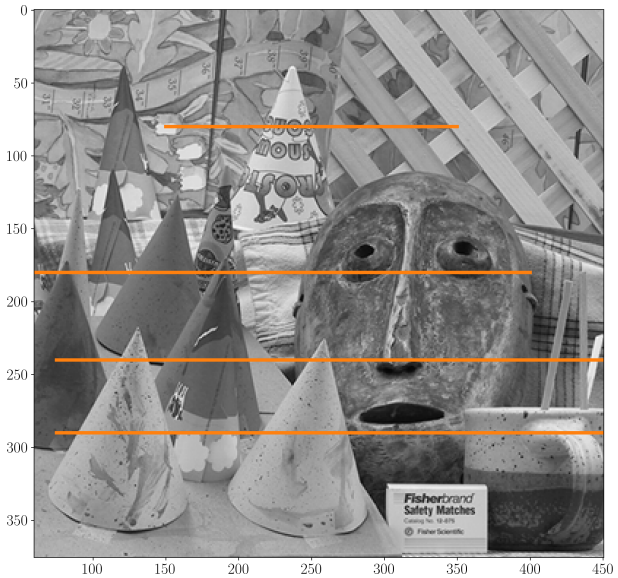
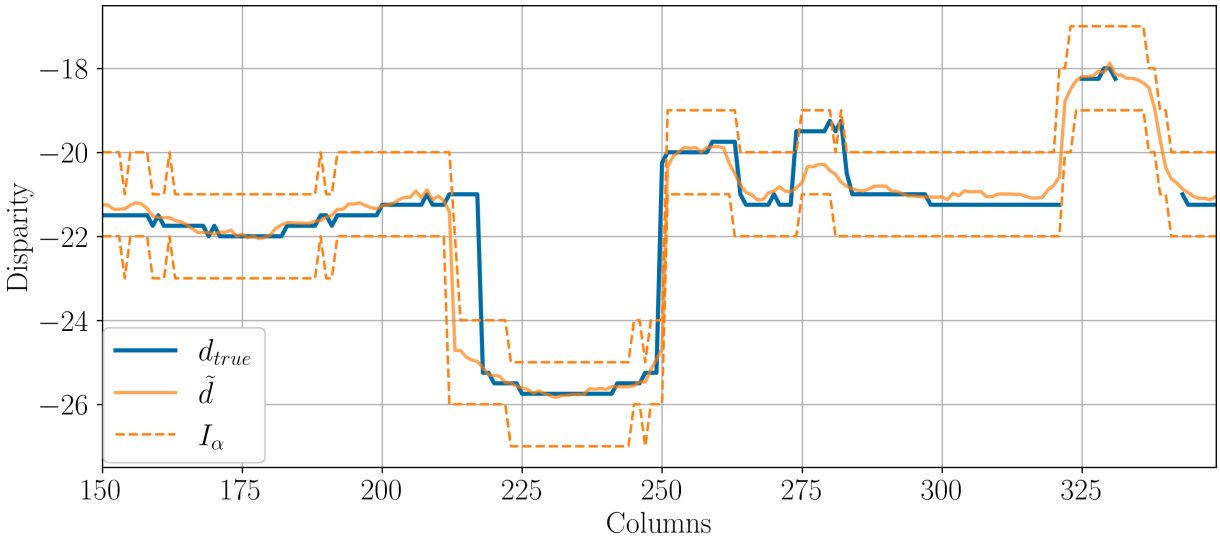
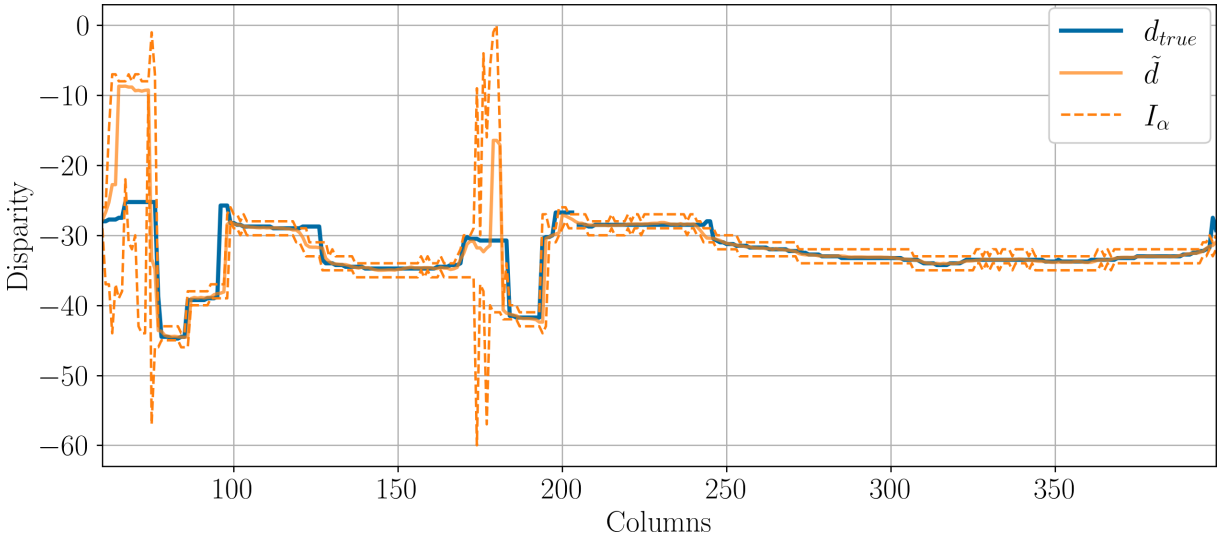


Figure 5.4: Left stereo image from Middlebury Cones. Disparity intervals I_α along orange lines are detailed in Figures 5.5, 5.6 and 5.11 to 5.14

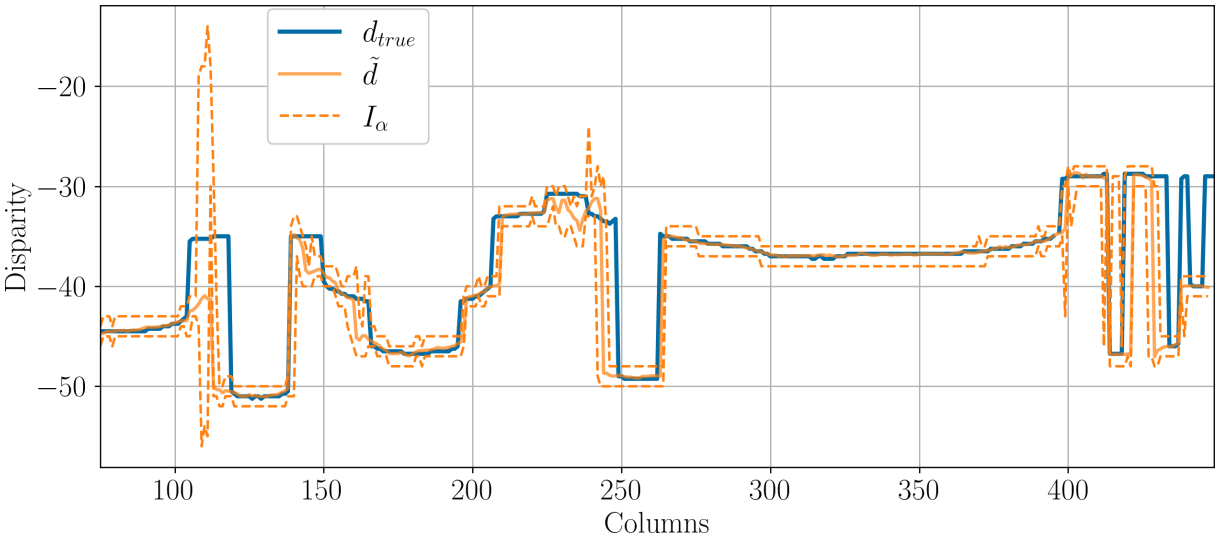


(a) I_α along row 80

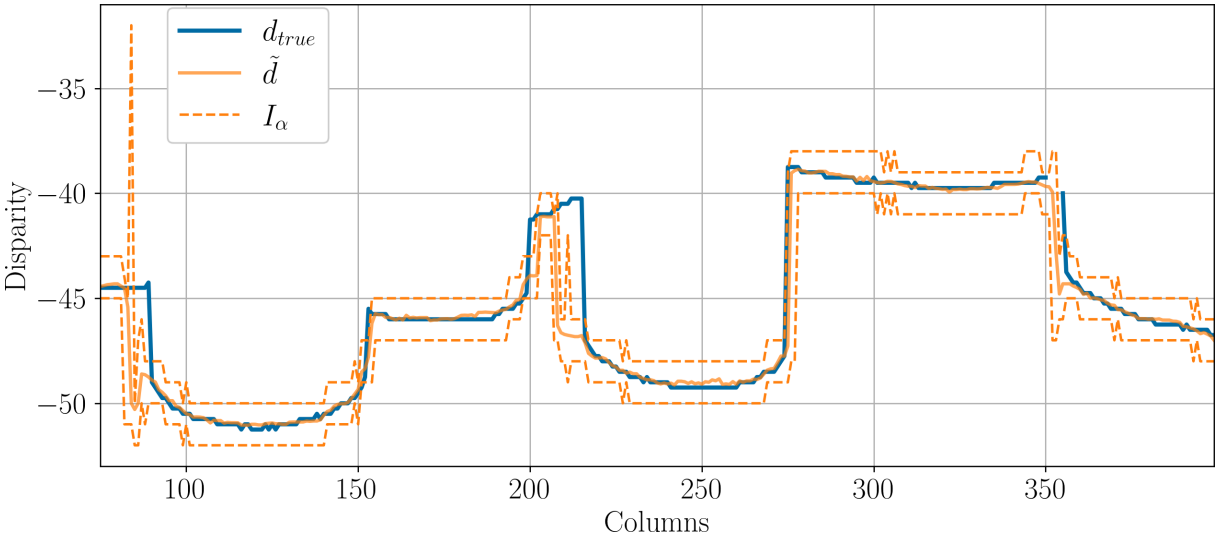


(b) I_α along row 180

Figure 5.5: I_α for the two top rows highlighted in Figure 5.4



(a) I_α along row 240



(b) I_α along row 290

Figure 5.6: I_α for the two bottom rows highlighted in Figure 5.4.

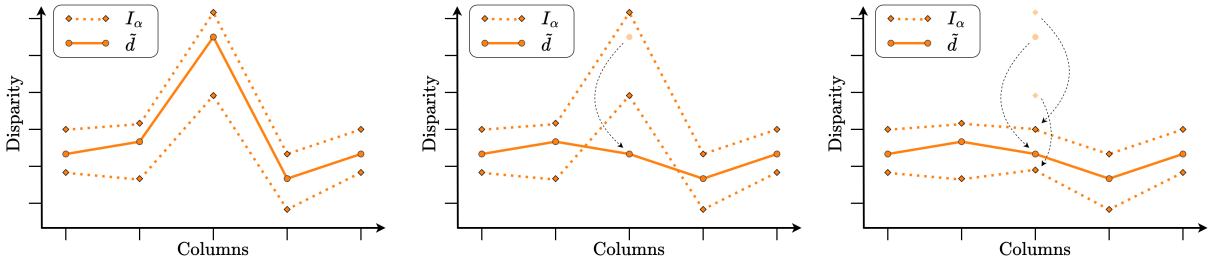
strategy. However, we are interested to see if intervals can contain the true disparity d_{true} even when the predicted disparity is far from it. Figures 5.5 and 5.6 represent the disparity intervals I_α, predicted disparity $\tilde{d}$ and true disparity d_{true} for the different rows. A first observation is that intervals correctly contain the true disparity in regions where there are no strong variations of disparities. In Figure 5.5(b), intervals in columns 50 to 75 and around 175 are much larger than in the rest of the figure. In those areas, the predicted disparity is also far from the ground truth. This translates the fact that the method for creating intervals is able to detect the difficulties encountered by the correlator when predicting a disparity, and to adapt the size of intervals consequently. On the downside, we can see that near strong variations of the disparity, intervals tend to "miss" the discontinuities. Indeed, they do not contain the true disparity around those areas, as observed near columns 215 and 250 of Figure 5.5(a), columns 95, 125 of Figure 5.5(b), columns 100, 115, 150, 200, 250 of Figure 5.6(a) and finally columns 90, 200 and 220 of Figure 5.6(b).

The method presented in this section seems to offer a good estimation of the error in the disparity estimation step. Some errors remain near disparity discontinuities, which we will try to rectify in Section 5.1.4.

5.1.3 Ensuring Coherence Between the Predicted Disparity and Confidence Intervals

We propose a method for creating confidence intervals that should include the true disparity at least 90% of the time. It should, however, always include the predicted disparity $\tilde{d}$. Indeed, it would not make much sense to provide a confidence interval and a prediction that is not included in the interval. As $\tilde{d}$ is the maximum of each possibility curve, it will also be selected as the predicted disparity by the winner-takes-all strategy and will thus belong to the confidence interval. However, the disparity map $\tilde{d}$ is often post-processed to improve its quality, mainly using a filtering and a refinement step. Those steps modify the disparity map, and we must ensure we modify the confidence intervals accordingly so that they remain coherent with the predicted disparity.

As detailed in Section 1.3.3, a filtering step is usually carried out on the disparity map in order to remove potential outliers. The filter applied in our experiments and in many other pipelines is a median filter [Scharstein et al., 2001]. Applying the filter only to the disparity map without processing the intervals accordingly can result in inconsistencies, as illustrated in Figure 5.7. Fortunately, separately applying the same median filter to the lower bounds and upper bounds of the intervals is sufficient to ensure coherence. Indeed,



(a) Without filtering (b) Filtering only $\tilde{d}$ (c) Filtering $\tilde{d}$ and I_α

Figure 5.7: Effect of a median filter on the predicted disparity $\tilde{d}$ and confidence intervals I_α. For the sake of the example, we only filter the middle point by looking at its neighbors. (a) contains the unfiltered curves. (b) contains unfiltered intervals, and filtered $\tilde{d}$. (c) filtered intervals and filtered $\tilde{d}$.

because for all pixels $p_1, \ldots, p_n$ considered in the filtering, it holds:

$$\underline{I}_\alpha(p_i) \leqslant \tilde{d}(p_i) \leqslant \overline{I}_\alpha(p_i) \tag{5.8}$$

then it is possible to prove that

$$\underset{p_1,\ \ldots,\ p_n}{\text{median}}\ \underline{I}_\alpha(p_i) \leqslant \underset{p_1,\ \ldots,\ p_n}{\text{median}}\ \tilde{d}(p_i) \leqslant \underset{p_1,\ \ldots,\ p_n}{\text{median}}\ \overline{I}_\alpha(p_i) \tag{5.9}$$

The proof of this result can be found in the Annex, using Proposition 14.

Another processing applied to the disparity map is the sub-pixel refinement of its values, presented in Section 1.3.3 and Figure 1.20. The idea is to slightly modify the value of the disparity by interpolation of the cost curve, in order to obtain sub-integers disparity values. The modification cannot change a disparity more than a pixel away from its original value. If the predicted disparity $\tilde{d}$ equals one of the bounds of its confidence interval I_α, the sub-pixel refinement step can shift the predicted disparity to a value slightly outside the confidence intervals. In this case, we simply extend the interval by one pixel. For instance, if $\tilde{d} = \underline{I}_\alpha$, then the new confidence interval I'_α equals:

$$I'_\alpha = [\underline{I}_\alpha - 1,\ \overline{I}_\alpha] \tag{5.10}$$

This stretching is simple, and also presents the advantage of working with any type of sub-pixel refinement that do not modify the predicted disparity value from more than 1 pixel (*i.e.* V-fit, parabola *etc.*).

5.1.4 Regularization of Intervals in Low Confidence Areas

Disparity intervals estimation has the potential to perform well even when the predicted disparity is far away from the ground truth. This model however encounters some per-

formance issues near depth discontinuities, and does not currently satisfy the aimed 90% accuracy objective (see Annex). This can be explained as follows: as the SGM regularization attempts to impose continuity on disparities, it results in cost curves that do not favor the correct disparity in the region where the continuity hypothesis is not valid. Considering that cost curves are equivalent to experts' opinions in those areas can be over-optimistic, and the resulting intervals thus cannot be completely trusted. We will now present how we can adapt the model in those regions, in order to correct those flaws.

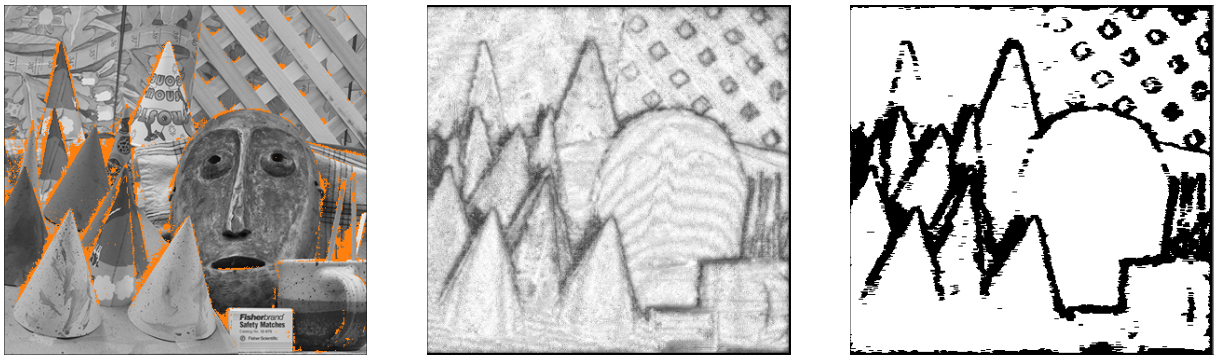
The first challenge to tackle is to determine if we are able to detect regions where discontinuities occur, in order to process intervals differently in those regions. Using the predicted disparity map might be a lead, but we saw in Figures 5.11 to 5.14 that there is usually a shift between predicted disparity discontinuities and true disparity discontinuities. Instead, we could consider to use confidence measures computed alongside the disparity map, which are usually good candidates to detect discontinuities. In particular, we chose the confidence from ambiguity measure c_{amb} presented in Equation (1.30) from Chapter 1. As a reminder, the confidence from ambiguity of a cost curve is computed as follows: given a value $\eta > 0$, we compute the number of disparities whose cost is within η to the minimum of the cost curve, then we compute the integral for all η and normalize it between 0 and 1. It is formally transcribed by the following equations from Section 1.4.3:

$$
\begin{aligned}
amb(row,\ col,\ \eta) &= \#\{d \mid C_V(row,\ col,\ d) \leqslant \min_{\delta} C_V(row,\ col,\ \delta) + \eta\} \\
\mathrm{AUC}_{amb}(row,\ col) &= \frac{1}{\max \eta - \min \eta} \int_{\eta} amb(row,\ col,\ \eta) d\eta \\
c_{amb}(row,\ col) &= \frac{\max \mathrm{AUC}_{amb} - \mathrm{AUC}_{amb}(row,\ col)}{\max \mathrm{AUC}_{amb} - \min \mathrm{AUC}_{amb}}
\end{aligned}
$$

Note that other confidence measures could be considered instead of the ambiguity, but this confidence measure has the advantage of performing well, being explainable (which is not always the case for confidence measures based on deep learning) and being already implemented in the stereo pipeline we use. Figure 5.8 shows that pixels with wrong intervals I_α usually present a low confidence from ambiguity as well, meaning that we can use this confidence measure to process confidence intervals differently.

The method developed for detecting low confidence areas is to apply a simple threshold τ_{amb} on the confidence from ambiguity c_{amb}. However, c_{amb} is computed pixel-wise, which can lead to high-frequency spatial variations of the confidence. To smooth the ambiguity curve, a minitive kernel of size $(1,\ 2 \times k_{amb} + 1)$ is applied. A pixel $(row,\ col)$ is thus considered to be in a low confidence area if it verifies:

$$
\min_{-k_{amb} \leqslant k \leqslant k_{amb}} c_{amb}(row,\ col + k) \leqslant \tau_{amb} \tag{5.11}
$$



(a) Left stereo image. Pixels where $d_{true} \notin I_\alpha$ appear in orange.

(b) Confidence from ambiguity c_{amb}. Dark pixels have a low confidence, bright pixels have a high confidence.

(c) Binary mask, low confidence areas are indicated by black pixels.

Figure 5.8: Position of wrong intervals in the left image, the corresponding confidence map and the low confidence area mask obtained using Equation (5.11)

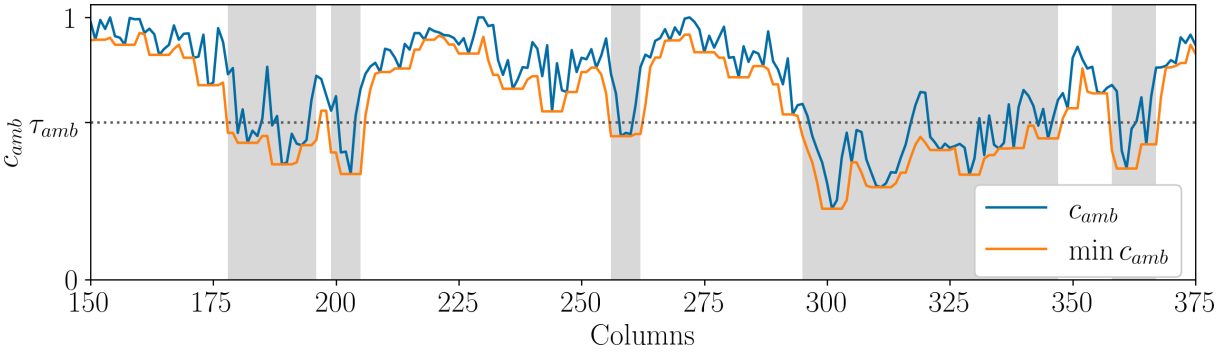


Figure 5.9: Confidence from ambiguity c_{amb} and smoothed confidence $\min c_{amb}$ from Equation (5.11) for row 110 of the Middlebury Cones image. Low confidence areas, where $\min c_{amb}$ is less than τ_{amb} are highlighted in gray.

We used $k_{amb} = 2$ and $\tau_{amb} = 0.6$ in our experiments. For simplicity, we will refer to $\min c_{amb}$ instead of $min_{-k_{amb} \leqslant k \leqslant k_{amb}} c_{amb}$ in the following. Additional investigations on those parameters are presented in the Annex. Figure 5.9 illustrates the impact of smoothing the confidence from ambiguity, and displays the threshold used to detect low confidence areas. We can see around columns 185 and 320 that the kernel smooths isolated confidence values that would not be detected by the threshold otherwise. Figure 5.8(c) presents the position of pixels in low confidence areas over the whole left image. As a quick indicator of this method performance, 83% of intervals that do not contain the ground truth (orange pixels in Figure 5.8(a)) are also contained in low confidence areas. Once again, it is possible to use any other confidence measure to detect areas where intervals perform badly. The choice of this method in particular is motivated by its good performance while remaining simple in comprehension and implementation.

Having computed regions of low confidence, we can now process the intervals differently in those areas. The main idea here is that information contained in low confidence

cost curves should be handled with care. The hypothesis that a cost curve can be interpreted as an expert stating his opinion on which disparities are most probable is questionable in those areas. We also cannot infer the confidence intervals from intervals in neighboring high confidence areas, as there is no guarantee the disparities are the same as their high confident neighbors. We instead chose to proceed in two steps. First, we compute a neighboring set of intervals for each low confidence pixel. Then, we use the information from this set to determine a new disparity interval by consensus. We modify the value of the considered pixel from this consensual interval.

We first detail how the set of neighboring pixels is determined. Let $(row,\ col)$ be a low confidence pixel. We define a segment $S(row,\ col)$ as the set containing $(row,\ col)$ and all adjacent low confidence pixels from the same row. An example of $S(row,\ col)$ is presented in Figure 5.10 as an orange rectangle. Two segments $S(row,\ col)$ and $S(row+1,\ col')$ are considered adjacent if two of their pixels are directly on top of one another. The low confidence neighboring $\mathfrak{N}(row,\ col)$ is defined as the set of low confidence pixels in segment $S(row,\ col)$ or in adjacent segments within $n_{\mathfrak{N}}$ rows. In practice, we use $n_{\mathfrak{N}} = 2$. $S(row,\ col)$ is formally defined as:

$$S(row,\ col) = \{\ (row,\ col')\ s.t.\ \forall c \in [\![col, col']\!],\ \min c_{amb}(row,\ c+k) \leqslant \tau_{amb}\ \} \tag{5.12}$$

where $[\![col,\ col']\!]$ assumes that $col \leqslant col'$, and is replaced by $[\![col',\ col]\!]$ if not. The formal definition of the set of pixels $\mathfrak{N}(row,\ col)$ is then:

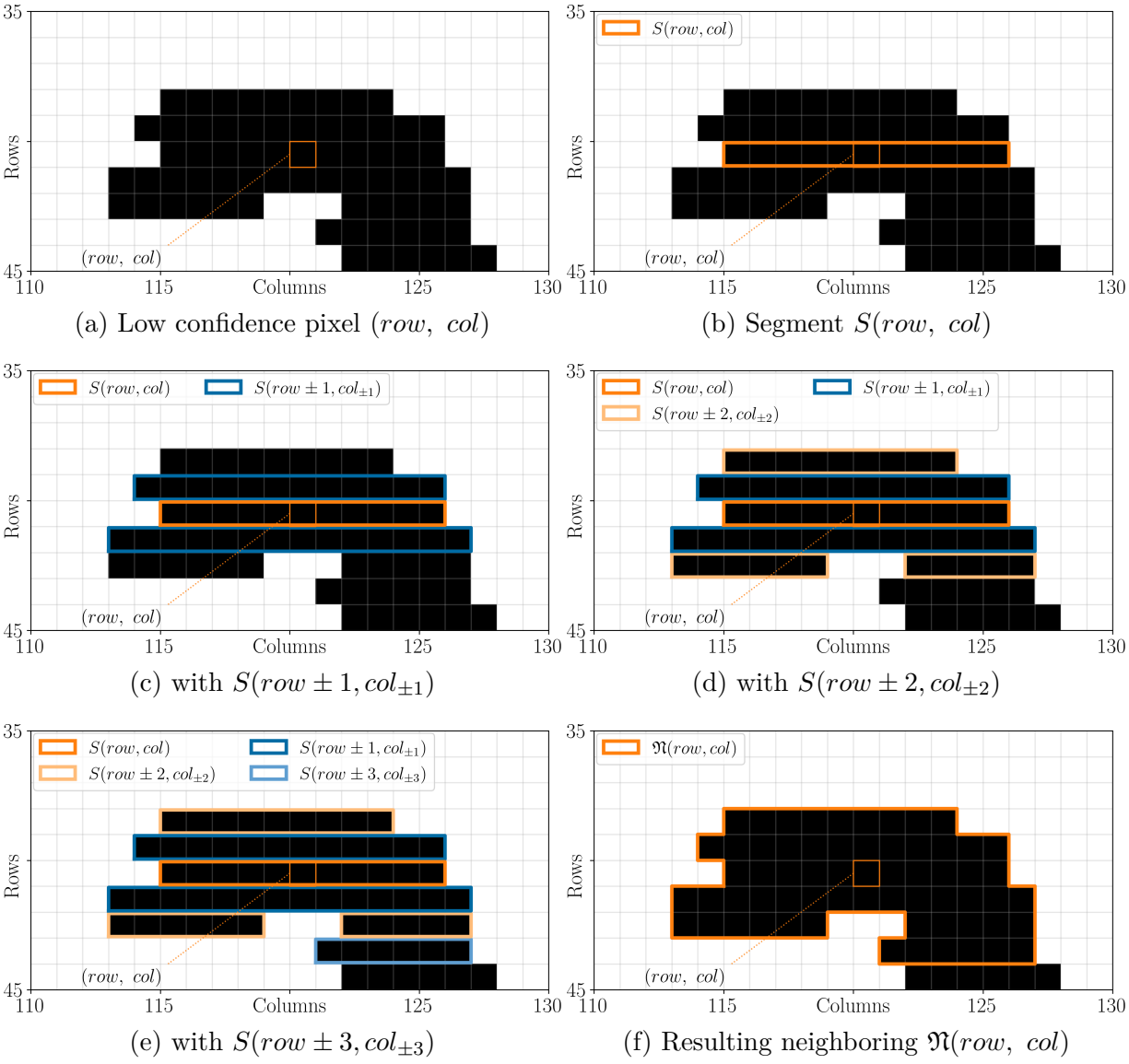
$$\begin{aligned}\mathfrak{N}(row,\ col) = \{p \in \bigcup_{-n_{\mathfrak{N}} \leqslant k \leqslant n_{\mathfrak{N}}} & S(row+k,\ col_k) \\ & s.t.\ S(row+(k+1),\ col_{k+1}) \text{ is adjacent to } S(row+k,\ col_k)\}\end{aligned} \tag{5.13}$$

with $col_0 = col$, and $n_{\mathfrak{N}}$ the number of consecutive rows considered. Figure 5.10 displays a graphical example of $S(row,\ col)$ and $\mathfrak{N}(row,\ col)$ from the Middlebury Cones left image.

The value of the regularized interval I_α^{reg} of $(row,\ col)$ is obtained by consensus between the confidence intervals of $\mathfrak{N}(row,\ col)$. Its upper and lower bounds are respectively the q^{th} quantile of upper bounds of $\mathfrak{N}(row,\ col)$ and the $(1-q)^{\text{th}}$ quantile of lower bounds of $\mathfrak{N}(row,\ col)$:

$$\begin{aligned}I_\alpha^{reg} = [&\mathcal{Q}_{1-q}(\{\underline{I}_\alpha(r,\ c) \mid (r,\ c) \in \mathfrak{N}(row,\ col)\}), \\ &\mathcal{Q}_q(\{\overline{I}_\alpha(r,\ c) \mid (r,\ c) \in \mathfrak{N}(row,\ col)\})]\end{aligned} \tag{5.14}$$

where $\mathcal{Q}_q$ refers to the q^{th} quantile of a set. In practice, we use $q = 90\%$. This way, the



(a) Low confidence pixel $(row,\ col)$

(b) Segment $S(row,\ col)$

(c) with $S(row \pm 1, col_{\pm 1})$

(d) with $S(row \pm 2, col_{\pm 2})$

(e) with $S(row \pm 3, col_{\pm 3})$

(f) Resulting neighboring $\mathfrak{N}(row,\ col)$

Figure 5.10: Segments $S(row,\ col)$ and neighboring $\mathfrak{N}(row,\ col)$. The image is an extract of the binary mask from Figure 5.8(c), where low confidence pixels appear in black and high confidence pixels appear in white. (a): low confidence pixel $row,\ col$. From (b) to (e): segment $S(row,\ col)$ and adjacent segments over 3 rows above and below. (f): resulting neighboring $\mathfrak{N}(row,\ col)$.

lower bound of I_α^{reg} is lower than 90% of lower bounds of intervals in $\mathfrak{N}(row,\ col)$, and its upper bound is larger than 90% of upper bounds of intervals in $\mathfrak{N}(row,\ col)$. Figures 5.13 and 5.14 present confidence intervals without and with regularization, for the same rows as in Figures 5.5 and 5.6.

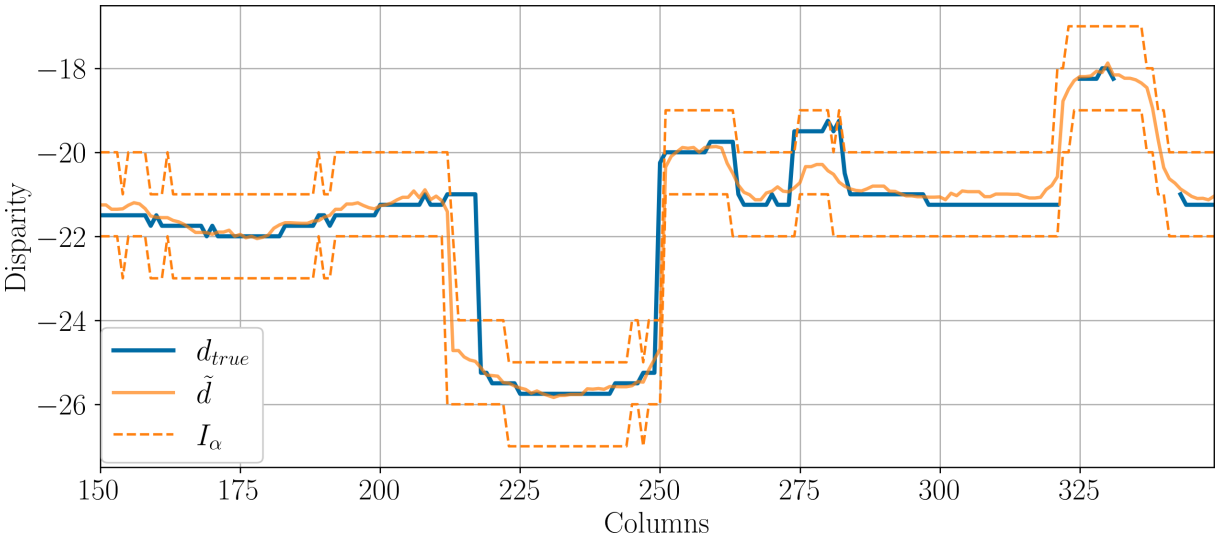
> **Remark:** It does not necessarily mean that I_α^{reg} includes 90% of intervals of $\mathfrak{N}(row,\ col)$. Indeed, there can be intervals whose lower bound is greater than the $(1-q)^{\text{th}}$ quantile of lower bounds while also being greater than the q^{th} quantile of upper bounds. The intervals will then overlap while not being included in one another.

We can notice that it is possible, although unlikely, that the q^{th} quantile of the upper bounds of $\mathfrak{N}(row,\ col)$ is lesser than the predicted disparity $\tilde{d}(row,\ col)$ (or similarly, the $(1-q)^{\text{th}}$ quantile of lower bounds can be greater than $\tilde{d}(row,\ col)$). In that case, in order to ensure the coherence of the predicted disparity with the confidence intervals, the bounds of the intervals are extended until they include $\tilde{d}$.

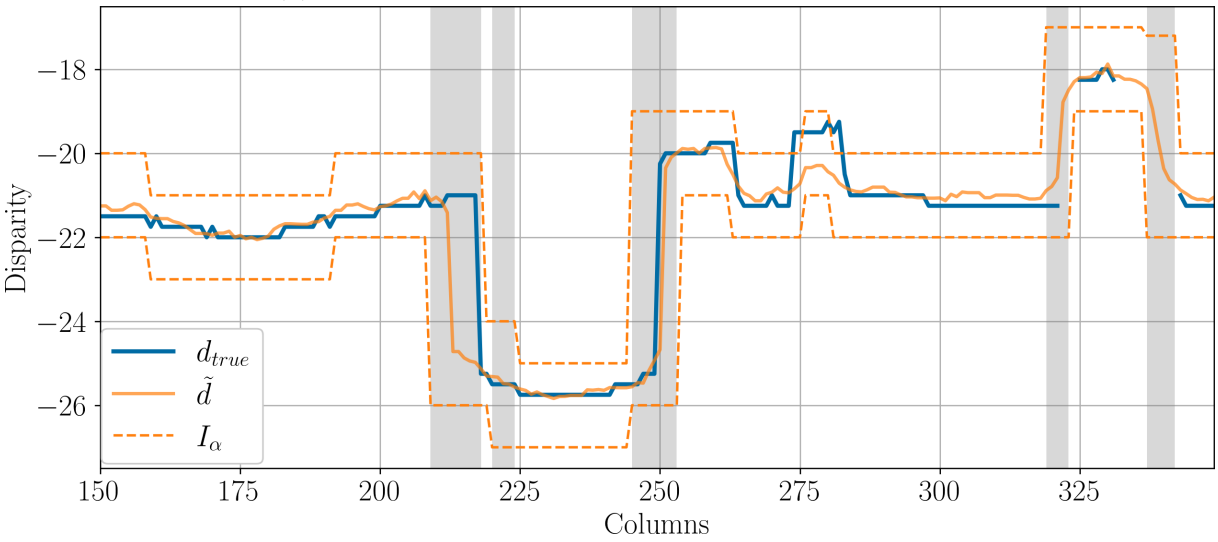
Two pixels $(row,\ col)$ and $(row,\ col')$ of $S(row,\ col)$ will have the same neighboring $\mathfrak{N}(row, col)$. Therefore, they will also share the same value for their regularized interval I_α^{reg}. This can be observed in Figures 5.11(b), 5.12(b), 5.13(b) and 5.14(b), where positions of low confidence pixels that are regularized are indicated using gray areas.

In the following, we always regularize intervals in low confidence areas. We will thus refer to them simply as I_α instead of I_α^{reg}, to simplify notations.

In theory, it is possible to perform the regularization of intervals before the filtering and refinement steps, but we chose to always regularize the intervals after those steps. This prevents outliers removed by the filtering step from influencing values of regularized intervals. Figures 5.11 to 5.14 allow visualizing the impact of the regularization of intervals after the filtering and refinement steps. In Figure 5.11(b) near column 215, we can see that the regularization allows to create correct confidence intervals. In Figure 5.12, columns 70 and 175 also create correct intervals, in regions where the predicted disparity $\tilde{d}$ is far away from the ground truth. We can say that the intervals are (almost) as small as possible so as to both contain $\tilde{d}$ and the true disparity d_{true}, as they are necessarily constant in low confidence areas (gray areas in the figure). Column 175 of Figure 5.12 also shows that filtering and regularization methods are able to discard outliers: bounds of non-regularized intervals reach values between -60 and 0 near column 175, while the bounds of regularized intervals stay between -42 and -15. However, this method is not perfect, as it sometimes overestimates the size of intervals as in Figure 5.14(b) near column 110, or do not predict correct intervals at all, as observed around column 450.

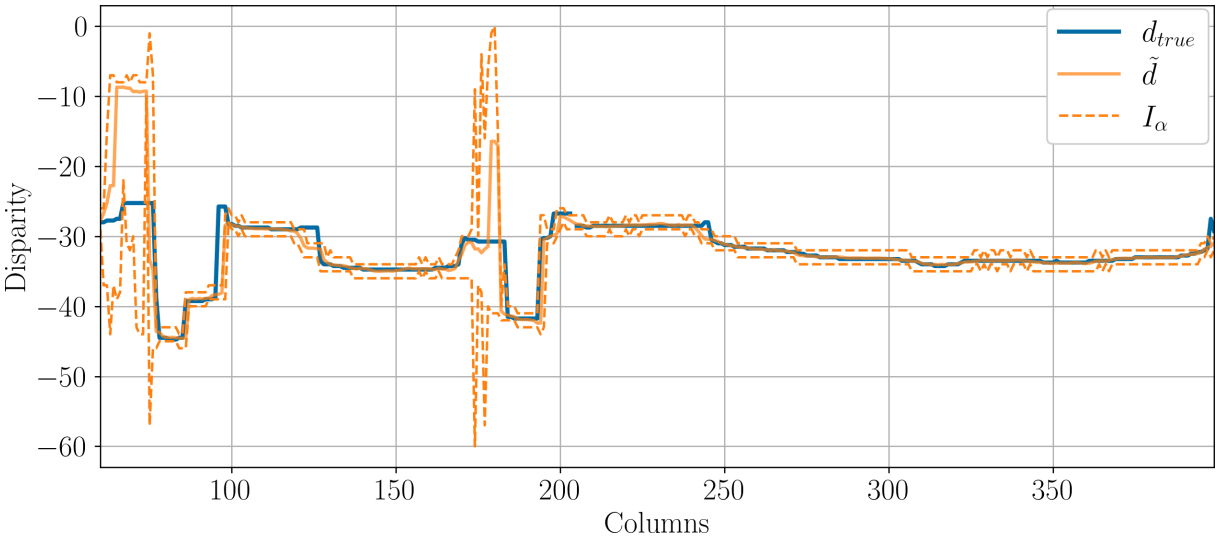


(a) I_α without regularization in low confidence areas

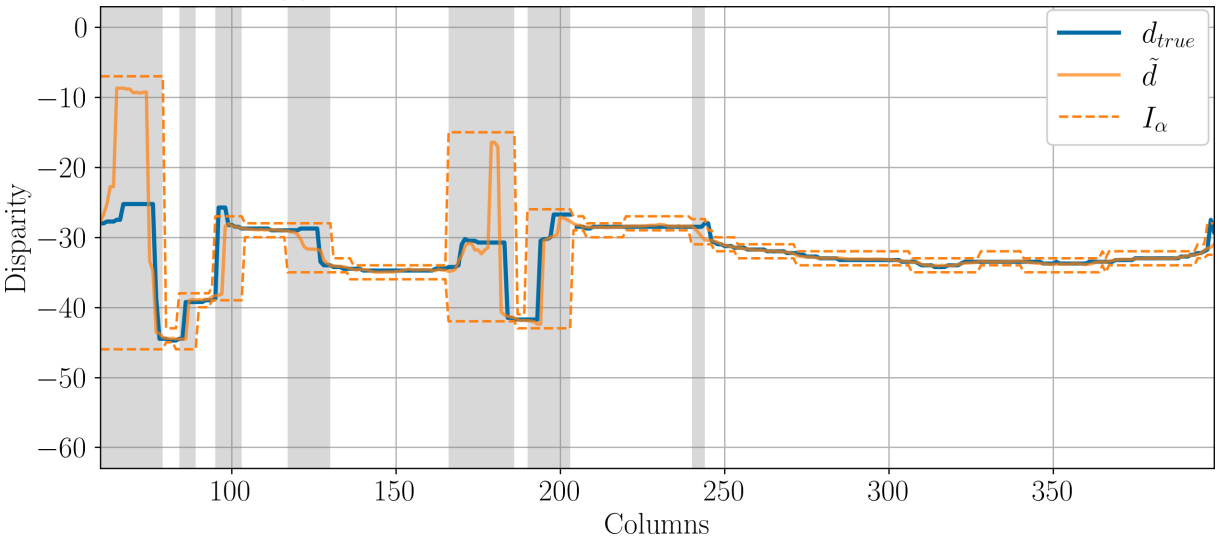


(b) I_α with regularization in low confidence areas

Figure 5.11: I_α with and without regularization in low confidence area, for row 80 of the image of Figure 5.4. The gray areas indicate low confidence areas.

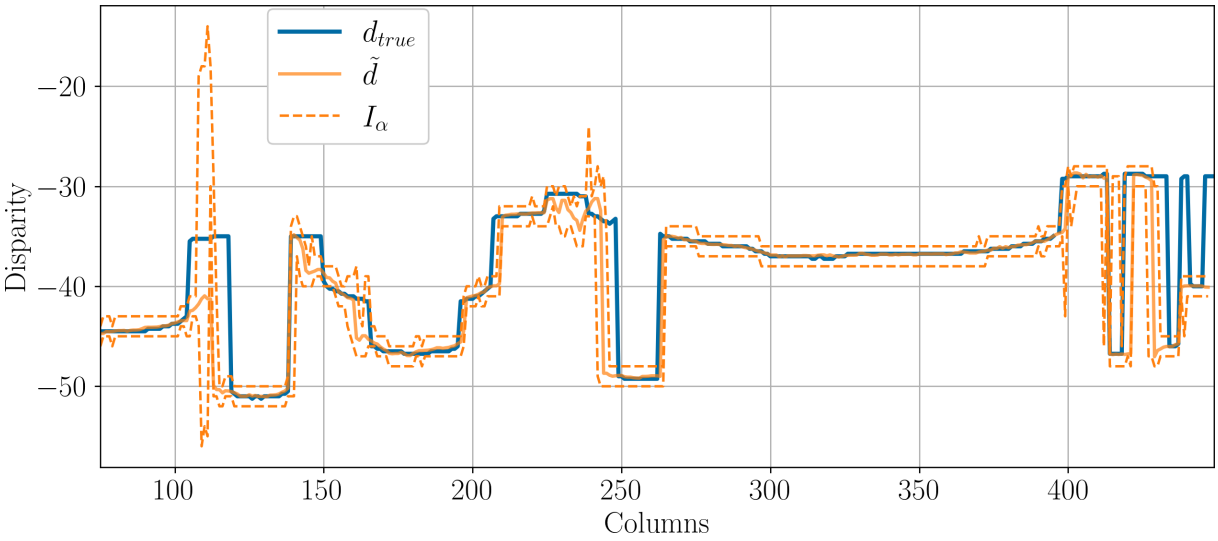


(a) I_α without regularization in low confidence areas

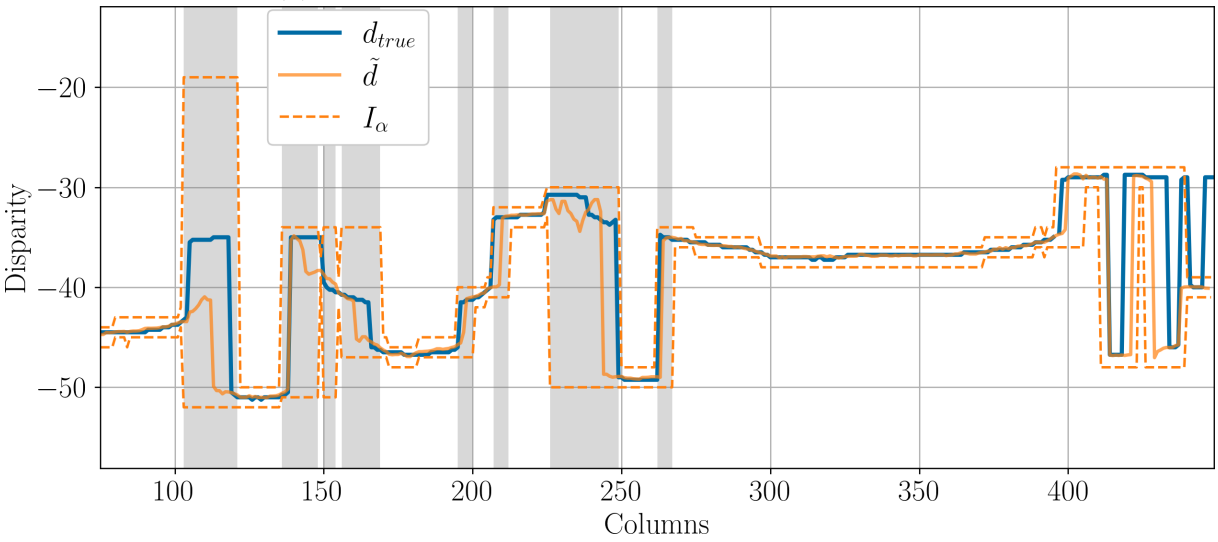


(b) I_α with regularization in low confidence areas

Figure 5.12: I_α with and without regularization in low confidence area, for row 180 of the image of Figure 5.4. The gray areas indicate low confidence areas.

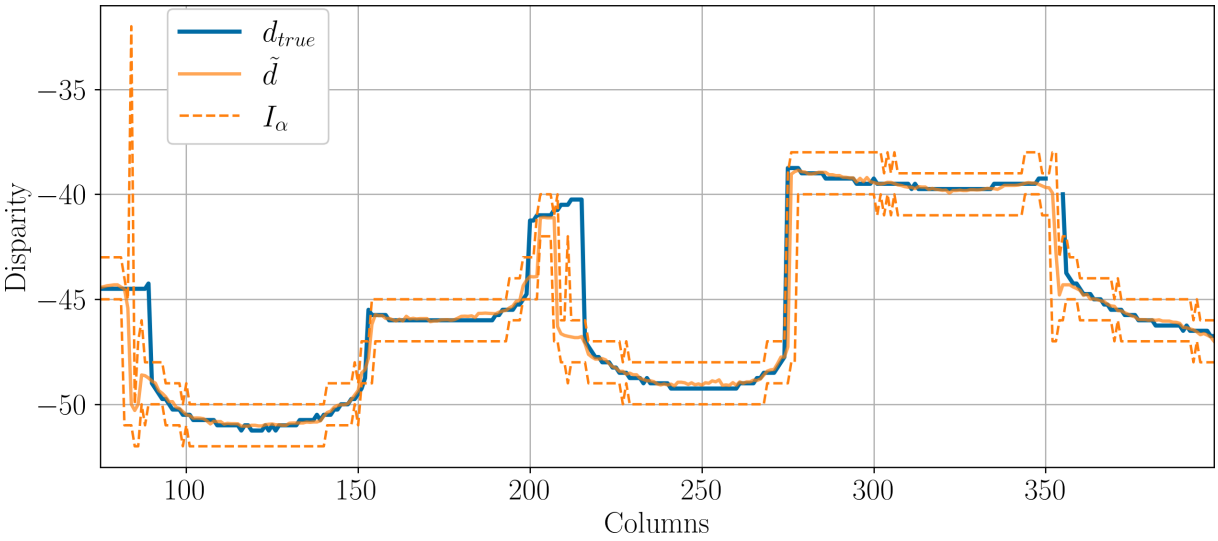


(a) I_α without regularization in low confidence areas

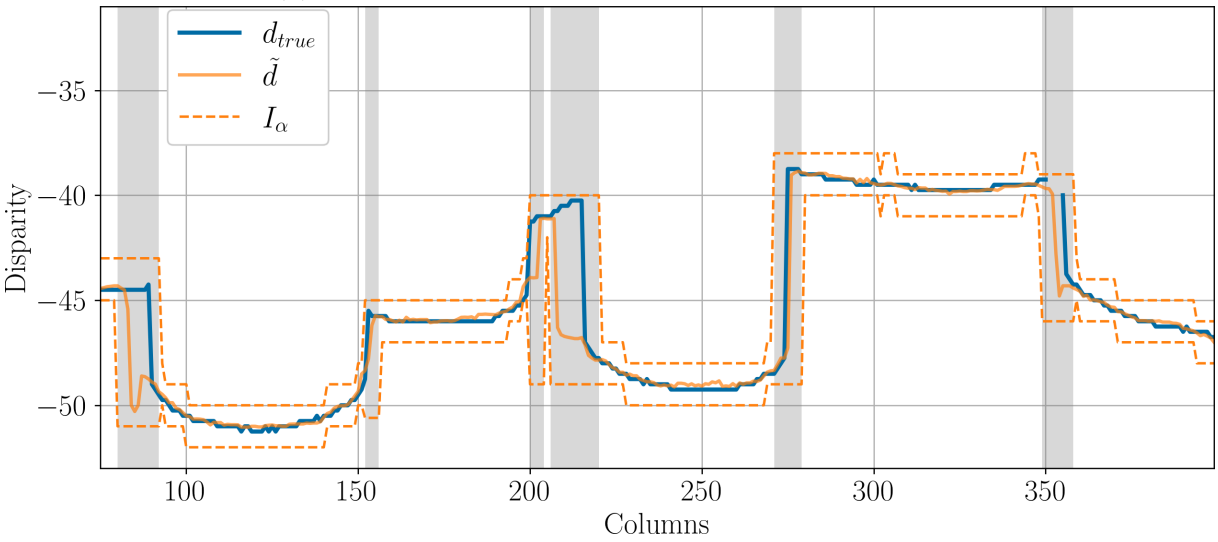


(b) I_α with regularization in low confidence areas

Figure 5.13: I_α with and without regularization in low confidence area, for row 240 of the image of Figure 5.4. The gray areas indicate low confidence areas.



(a) I_α without regularization in low confidence areas



(b) I_α with regularization in low confidence areas

Figure 5.14: I_α with and without regularization in low confidence area, for row 290 of the image of Figure 5.4. The gray areas indicate low confidence areas.

(a) Without regularization, filtering and refinement

(b) With regularization, filtering and refinement

Figure 5.15: Left image from Middlebury cones. Confidence intervals that do not contain the ground truth appear in orange.

Figure 5.15 allows visualizing the position of wrong intervals in the left stereo image. We can see clear improvements between the method with and without regularization. This can especially be observed in the bottom right corner of images near the sticks inside the cup, or in general near cones borders. Quantitatively, wrong intervals represent around 5% of pixels without regularization, and 1.6% with regularization. Those numbers are presented for information purposes only, as further scores will be computed on different scenes, allowing for more in-depth analysis.

This concludes the method for computing disparity confidence intervals in the dense stereo matching part of the photogrammetry pipeline. We will see next how those intervals can be propagated into confidence intervals for the DSMs.

5.2 Evaluation of Disparity Intervals

5.2.1 Metrics for Evaluating the Accuracy and Size

In order to evaluate the performance of the disparity confidence intervals, we will now introduce some metrics. To the best of our knowledge, there is no current method for producing disparity confidence intervals to use for comparison. The objective of this section is to propose a range of tools able to quantify the trade-off between accuracy and size of intervals. For each metric, we do not consider pixels without ground truth, pixels at the border of images that cannot fully explore their disparity range, or pixels discarded

by the cross-checking test (Equation (1.14)). Pixels at the border of an image for which the disparity range cannot be fully explored were also discarded, as the ground truth exists but is often unreachable.

Accuracy

The first and most obvious metric is the proportion of correct intervals, *i.e.* intervals containing the true disparity, which we call accuracy *acc*:

$$acc = \frac{\#\{I_\alpha \mid s.t.\ d_{true} \in I_\alpha\}}{\#\{I_\alpha\}} \tag{5.15}$$

We want to maximize the accuracy, and fixed ourselves a minimal objective of 90% accuracy for our method.

Residual Error

It is also interesting to measure the magnitude of the error. We thus introduce another metric called residual error ε, computed for intervals that do not contain the ground truth and defined as:

$$\varepsilon = \text{median}\left(\frac{\min(|d_{true} - \overline{I}_\alpha|, |d_{true} - \underline{I}_\alpha|)}{d_{max} - d_{min}}\right) \tag{5.16}$$

where $[d_{min},\ d_{max}]$ is the considered disparity range. Normalization by the disparity range yields a metric that can be compared across datasets with different disparity ranges. For instance, in the 2003 Middlebury "Cones" dataset, the disparity range is $[-60,\ 0]$, while in the 2014 Middlebury "Shopvac-perfect" dataset, the disparity range is $[-1100,\ 0]$.

Residual error and other metrics are defined using the median of a set, instead of the mean, so as to be less sensitive to outliers. We will still present the distributions of the metrics without the median operator in Section 5.2.3. The residual error allows quantifying how "wrong" were the intervals. A residual error near 0 translate the fact that the confidence intervals were really close to capture the ground truth, while a residual error near 1 indicate that the intervals were far off the ground truth. Figure 5.17 helps to visualize ε.

Because half of pixels have a relative error less than ε, extending each bound of the intervals by $\varepsilon \cdot (d_{max} - d_{min})$ effectively divides the global error by two. The accuracy would therefore be $acc' = acc + \frac{(1 - acc)}{2}$. ε thus provides a quick and easy estimation of the length missing to the intervals in order to divide the missing accuracy by two.

Relative Size

We now introduce a metric to measure the size of intervals. This metric evaluates the relative size of intervals, with regard to the size of the considered disparity range:

$$s_{rel} = \text{median}\left(\frac{\overline{I}_\alpha - \underline{I}_\alpha}{d_{max} - d_{min}}\right) \tag{5.17}$$

where $[d_{min},\ d_{max}]$ is the considered disparity range. We use the same normalization as for the residual error ε We want to minimize s_{rel} to ensure our intervals are not too large.

Relative Over-Estimation

During the regularization of confidence intervals, we purposefully extended the bounds of the intervals in low confidence areas. The relative size of intervals s_{rel} in those areas will, by design, be very large. We therefore propose to evaluate the relative size only on high confidence areas, and to introduce a specific measure for low confidence areas, called relative over-estimation. Defined when intervals correctly contain the true disparity, it is computed as follows:

$$o_{rel} = \text{median}\left(1 - \frac{\Delta|d_{true} - \tilde{d}|}{\overline{I}_\alpha - \underline{I}_\alpha}\right) \tag{5.18}$$

where $\Delta|d_{true} - \tilde{d}|$ is defined as:

$$\begin{aligned}\Delta|d_{true} - \tilde{d}| = \max(\{|d_{true}(row',\ col') - \tilde{d}(row'',\ col'')|,\ \text{with} \\ (row',\ col') \in \mathfrak{N}(row,\ col),\ (row'',\ col'') \in \S(row,\ col)\})\end{aligned} \tag{5.19}$$

In other words, $\Delta|d_{true} - \tilde{d}|$ is the maximal difference between the true disparity and the predicted disparity over the same low confidence segment. We consider $S(row,\ col)$ and not the whole neighboring $\mathfrak{N}(row,\ col)$ as the intervals have the same value only in a segment S (see Section 5.1.4 for more details). Note that the true disparity and predicted disparity can belong to two different pixels in the computation of $\Delta|d_{true} - \tilde{d}|$. $\Delta|d_{true} - \tilde{d}|$ is the size of the optimal interval, *i.e.* the smallest interval containing both d_{true} and $\tilde{d}$ in the low confidence area. $1 - \dfrac{\Delta|d_{true} - \tilde{d}|}{\overline{I}_\alpha - \underline{I}_\alpha}$ therefore represent the superfluous proportion of intervals, or in other words, the proportion of $\overline{I}_\alpha - \underline{I}_\alpha$ that is over-estimating the error $\Delta|d_{true} - \tilde{d}|$. Figure 5.16 illustrates the meaning of $\Delta|d_{true} - \tilde{d}|$ and I_α for the computation of the relative over-estimation inside a low confidence area.

Because we want to obtain the smallest possible correct intervals, it only makes sense to compute o_{rel} for confidence intervals that do contain the true disparity. Indeed, consid-

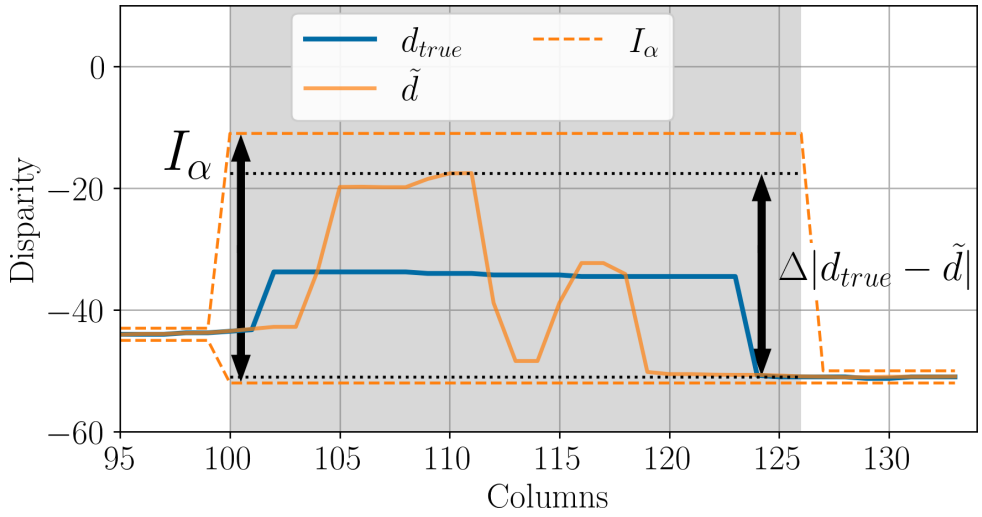


Figure 5.16: $\Delta|d_{true} - \tilde{d}|$ and I_α for computing the relative estimation (Equation (5.18)) over a low confidence area in gray.

ering wrong intervals could reduce the over-estimation and induce bias in our conclusions. Doing so also insure o_{rel} is always contained between 0 and 1, and can be expressed as a percentage.

We want the relative over-estimation to be as close to 0 as possible, as $o_{rel} = 0$ means that I_α is the optimal interval for at least half of the low confidence pixels. $1 - \frac{\Delta|d_{true} - \tilde{d}|}{\overline{I}_\alpha - \underline{I}_\alpha} = 0.1$ means that the superfluous part of the interval represents a tenth of its total length. Similarly, $1 - \frac{\Delta|d_{true} - \tilde{d}|}{\overline{I}_\alpha - \underline{I}_\alpha} = 0.5$ means that the superfluous part of the interval represents half of their total length.

Reaching a relative over-estimation of 0 is not realistic, as it would be equivalent to say that we exactly know the position of the true disparity for each pixel. This is irrational, as it would mean we have the perfect stereo matching algorithm, but somehow still predicted a wrong disparity. A more realistic objective is to be around 50%, even though less is better.

> **Remark:** The relative over-estimation could theoretically reach a value of 1 regardless of the size of intervals if $\Delta|d_{true} - \tilde{d} = 0$. This is the case if and only if the predicted disparity and the true disparity are equal and constant for all pixels in the considered low confidence area. This is however very unlikely, as low confidence areas are usually areas with strong variations of disparity, and where the stereo algorithm struggles to predict a correct disparity.

Because we evaluate the size of intervals using two different metrics depending on if intervals are in low confidence areas or not, it is interesting to measure the proportion of

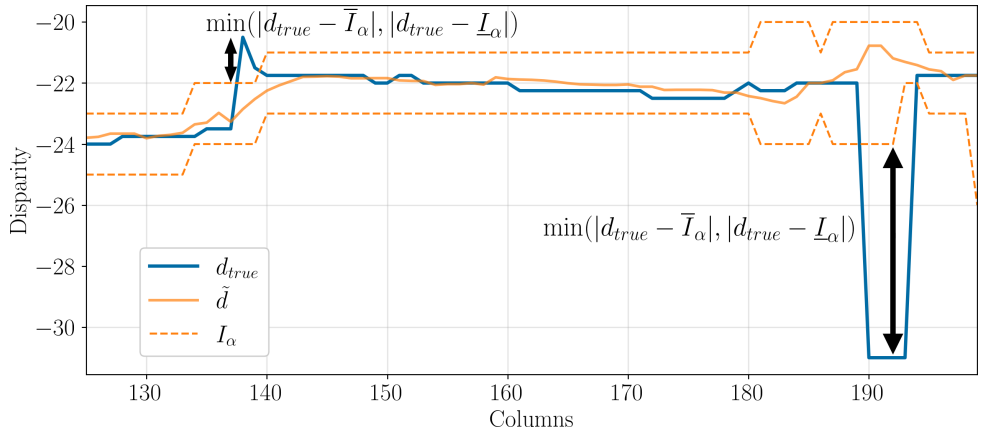


Figure 5.17: Different values of $\min(|d_{true} - \overline{I}_\alpha|, |d_{true} - \underline{I}_\alpha|)$ for computing the residual error of Equation (5.16)

low confidence areas in the scene p_{amb}:

$$p_{amb} = \frac{\#\{I_\alpha \mid I_\alpha \in \text{low confidence area}\}}{\#\{I_\alpha\}} \tag{5.20}$$

A proportion p_{amb} of intervals size will be evaluated using o_{rel}, and a proportion $1 - p_{amb}$ using s_{rel}. A high p_{amb} thus indicates many regularizations and consequently larger intervals in general.

Before measuring the accuracy of intervals using acc, to quantify their errors using ε or measuring their relative size with s_{rel} and o_{rel}, we will present the different considered datasets used for our evaluation.

5.2.2 Stereo Matching Dataset

We evaluate the disparity confidence intervals on 76 scenes from the Middlebury datasets. Those datasets are composed of two stereo images of indoor scenes, for which the true disparity is exactly known. It is divided in different years: 2003, 2005, 2006, 2014, and 2021 [Scharstein and Szeliski, 2003, Scharstein and Pal, 2007, Hirschmuller and Scharstein, 2007, Scharstein et al., 2014], and the complexity of scenes increases with years. Figure 5.18 present some scenes from each dataset. Each dataset is available in different resolutions. We use quarter-size and third-size versions of the data for 2003, 2005 and 2006 datasets and full resolution for 2014 and 2021 datasets in order to include a diversity of resolutions in our experiments. As a result, ranges of considered disparities also vary greatly. For instance, disparity ranges of the 2014 dataset are very large (more than a thousand disparity wide), which presents a significant challenge for the stereo matching algorithm, as many disparities need to be explored. On those scenes, our stereo matching algorithm performs far less well than on the 2003 dataset. This allows evaluating whether the disparity confidence intervals can perform well even when the main disparity predic-

(a) 2003 Cones (b) 2005 Art (c) 2006 Aloe

(d) 2014 Jadeplant Perfect (e) 2021 Artroom1 (f) 2021 Octogons1

Figure 5.18: Example of left images from different Middlebury datasets

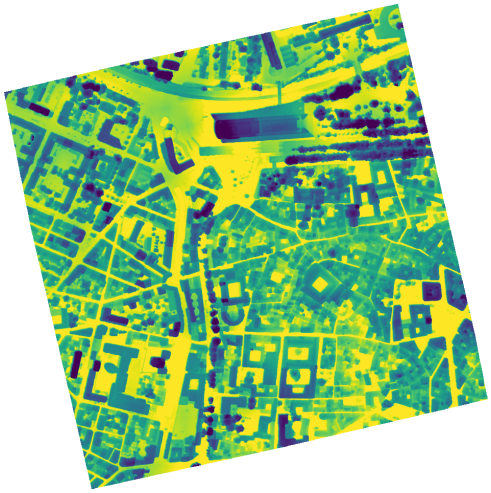
tions do not. Each dataset does not contain the same number of images, and shapes of images can vary across datasets and inside each dataset. For indication purposes, datasets from 2003, 2005, 2006, 2014, and 2021 respectively contain 2, 6, 21, 23 and 25 scenes.

The Middlebury dataset contains generic indoor scenes with a high variety of 3D structures, which highlights the potential of our method for any stereo algorithm. Our objective is however to process satellite images, so we also evaluate our method on satellite data, which can greatly differ from indoor scenes. For this purpose, we use 80 pairs of 1845×1845 satellite images in epipolar geometry, with a typical disparity range of $[-20, 10]$ (it slightly varies between images). Those images are all part of the same pair of Pléiades images over the city of Montpellier (France), which is large enough to contain both urban and rural areas, as presented in Figure 5.19. The ground truth of those images was obtained using the method described in [Cournet et al., 2020]. In a few words, this method first processes stereo images to obtain images in epipolar geometry. Then it considers the ground truth DSM of the scene, for instance a rasterized LiDAR HD data, and project it into the same geometry as one of the sensors. Then, using the disparity to altitude ratio computed alongside epipolar images, it converts the altitude of the re-projected ground truth DSM into disparities. Using the same epipolar grids, this true disparity map can be projected into epipolar geometry. We now have two stereo images and their associated "ground truth" disparity map in epipolar geometry, ready to be processed. This method has a few known drawbacks. First, the ground truth is

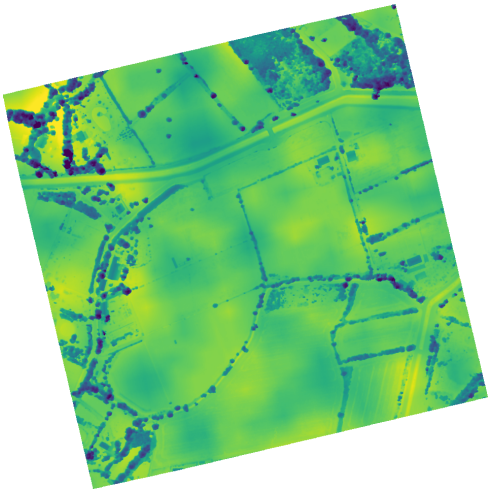
resampled in two different geometries, which relies on the quality of the epipolar grid, the planimetric accuracy of the ground truth data, and satellite images. Secondly, the images and ground truth were acquired a few years apart, which results in some new buildings being built or destroyed between stereo images and ground truth. The vegetation also changed, so the ground truth is not always 100% correct. The dataset initially contained 327 pairs of images, but we detected major differences between the ground truth and the epipolar images, as illustrated in Figure 5.20, so we restricted our selection to 80 pairs that did not exhibit major differences. The evaluation metrics on this dataset must thus be taken with care.

(a) Left epipolar image

(b) Left epipolar image



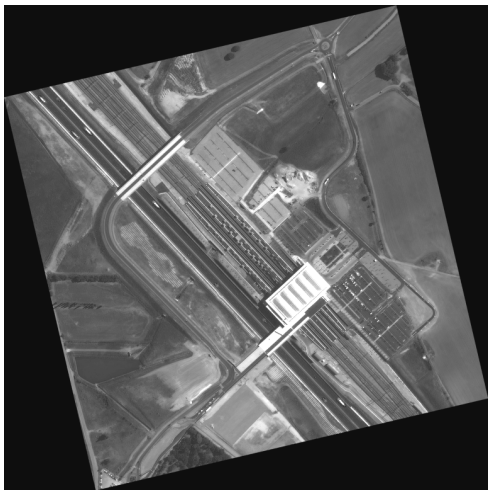
(c) Ground truth disparity



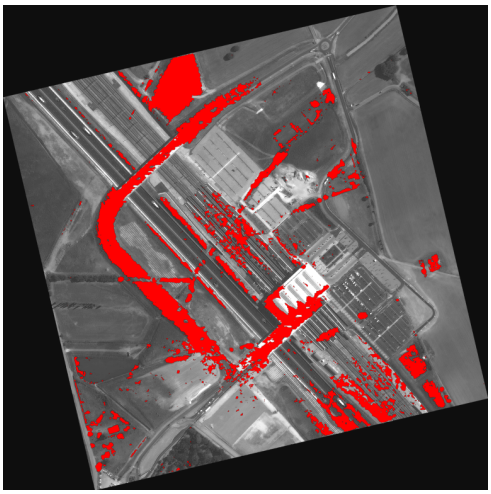
(d) Ground truth disparity

Figure 5.19: One rural and one urban scene from the Montpellier dataset.

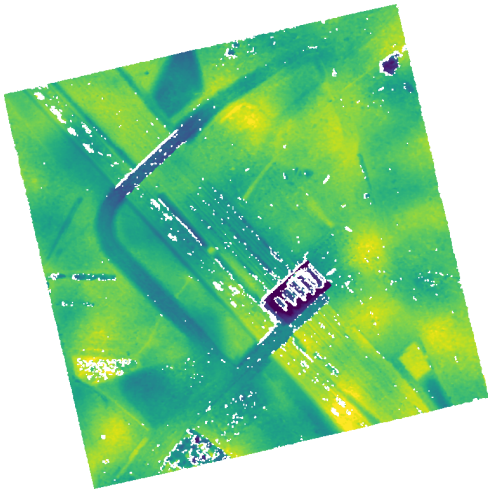
In the next section, we will use the Middlebury datasets to evaluate our method on general indoor scenes, and images of Montpellier (abbreviated as MTP) to determine if we can generalize our results to satellite imagery. We will also consider two cost functions: the CENSUS cost function and MC-CNN cost function, both detailed in Section 1.3.3.



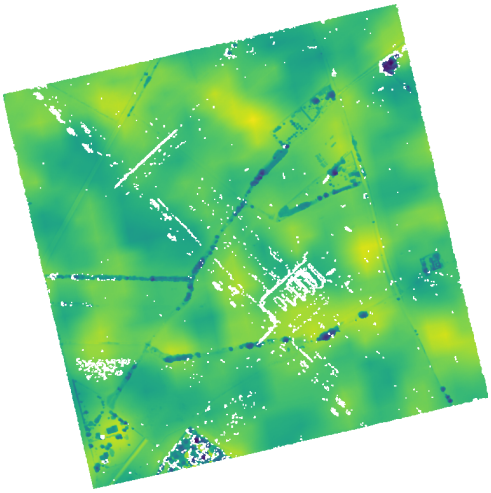
(a) Left epipolar image



(b) Wrong intervals (red)



(c) Predicted disparity



(d) Ground truth disparity

Figure 5.20: Important differences between the ground truth and epipolar images. We can see that the road appearing in the left image from Figure 5.20(a) does not appear in the ground truth in Figure 5.20(d), which results in false negative on the intervals from Figure 5.20(b). The road must have been constructed in between the ground truth acquisition and image acquisition. Images with such differences were removed from the Montpellier dataset.

We also remind here the configuration of the stereo matching algorithm used. The CENSUS cost function uses a 5×5 window, and SGM penalties $P_1 = 8$, $P_2 = 32$. The MC-CNN cost function uses a 11×11 window, and SGM penalties $P_1 = 2.3$, $P_2 = 42.3$. The disparity refinement step uses V-fit interpolation. We also a 3×3 median filter on the disparity map.

5.2.3 Results

This section will evaluate each metric on the different datasets, and for the two considered cost functions, CENSUS and MC-CNN. Because there are a few metrics and many scenes,

we first give global results, and gradually go into more details across tables and figures. We first consider global results on Middlebury datasets, as they possess a ground truth of better quality. Each metric is evaluated and averaged over the whole dataset in Table 5.1 to get an overall estimation of its performance. To have a more detailed estimation of scores distribution across scenes, we then plot histograms of each metric across all datasets in Figures 5.21 to 5.24. Finally, we will also discuss the metrics by looking at some details of particular scenes in Figures 5.26 to 5.28. Figure 5.25 also presents the distribution from which some metrics are computed over the Middlebury Cones stereo images.

	Year	2003	2005	2006	2014	2021
$acc \uparrow$	CENSUS	97.6%	96.4%	99.1%	94.8%	91.6%
	MC-CNN	97.0%	97.5%	99.3%	98.5%	99.1%
$\varepsilon \downarrow$	CENSUS	2.5%	3.1%	5.3%	0.2%	0.9%
	MC-CNN	4.0%	5.6%	10.0%	3.8%	8.3%
$s_{rel} \downarrow$	CENSUS	3.3%	2.6%	2.6%	0.6%	1.2%
	MC-CNN	3.3%	2.6%	3.0%	1.5%	3.3%
$o_{rel} \downarrow$	CENSUS	55.8%	66.9%	73.9%	70.9%	67.5%
	MC-CNN	58.5%	71.3%	80.4%	86.3%	78.0%
p_{amb}	CENSUS	20.8%	29.1%	27.1%	47.1%	59.2%
	MC-CNN	15.3%	41.1%	39.9%	43.2%	69.1%
d_1	CENSUS	93.4%	88.9%	91.8%	73.7%	58.2%
	MC-CNN	92.8%	89.1%	93.3%	75.5%	62.6%

Table 5.1: Average metrics over the different Middlebury datasets, depending on the cost function. Up arrows indicate that the optimal score is 100%, and 0% for down arrows. p_amb is the proportion of low confidence area, computed with the ambiguity. d_1 is an indicator of the predicted disparity performance defined in Equation (5.21).

In Table 5.1, each metric is evaluated and averaged over each Middlebury dataset: 2003 to 2021. We also indicate the proportion of low confidence area computed with the ambiguity p_amb, as well as the proportion of accurate predicted disparity d_1 defined as:

$$d_1 = \frac{\#\{\tilde{d} \ s.t. \ |\tilde{d} - d_{true}| < 1\}}{\#\{\tilde{d}\}} \tag{5.21}$$

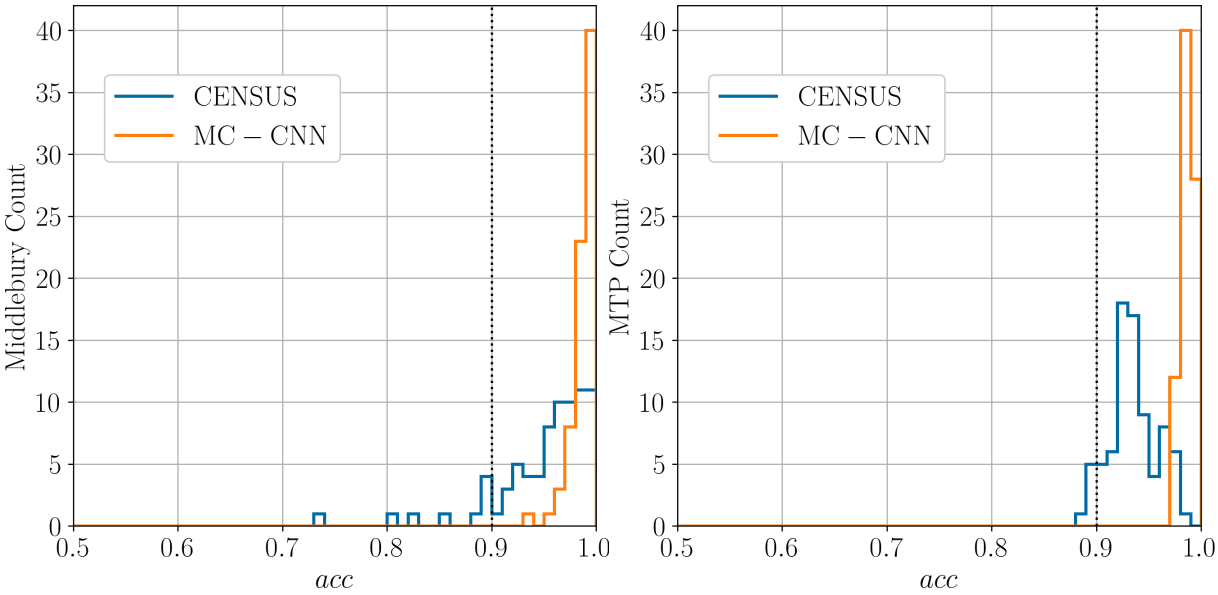
d_1 serves as an indicator of whether the stereo algorithm predicted a good disparity, regardless of confidence intervals. The conclusions drawn for this table are quite general, as we are looking at averages across multiple datasets. We will delve into more details later in this section. Here are some of the key takeaways from this table:

- Accuracy *acc*: the first observation is that the accuracy is greater than the 90% objective on each dataset, and for both considered cost functions. In fact, the average is always higher than 91.6%, and reaches 99.3% accuracy on some datasets (2006 with MC-CNN). Except for the 2003 dataset, it seems that intervals computed using MC-CNN are more accurate than those with CENSUS. Comparing the accuracy of intervals with the performance d_1 of the predicted disparity, we can observe that confidence intervals are accurate even when the predicted disparity struggles to correctly estimate the true disparity. It is especially apparent for the 2021 dataset, where d_1 is around 60% for both cost functions, while *acc* is above 90%. This high accuracy must however be considered alongside the size of the intervals with s_{rel} and o_{rel}.

- Residual error ε: residual errors values are between 0.2% and 10% for all datasets, which is relatively low. The residual error is always lower for CENSUS intervals than for MC-CNN ones. In particular, the residual error using CENSUS is almost ten times smaller than that using MC-CNN for the 2021 dataset. This means that it is easier to improve the accuracy of CENSUS than MC-CNN intervals, as they miss the true disparity by a smaller fraction of the disparity range. With the conclusions drawn from the accuracy results, it seems that the last few missing accuracy percent are the hardest to recover.

- Relative size s_{rel}: The relative size of intervals in high confidence areas is relatively low. Across datasets, it is on average between 0.6% and 3.3%. This broadly means that half of intervals sizes in high confidence areas are less than 5% of the considered disparity range. This is a very reasonable size for confidence intervals. For datasets 2003 to 2006, both cost functions lead to similar relative sizes. For datasets 2014 and 2021, the MC-CNN cost function leads to intervals around 3 times bigger than CENSUS intervals. This is coherent with the higher accuracy of MC-CNN intervals on those datasets.

- Relative over-estimation o_{rel}: The range of relative over-estimation of intervals across datasets is more widely spread. It ranges between 46.8% and 72.6%. Except for the 2003 dataset, the MC-CNN cost function clearly leads to intervals with higher over-estimation than those obtained using CENSUS. CENSUS intervals typically over-estimate intervals in low confidence areas by around 50%, while it is around 60% or 70% for MC-CNN intervals. This means that, when using CENSUS, half of low confidence pixels have 50% or less of their size that is superfluous. For MC-CNN, the proportion of superfluous size reaches 60% to 70%. For CENSUS, this score is around what we deem to be a good performing intervals. For MC-CNN, it seems to be a bit large. We must however take into consideration the complexity



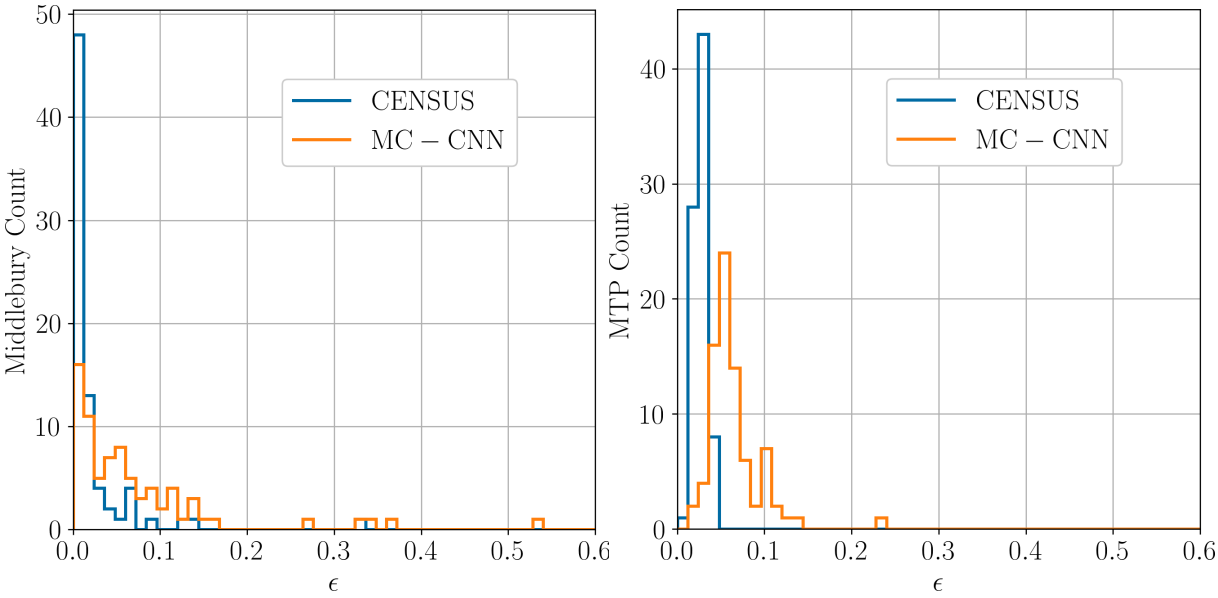
(a) Middlebury (b) Montpellier

Figure 5.21: Accuracy *acc* histograms over the different datasets depending on the cost function. Each histogram counts the number of scenes. The vertical dotted line represent the 90% objective.

of datasets 2014 and 2021. Indeed, we will see in some examples that the correlator has a hard time producing a good quality disparity map for some scenes, and that over-estimating the intervals is necessary to produce accurate intervals (for instance, in Figure 5.28).

- Proportion of low confidence area p_{amb}: Table 5.1 also indicates the proportion of low confidence areas. This provides indicative insights on the proportion of intervals that have been evaluated using o_{rel}, the rest being evaluated using s_{rel}. It is unclear if one cost function consistently results in more low confidence areas than the other. The proportion of low confidence areas do not seem to be correlated to the accuracy or to the residual error.

We have a good first estimation of the performance of the method for creating disparity confidence intervals. We can now delve into more details by looking at the distribution of score across all scenes, for the Middlebury datasets and the Montpellier images separately. The results are presented in Figures 5.21 to 5.24. Figure 5.21(a) presents the accuracy distribution across all Middlebury scenes. Regarding the Middlebury datasets, those figures support observations made in Table 5.1, as the distribution of most metrics are concentrated on the same values. The residual error varies between 0% and 10% for CENSUS intervals, while MC-CNN intervals present larger values. The relative size of intervals is for the most part less than 3%, with MC-CNN intervals presenting slightly larger relative size than CENSUS ones. For the CENSUS cost function, there are in



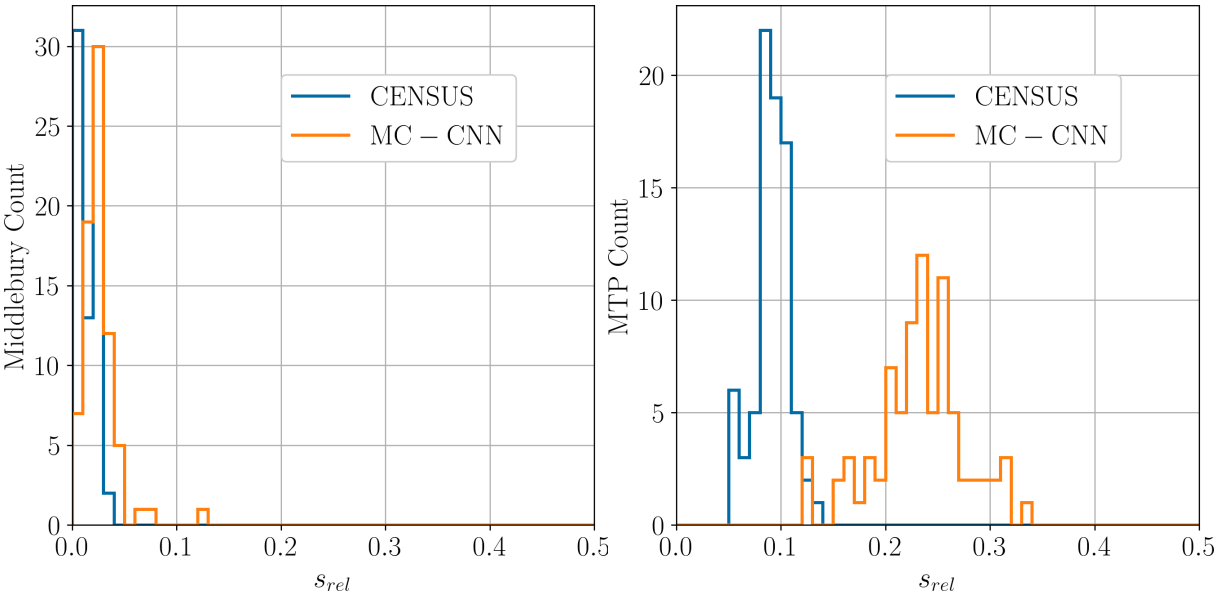
(a) Middlebury (b) Montpellier

Figure 5.22: Residual error ε histograms over the different datasets depending on the cost function. Each histogram counts the number of scenes.

total 9 scenes across the 76 Middlebury images that do not verify the 90% accuracy threshold, 4 of which have an accuracy between 89% and 90%. Those scenes are all taken from the 2014 and 2021 datasets, which are complex scenes with very large disparity intervals, typically not found in the stereo satellite images we consider for building a DSM. Intervals computed using the MC-CNN cost function all verify the 90% accuracy threshold. Another noteworthy remark concerns the relative over-estimation: CENSUS intervals values are concentrated around 50% but spread almost across the entire range, while MC-CNN intervals are more uniformly distributed along values in $[0.4, 0.9]$. This does not change the observation that using the MC-CNN cost function leads to a greater over-estimation of intervals than using the CENSUS cost function.

The differences between CENSUS intervals and MC-CNN intervals are also observed on the Montpellier dataset. The 90% accuracy objective is respected, and MC-CNN intervals are more accurate in general. Their residual error, relative size and relative over-estimation are all larger than those of CENSUS intervals. In general, the residual error, relative size and relative over-estimation on the Montpellier dataset are greater than on the Middlebury datasets, while still remaining relatively low. This is due to the smaller disparity range for those images (typically $[-20, 10]$) in compared to Middlebury ranges.

> **Remark:** MC-CNN cost function generates larger intervals than the CENSUS cost function, but they are more accurate in consequence. Tuning different parameters

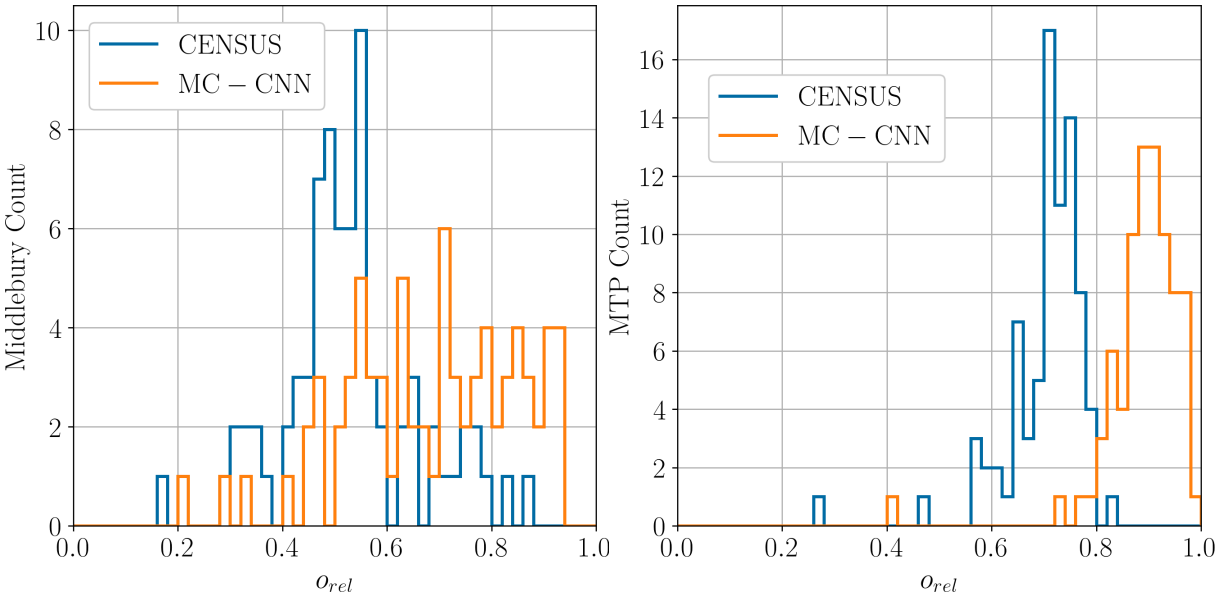


(a) Middlebury (b) Montpellier

Figure 5.23: Relative size s_{rel} histograms over the different datasets depending on the cost function. Each histogram counts the number of scenes.

> in our method, such as the value of $\alpha = 0.9$, the ambiguity threshold τ_{amb} or the size of the ambiguity kernel k_{amb}, could lead to the same performance of the two cost functions. This was not explored, as our primary objective is to propagate the disparity intervals to the rest of the photogrammetry pipeline.

We evaluated the metrics quantitatively across datasets and scenes in Table 5.1 and Figures 5.21 to 5.24. Those analyses gave us a global overview of the performance of the confidence intervals. However, providing a quantitative analysis on *local* performance is more complex and tedious. Figure 5.25 displays histograms (in number of pixels) of distributions from which ε, s_{rel} and o_{rel} are the medians, in the case of the Middlebury Cones stereo images. It provides a better estimation of the behavior of each metric. We can see in Figures 5.25(a) and 5.25(c) that the span of relative sizes and residual errors are close to their median ε and s_{rel}. In contrast, the relative over-estimation is spread over the range $[0, 1]$. There are also more intervals with an over-estimation near 1, which is due to $\Delta|d_{true} - \tilde{d}|$ being small for a number of pixels. We also present a qualitative analysis of confidence intervals in Figures 5.26 and 5.27. The accuracy can be observed on Figures 5.26(b) and 5.27(b), as the proportion of red pixels (indicating wrong intervals) is relatively low. The size of intervals in high confidence areas can be estimated by looking at pixels from columns 900 to 950 of Figure 5.26(c) or columns 700 to 800 of Figure 5.27(b). They seem to be around 2 disparities wide, which is quite small. In low confidence areas, for instance near column 850 of Figure 5.26(c) and column 830 of Figure 5.27(c), intervals do not over-estimate too much the maximum potential



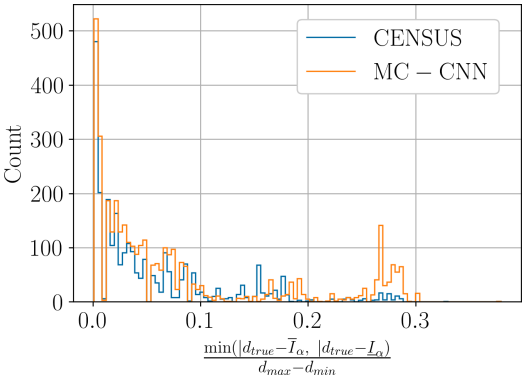
(a) Middlebury (b) Montpellier

Figure 5.24: Relative over-estimation o_{rel} histograms over the different datasets, depending on the cost function. Each histogram counts the number of scenes.

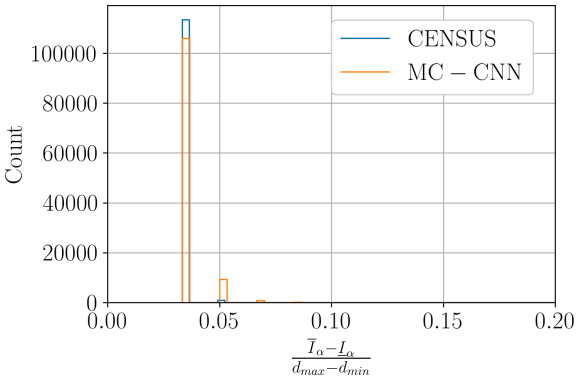
error between the predicted disparity and the true disparity. Figure 5.28 illustrates a case where the stereo matching algorithm has very poor performance of when it comes to the disparity prediction. Most areas are low confidence areas, and the predicted disparity is sometimes hundred of pixels apart from the true disparity. However, the intervals are still correct, although they are necessarily very large.

The evaluation of those different metrics indicates that our method seems to perform well for estimating disparity confidence intervals. An ablation study on the influence of parameters is conducted in the Annex.

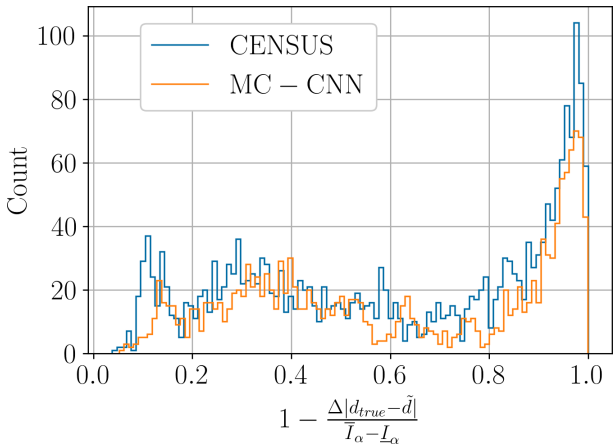
We will use the method presented in this chapter for creating elevation intervals in Chapter 6. However, it is worth noting that this method is not set in stone, and that other approaches could build upon it to improve its accuracy and size performance. For instance, we regularized intervals in low confidence areas using a quantile approach. Another solution could have been to use a different cost function in those areas, one that performs better near depth discontinuities and is less sensitive to the adherence effect. We also computed intervals after SGM regularization, but using the information contained in the cost volume both before and after SGM regularization could lead to more accurate intervals. Those leads were not considered in the context of this thesis, but are considered for future work.



(a) Distribution of the relative error of intervals used to compute ε in Equation (5.16).

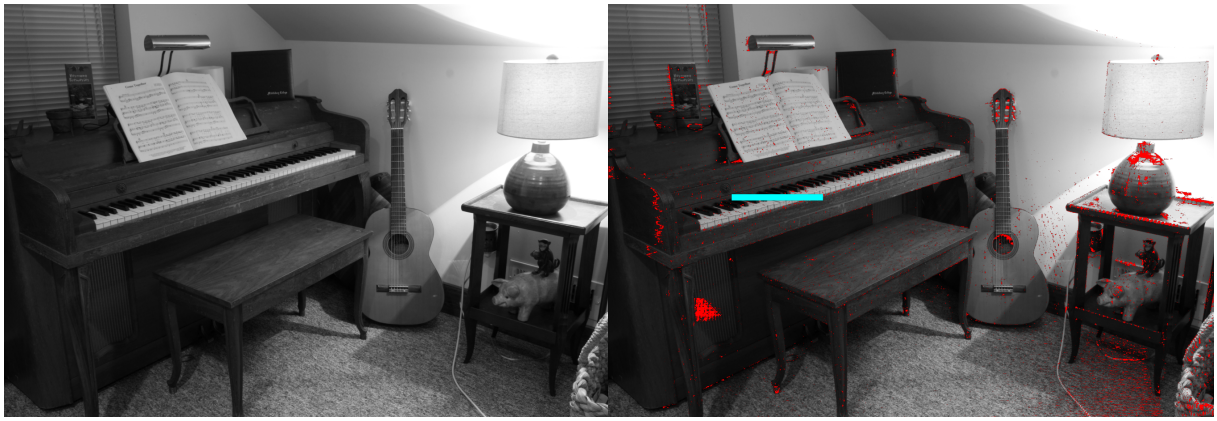


(b) Distribution of the relative size of intervals used to compute s_{rel} in Equation (5.17).

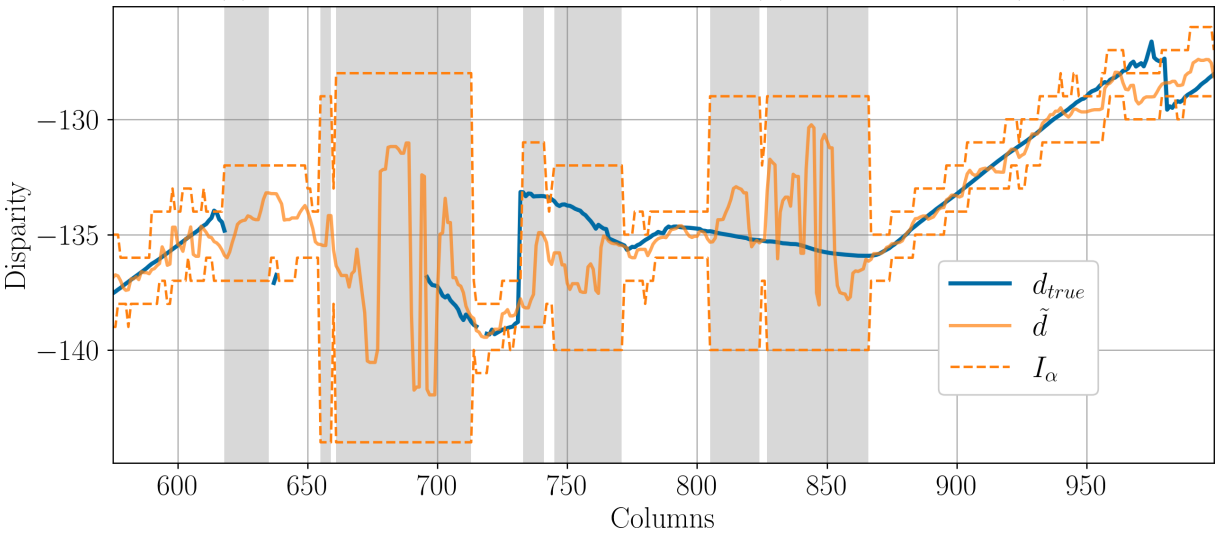


(c) Distribution of the relative over-estimation of intervals used to compute o_{rel} in Equation (5.18).

Figure 5.25: Distribution (in number of pixels) of the relative error (a), relative size (b) and relative over-estimation (c) of intervals for the Middlebury Cones dataset. ε from Equation (5.16) is the median of the distribution in (a). s_{rel} from Equation (5.17) is the median of the distribution of (b). o_{rel} from Equation (5.18) is the median of the distribution of (c).

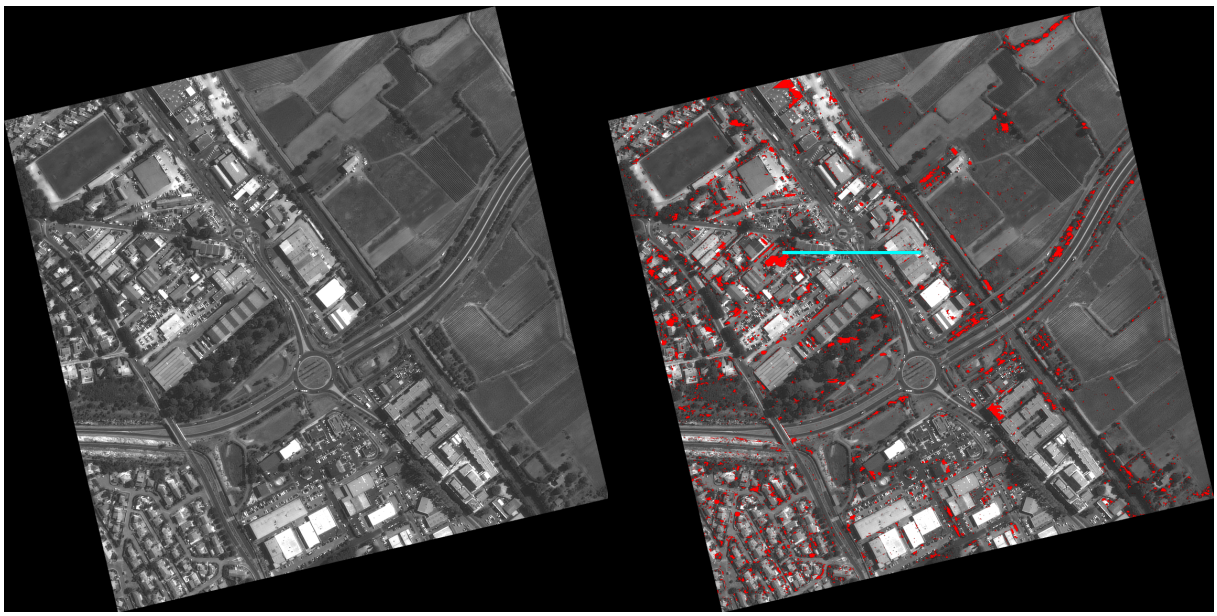


(a) Left Image (b) Wrong intervals (red)

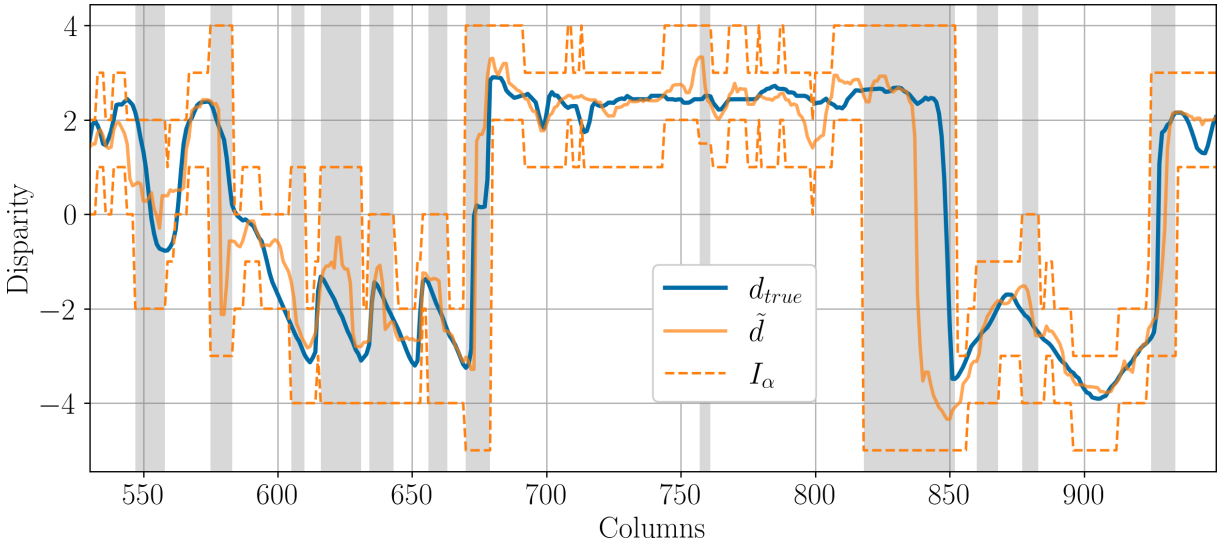


(c) True disparity, predicted disparity and intervals along the blue line of (b). Gray areas represent low confidence areas.

Figure 5.26: "Piano" scene from Middlebury 2014 dataset.



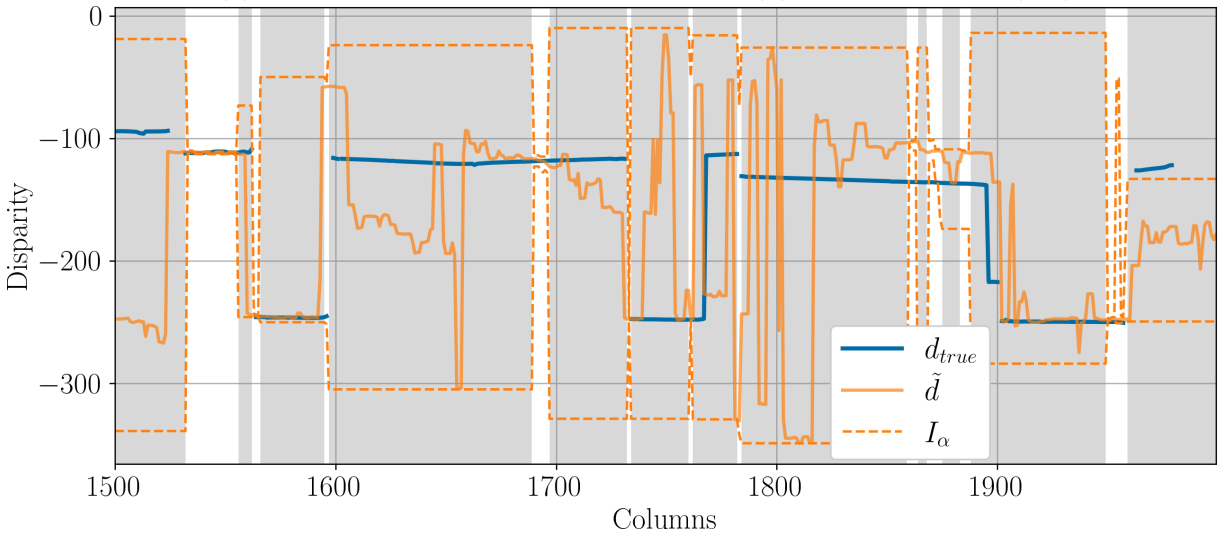
(a) Left Image (b) Wrong intervals (red)



(c) True disparity, predicted disparity and intervals along the blue line of (b). Gray areas represent low confidence areas.

Figure 5.27: Scene from the Montpellier dataset.

(a) Left Image (b) Wrong intervals (red)



(c) True disparity, predicted disparity and intervals along the blue line of (b). Gray areas represent low confidence areas.

Figure 5.28: "Sword2" scene from Middlebury 2014 dataset. This scene is an example of bad disparity prediction but correct (although large) confidence intervals.

Conclusion

In this chapter, we developed a method for computing disparity confidence intervals in a dense stereo matching problem. Confidence intervals are correct even when the predicted disparity is far from the ground truth. This method can be applied to any cost-volume based stereo algorithm, but will produce different intervals depending on the cost function used. As our main objective in this thesis is to produce elevation confidence intervals on DSMs, we will therefore propagate disparity intervals into elevation confidence intervals in the next chapter. We will also evaluate whether the intervals preserve their accuracy and size performances when transformed into elevation intervals.

Chapter 6

Producing Elevation Confidence Intervals

In Chapter 5, we presented a method for computing disparity confidence intervals alongside each predicted disparity $\tilde{d}$. In the photogrammetry pipeline, the predicted disparity is triangulated, filtered (Section 1.3.4) and then rasterized (Section 1.3.5) to obtain the final DSM. We will see in this chapter how we can apply those steps to disparity confidence intervals in order to propagate them into elevation confidence intervals, associated with the DSM values. This chapter takes up work and data published in [Malinowski et al., 2024b].

In Section 1.4.1 of Chapter 1, we presented related work where confidence intervals were computed based on a DSM, mainly in [Oksanen, 2006, Panagiotakis et al., 2018, Deschamps-Berger, 2021]. In those studies, a single global confidence interval associated with a DSM is computed *a posteriori*, *i.e.* based solely on the DSM (and reference points), regardless of the method used to obtain it (*i.e.*, photogrammetry, LiDAR, Radar). The methodology presented in this chapter estimates the uncertainty independently for each pixel, leading to small confidence intervals in confident areas, and bigger confidence intervals where the algorithms may have performed badly. We differ in this regard to previous work, and it does not seem relevant to compare our intervals to theirs, as their most similar characteristics is their name "interval", but neither share the same form (single *vs.* multiple intervals), nor are based on the same data.

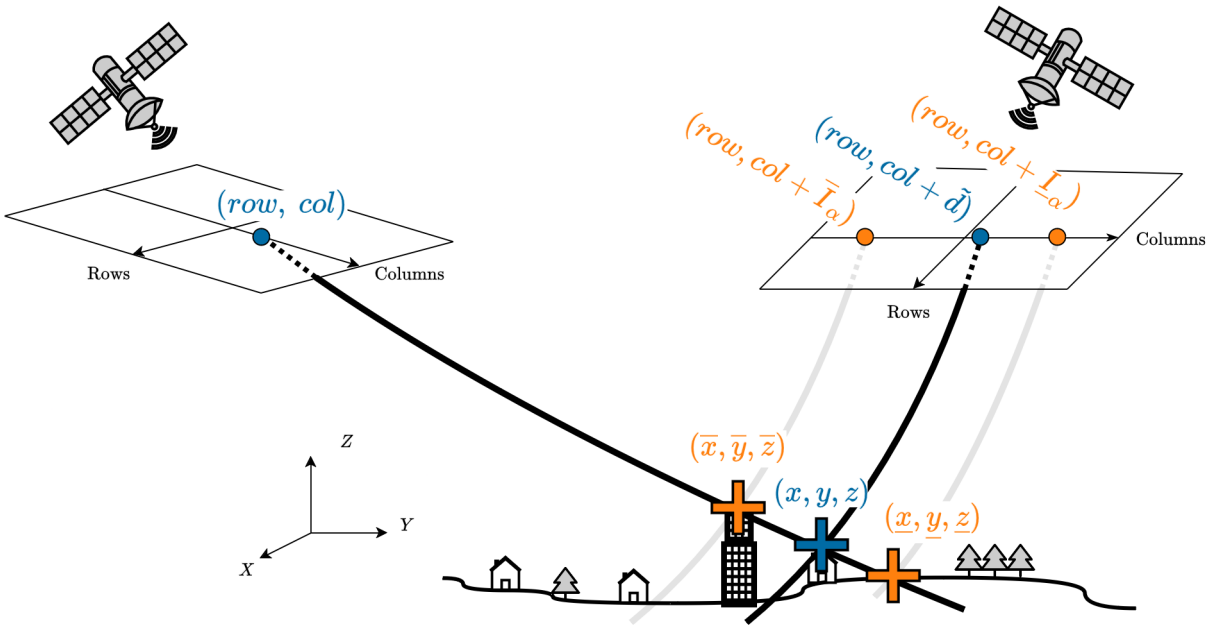


Figure 6.1: Triangulation of the three pairs of lines of sight. The angle between lines of sight is exaggerated for the purpose of this illustration.

6.1 From Disparity Intervals to Elevation Intervals

6.1.1 Triangulation of Disparity Intervals

As indicated by their name, disparity confidence intervals are expressed in pixel of disparities. Disparities are used to determine pairs of intersecting lines of sight from different sensors, which are then triangulated to obtain a 3D point. With disparity confidence intervals, we now have 3 pairs of line of sight for each pixel instead of just one:

- The pair composed using the predicted disparity $\tilde{d}$
- The two pairs composed using the upper and lower disparities from $I_\alpha = [\underline{I}_\alpha,\ \overline{I}_\alpha]$.

Intersecting each pair of line of sight yields 3 3D points, as presented in Figure 6.1. We deduce the first point $(x,\ y,\ z)$ from predicted disparity $\tilde{d}$, and the two other points $(\underline{x},\ \underline{y},\ \underline{z})$, $(\overline{x},\ \overline{y},\ \overline{z})$ are deduced from $\underline{I}_\alpha$ and $\overline{I}_\alpha$ respectively.

Depending on the disposition of satellites, as well as which image is selected as the reference image, it is both possible that $\overline{z} \leqslant z \leqslant \underline{z}$ or $\underline{z} \leqslant z \leqslant \overline{z}$. In the following, and for simplicity, we will consider that $\underline{z} \leqslant z \leqslant \overline{z}$. This is not constraining, as we can just change the notations to unsure this inequality holds. We therefore have a 3D confidence "interval", defined as every point from the reference line of sight between $(\underline{x},\ \underline{y},\ \underline{z})$ and $(\overline{x},\ \overline{y},\ \overline{z})$. For instance in Figure 6.1, the reference line of sight is the one originating from the left satellite, and the confidence "interval" is the portion of this line of sight between the two orange crosses. It is not an interval *per se*, as RPC are polynomials and

not straight lines, but they are approximated by straight lines in the computations, so the distinction is superfluous.

The point cloud obtained from the triangulation of the disparity map is then filtered, as detailed in Section 1.3.4. If a 3D point is filtered, then we naturally also filter its corresponding interval with it.

6.1.2 Rasterization of 3D "Intervals"

The final step of the pipeline is the rasterization, as our objective is to produce elevation confidence intervals associated with every value contained in the DSM. However, rasterizing intervals along lines of sight as it stands raises an issue: we are not guaranteed that the rasterized intervals will remain coherent with the final DSM when projected on the regular grid. A solution to circumvent this issue is to consider that the planimetric shift $\Delta XY = \sqrt{(x - \underline{x})^2 + (y - \underline{y})^2}$ is small in comparison to the altimetric shift $\Delta Z = \sqrt{(z - \underline{z})^2}$ and that we can therefore neglect it. This hypothesis depends only on the incidence angle β of the reference image. Indeed, it holds that $\frac{\Delta Z}{\Delta XY} = \frac{1}{\tan(\beta)}$, where β is the incidence angle as depicted in Figure 6.2. Ideally, if the reference image has no incidence angle, then the ΔXY shift is null. Table 6.2 details the different incidence angles encountered in our experiments. The incidence angles are typically around 10°, *i.e.* ΔZ is around 5.6 times bigger than ΔXY. For the scene in Paris where the incidence reaches 6.1°, the ratio is around 10, while the only scene with 21.3°, near Peyto Lake, Canada, has a ratio around 2.6. For this last acquisition, the hypothesis that ΔXY is small compared to ΔZ is debatable, but we will still neglect ΔXY in the rasterization in order to stay consistent across our experiments. This will also allow verifying if this hypothesis can impact the performance of our method.

The lower and upper bounds can thus be aligned directly above and below the predicted 3D point $(x,\ y,\ z)$, and they thus become:

$$(\overline{x},\ \overline{y},\ \overline{z}) \ \rightarrow \ (x,\ y,\ \overline{z}) \tag{6.1}$$

$$(\underline{x},\ \underline{y},\ \underline{z}) \ \rightarrow \ (x,\ y,\ \underline{z}) \tag{6.2}$$

Figure 6.3 illustrates this modification, where the orange points along the line of sight are shifted in the $(X,\ Y)$ plane to be vertically aligned with $(x,\ y,\ z)$.

We remind here the general formulation of the rasterization step. Let $(x,\ y)$ be a cell of the DSM, and PC be the point cloud considered in the rasterization. The value of the

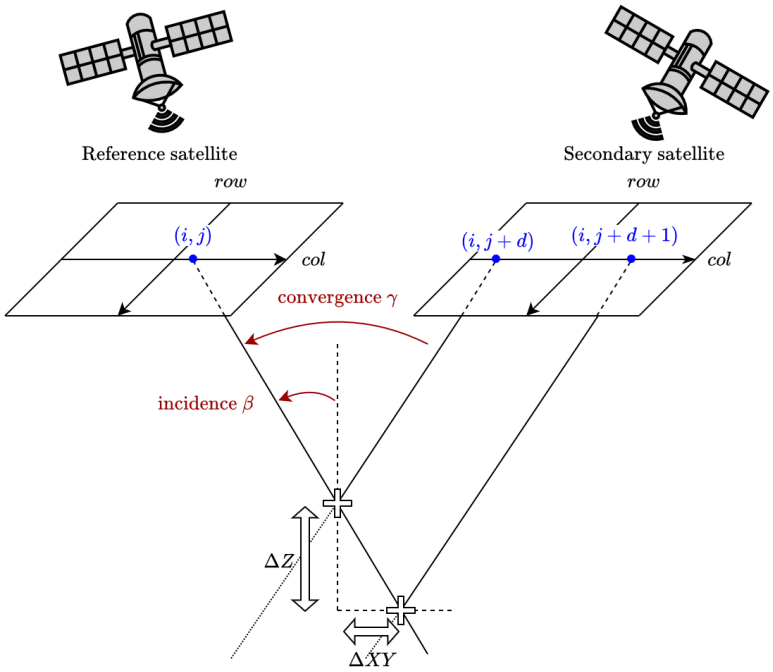


Figure 6.2: Acquisition angles of satellites

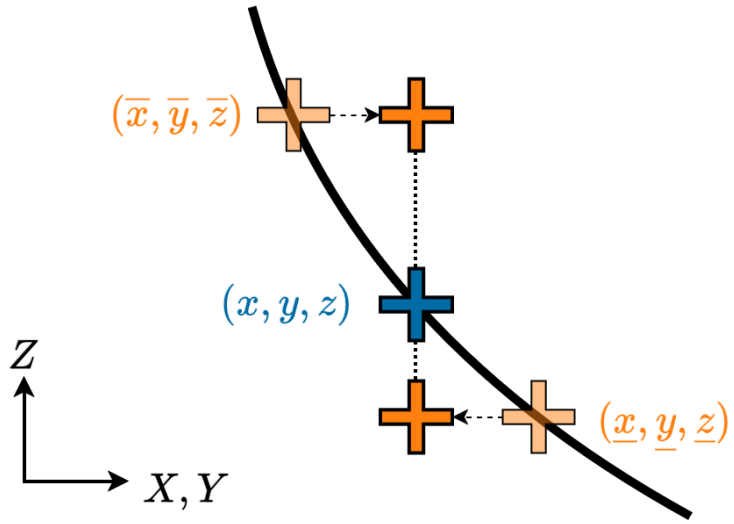


Figure 6.3: Aligning the confidence interval bounds along a line of sight

DSM is computed using a weighted mean of PC:

$$\mathrm{DSM}(x,y) = \frac{\sum\limits_{(x_i,y_i,z_i)\in PC} z_i \cdot w(x_i,y_i)}{\sum\limits_{(x_i,y_i,z_i)\in PC} w(x_i,y_i)} \tag{6.3}$$

where weights $w(x_i, y_i)$ are positive scalars. In CARS, the weights are computed using a Gaussian distribution, but other pipelines also use Inverse Distance Weightings which works similarly. In practice, only the points within a given distance of the center (x, y) of the cell are considered in the mean.

Because it holds that for any point: $\underline{z} \leqslant z \leqslant \overline{z}$, then computing the DSMs independently using $(x,\ y,\ z)$, $(x,\ y,\ \overline{z})$ and $(x,\ y,\ \underline{z})$ will ensure the consistency of the resulting DSMs:

$$\frac{\sum\limits_{(x_i,y_i,\underline{z}_i)\in PC} \underline{z}_i \cdot w(x_i,y_i)}{\sum\limits_{(x_i,y_i,\underline{z}_i)\in PC} w(x_i,y_i)} \leqslant \frac{\sum\limits_{(x_i,y_i,z_i)\in PC} z_i \cdot w(x_i,y_i)}{\sum\limits_{(x_i,y_i,z_i)\in PC} w(x_i,y_i)} \leqslant \frac{\sum\limits_{(x_i,y_i,\overline{z}_i)\in PC} \overline{z}_i \cdot w(x_i,y_i)}{\sum\limits_{(x_i,y_i,\overline{z}_i)\in PC} w(x_i,y_i)}$$

$$\underline{\mathrm{DSM}}(x,y) \leqslant \mathrm{DSM}(x,y) \leqslant \overline{\mathrm{DSM}}(x,y) \tag{6.4}$$

where $\underline{\mathrm{DSM}}(x,y)$ is the DSM computed using points $(x,\ y,\ \underline{z})$, and $\overline{\mathrm{DSM}}(x,y)$ is the DSM computed using points $(x,\ y,\ \overline{z})$. For each cell $\mathrm{DSM}(x,y)$ we have now computed an elevation interval $[\underline{\mathrm{DSM}}(x,y),\ \overline{\mathrm{DSM}}(x,y)]$.

As rasterization is the final step of the stereo pipeline, we now have propagated the confidence intervals all the way to the end of the pipeline while ensuring their coherency with the predicted DSM. Additionally, The production of elevation confidence intervals did not influence the values of the final DSM. We now need to evaluate the elevation confidence intervals on real data to verify if the potential errors occurring during the transformation of disparity intervals into elevation intervals do not question their accuracy.

6.2 Acquiring and Processing Data for Evaluation

6.2.1 Satellite and DSM datasets

This section will present the different images and ground truth DSMs used to evaluate elevation intervals. We will use ground truth DSMs that can be categorized into two categories.

The first category is composed of DSMs acquired over mountainous regions containing glaciers. They have been kindly provided by Etienne Berthier from LEGOS, Liss Marie Andreassen from the Norwegian Water Resources and Energy Directorate (NVE) and

Brian Menounos from the Natural Sciences and Engineering Research Council of Canada and the Tula Foundation (Hakai Institute). The data were acquired in the following regions:

- Two different DSMs in the mountainous region of Jotunheinem, Norway. We will refer to the two DSMs as Hellmem (Figure 6.7) and Graasubreen (Figure 6.6).
- Langfjordjøkelen glacier, Norway (Figure 6.8).
- Mountains near Peyto lake, in the Alberta province of Canada. (Figure 6.12)

The ground truth were acquired using a LiDAR sensor, rasterized at 50 cm resolution. We did not apply the rasterization ourselves and only had access to the rasterized DSMs.

The second category of ground truth data comes from the LiDAR HD program ([Monnet, 2023], `https://geoservices.ign.fr/lidarhd`) which intends to cover the entirety of the French territory (except for French Guiana) by the end of 2026. It is a very important source of information, with around 10 measured points per m^2 and a relative planimetric accuracy of 50 cm. As of the time we write this thesis, data over the French territory are not fully (publicly) available. From the data available at the time of this work, we selected different regions of interest in order to have a variety of landscapes (rural, urban, seaside *etc.*). Those landscapes all presented strong elevation variations in order to present a challenge for the stereo pipeline. We also want to determine if our method behaves differently depending on the nature of the scene. Here is the list of the considered regions:

- The city of Bordeaux (Figure 6.4)
- The city of Montpellier (Figure 6.10)
- The city of Paris (Figure 6.11)
- The city of Toulouse (Figure 6.14)
- Valleys and mountains near Grenoble, in the French Alps (Figure 6.5)
- Mediterranean coastline with the city of Monaco (Figure 6.9)
- Valleys and mountains at Pic du Midi, french Pyrenees (Figure 6.13)

LiDAR point clouds were then rasterized at 50 cm resolution with the same Gaussian rasterization method as in the stereo pipeline. Table 6.1 presents the different acquisition dates and shape of the ground truth DSMs.

We used Pléiades images for producing both DSMs and intervals that will be compared

to the ground truth. We did not directly order the Pléiades acquisitions, as it is costly and hard to synchronize with the acquisition date of the ground truth. From our available catalog, we selected stereo pairs that were acquired as close as possible from the acquisition date of the ground truth, or if multiple months separated the two acquisitions, we rather selected a similar period of the previous/next year to minimize seasonal changes. Table 6.1 allows comparing dates of acquisition of the LiDAR and Pléiades images.

We saw in Section 6.1 that the planimetric and altimetric accuracy depend on the incidence and convergence angles of the lines of sights of the satellites. Table 6.2 presents those angles for the different Pléiades images.

Area	Date for LiDAR	Date of Pléiades	GT Size (0.5m)
Bordeaux	2023-09-15	2022-08-04	6001×6001
Grenoble	2021-09-05	2020-09-17	10001×10001
Hellmem 1	2019-08-27	2019-08-27	7127×7298
Graasubreen 2	2019-08-27	2019-08-27	3912×2880
Langfjordjøkelen	2018-09-01	2018-09-01	5841×3689
Monaco	2021-05-13	2020-08-30	10001×10001
Montpellier	2021-05-28	2021-10-17	8001×8001
Paris	2023-03-03	2023-05-31	10001×10001
Peyto	2016-09-13	2016-09-13	13240×17874
Pic du Midi	2021-10-02	2021-10-16	10001×12001
Toulouse	2022-05-29	2022-06-28	12001×8001

Table 6.1: Acquisition date of Pléiades stereo or tri-stereo images, and the LiDAR ground truth

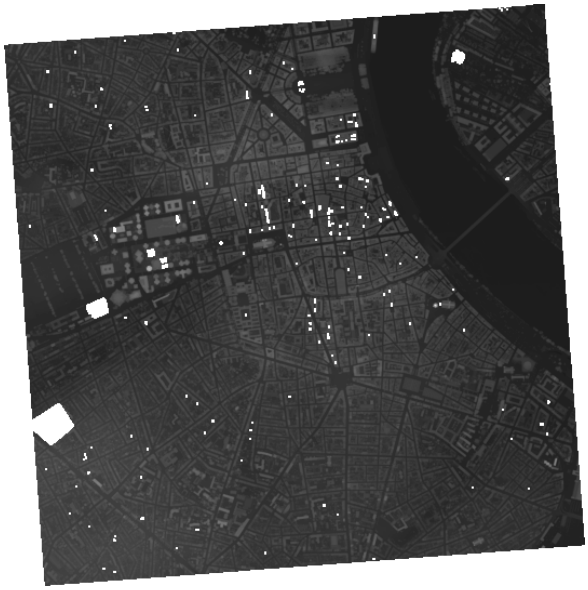
Area	Convergence angle	Left incidence angle	Right incidence angle
Bordeaux	23.1°	9.7°	14.2°
Grenoble	28.3°	12.4°	16.1°
Jotunheinem	22.5°	10.6°	14.5°
Langfjordjøkelen	21.4°	10.2°	13.8°
Monaco	29.2°	12.8°	18.1°
Montpellier	7.7°	11.5°	15°
Paris	4.9°	6.1°	7.9°
Peyto	17.2°	21.3°	21.6°
Pic du Midi	29.0°	13.7°	15.3°
Toulouse	11.4°	12.5°	18.7°

Table 6.2: Relevant angles of stereo pairs of Pléiades images. See Figure 6.2 for a schematic representation of those angles.

6.2.2 Water Masks

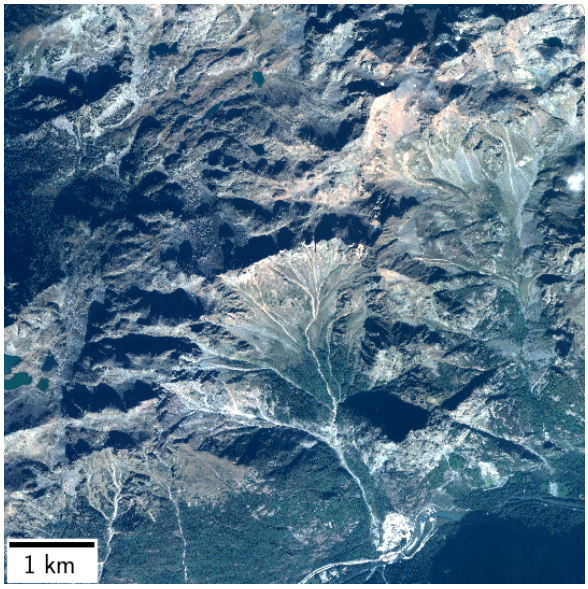
Both LiDAR data and stereo correlation present poor results on water surfaces. For the LiDAR data, wavelengths are often absorbed by still water and do not provide the sen-

(a) RGB image of Bordeaux (Pléiades © CNES 2022, Distribution AIRBUS DS)

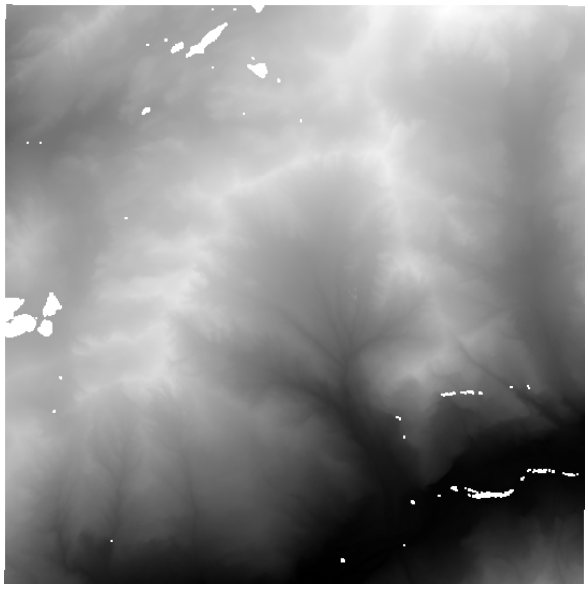


(b) LiDAR HD DSM

Figure 6.4: RGB image of Bordeaux and its associated ground truth DSM.

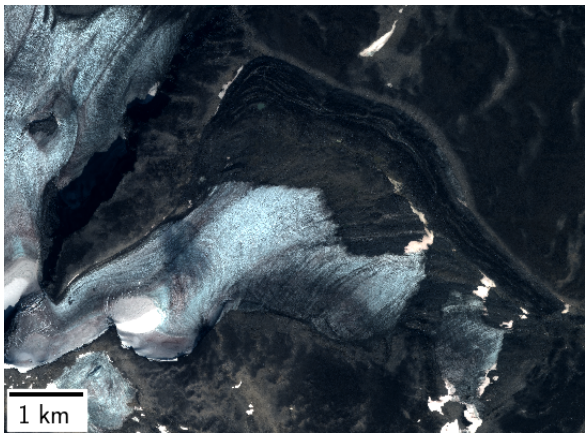


(a) RGB image of a region near Grenoble (Pléiades © CNES 2020, Distribution AIRBUS DS)

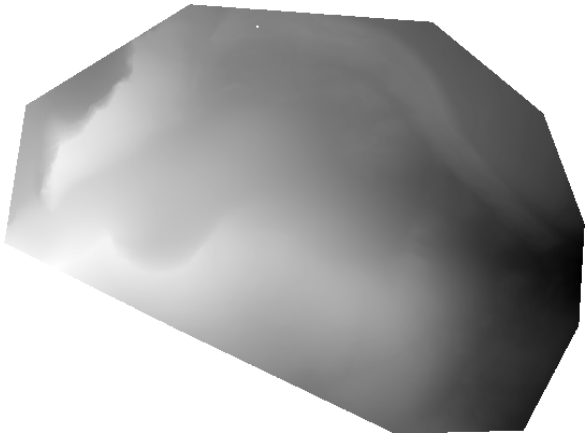


(b) LiDAR HD DSM

Figure 6.5: RGB image of Grenoble and its associated ground truth DSM.

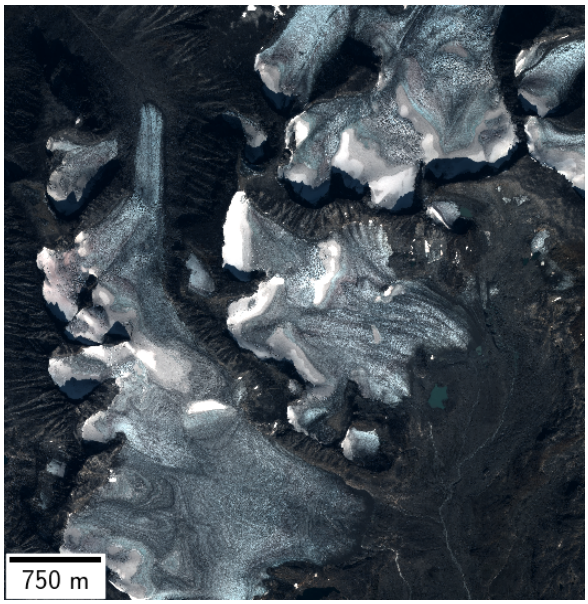


(a) RGB image of Graasubreen (Pléiades © CNES 2019, Distribution AIRBUS DS)

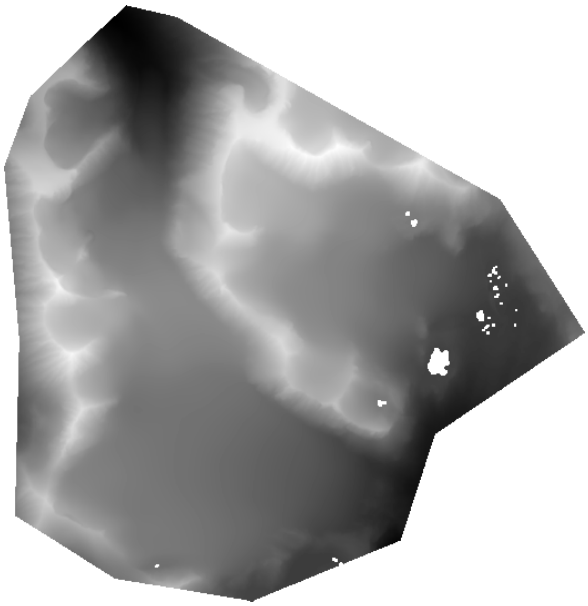


(b) LiDAR DSM

Figure 6.6: RGB image of Graasubreen and its associated ground truth DSM.

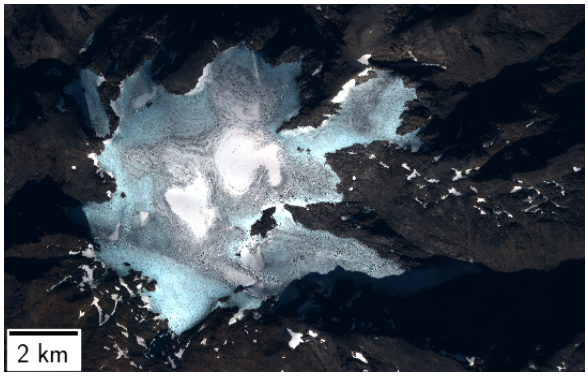


(a) RGB image of Hellmem (Pléiades © CNES 2019, Distribution AIRBUS DS)

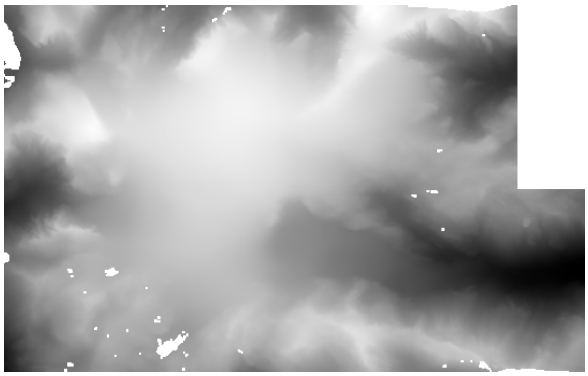


(b) LiDAR DSM

Figure 6.7: RGB image of Hellmem and its associated ground truth DSM.



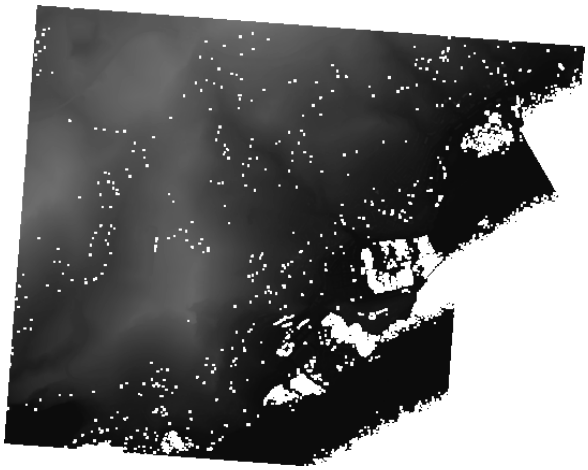
(a) RGB image of Langfjordjøkelen (Pléiades © CNES 2018, Distribution AIRBUS DS)



(b) LiDAR DSM

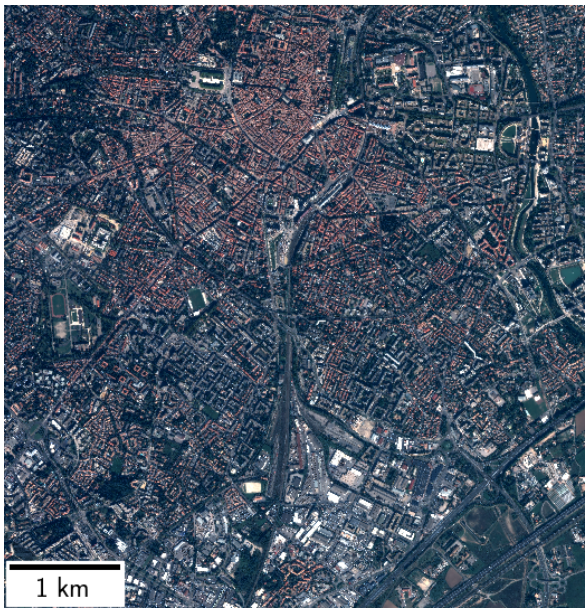
Figure 6.8: RGB image of Langfjordjøkelen and its associated ground truth DSM.

(a) RGB image of Monaco (Pléiades © CNES 2020, Distribution AIRBUS DS)

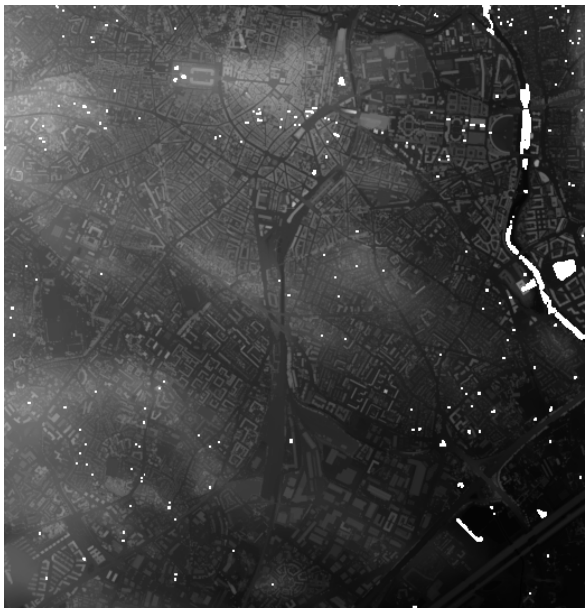


(b) LiDAR HD DSM

Figure 6.9: RGB image of Monaco and its associated ground truth DSM.

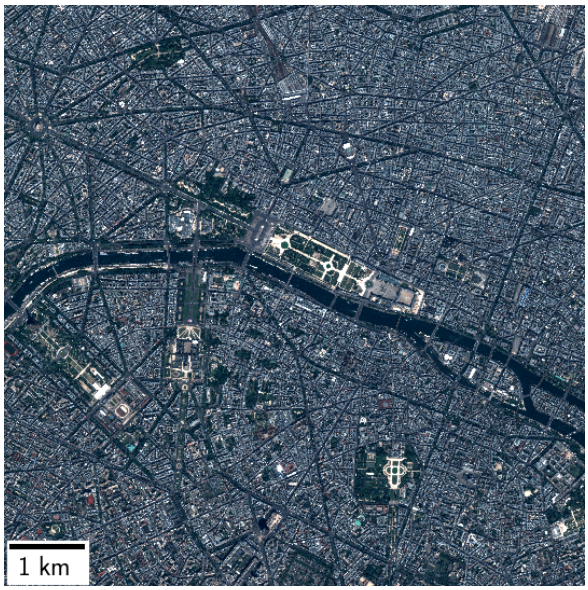


(a) RGB image of Montpellier (Pléiades © CNES 2020, Distribution AIRBUS DS)

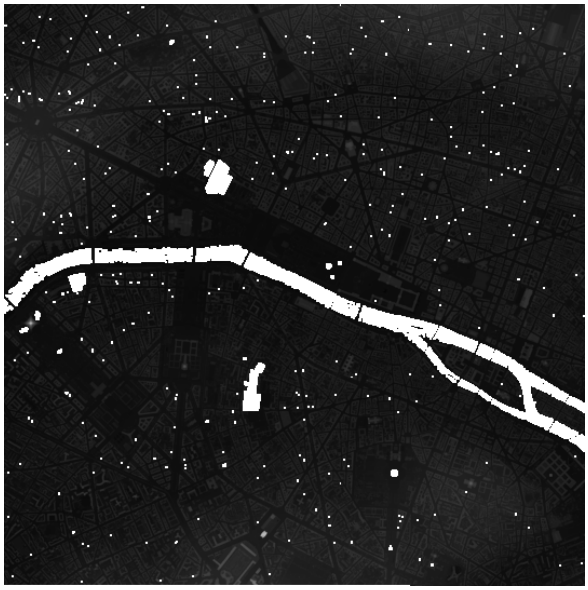


(b) LiDAR HD DSM

Figure 6.10: RGB image of Montpellier and its associated ground truth DSM.

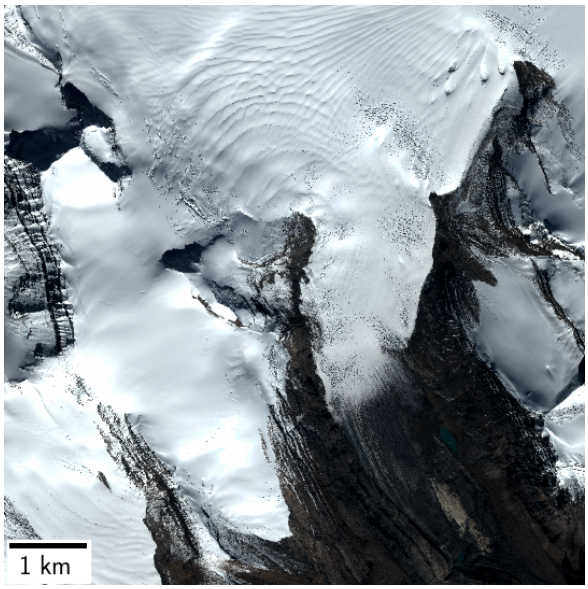


(a) RGB image of Paris (Pléiades © CNES 2023, Distribution AIRBUS DS)

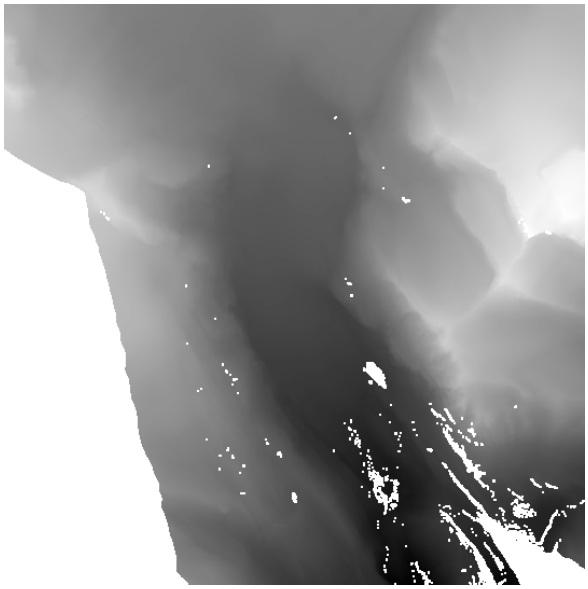


(b) LiDAR HD DSM

Figure 6.11: RGB image of and its associated ground truth DSM.

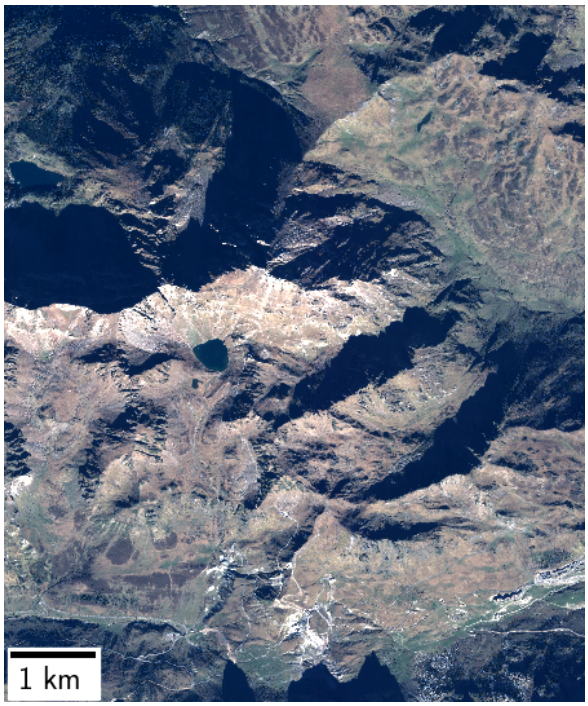


(a) RGB image of Peyto (Pléiades © CNES 2016, Distribution AIRBUS DS)

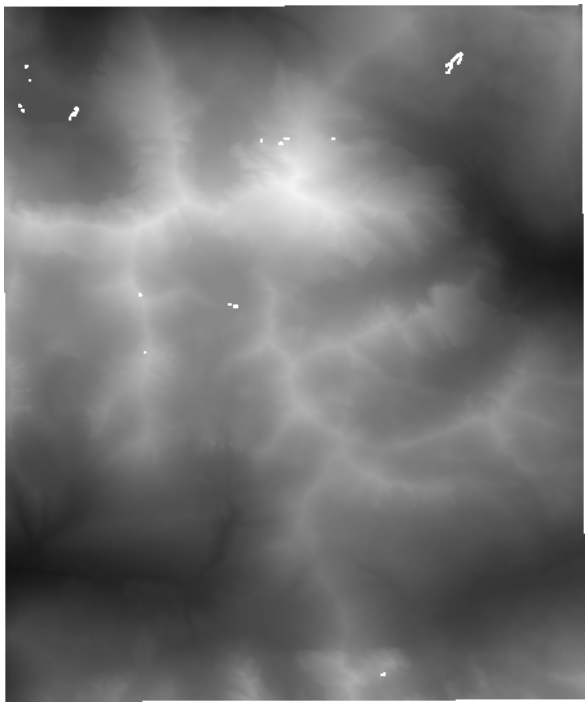


(b) LiDAR HD DSM

Figure 6.12: RGB image of Peyto and its associated ground truth DSM.

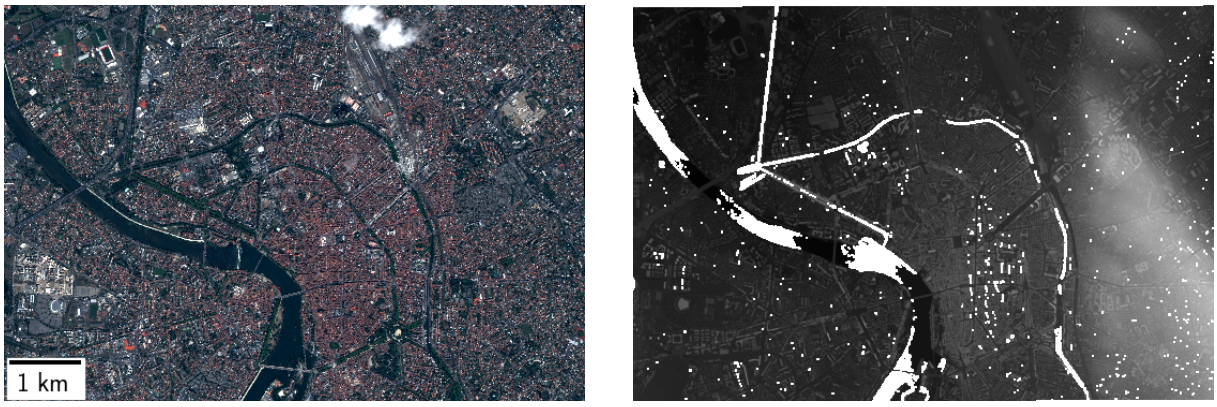


(a) RGB image of Pic du Midi (Pléiades © CNES 2021, Distribution AIRBUS DS)



(b) LiDAR HD DSM

Figure 6.13: RGB image of Pic du Midi and its associated ground truth DSM.



(a) RGB image of Toulouse (Pléiades © CNES 2022, Distribution AIRBUS DS)

(b) LiDAR HD DSM

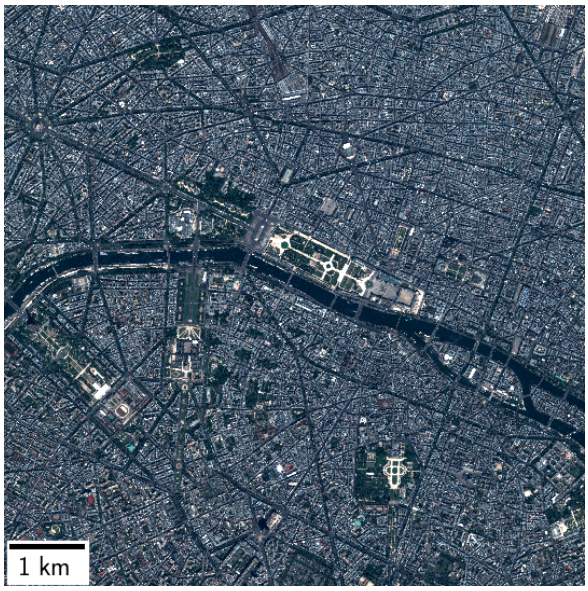
Figure 6.14: RGB image of Toulouse and its associated ground truth DSM.

sor with any feedback (although turbid water can usually send back some signal). For moving waters, the provided signal is often noisy and needs post-processing to remove artifacts. Stereophotogrammetry has usually even poorer results, because open waters are texture-less or uniform surfaces on which the correlation does not perform well. Furthermore, for Pléiades acquisitions, the water may have moved between images, which prevents water pixels from being correctly triangulated. For those reasons, we chose to add a water mask on images of cities crossed by a river (Bordeaux, Monaco, Montpellier, Paris, and Toulouse) or near the seaside (Monaco). The water mask was obtained using an algorithm developed by CNES, which uses a random forest trained on existing high resolution mapping of surface waters [Pekel et al., 2016] and the Normalized Difference Water Index [Gao, 1996]. Figure 6.15(b) presents a water mask produced on the image of Paris.

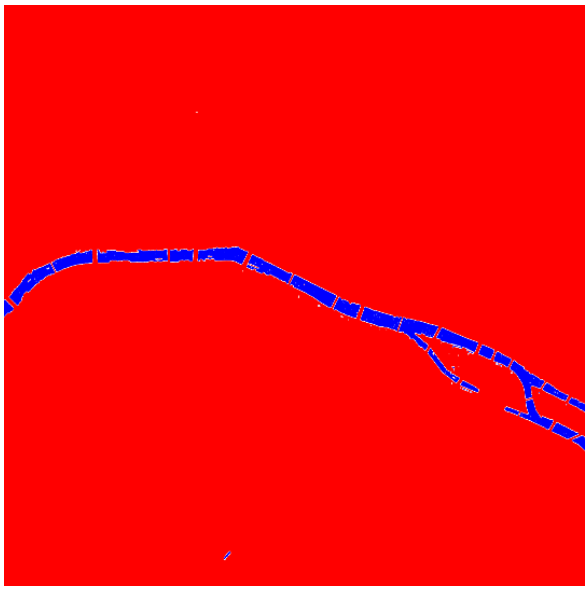
6.2.3 Co-registration

The DSMs obtained from LiDAR data and from stereophotogrammetry are not necessarily in the same projection reference system, and do not use the same altitude reference. Moreover, due to the limited accuracy of RPC models and GPS measures of the LiDAR, some planimetric and elevation biases may exist between the two DSMs. Those biases do not allow their comparison as such. We therefore reproject the ground truth data in the same reference system as their corresponding DSM produced by the CARS pipeline. We then rectify the planimetric and altimetric biases by applying the method presented in [Nuth and Kääb, 2011]. This process is called co-registration. We quickly present the method for estimating the altimetric and planimetric biases, and refer to the original publication for additional details.

It is possible to observe that the measured differences between two shifted DSMs vary



(a) RGB image of Paris (Pléiades © CNES 2023, Distribution AIRBUS DS)



(b) Water mask

Figure 6.15: RGB image of Paris and its associated water mask. Water is indicated by blue pixels

with the slope angle and the orientation of the slope. The different parameters influencing the measured variations of elevation, presented in Figure 6.16, are the following:

- We note sl the angle of a slope. $sl = 0°$ corresponds to a flat slope and $sl = 90°$ to a vertical slope.
- We note ψ the azimuth of the slope. If the slope faces north, then $\psi = 0°$. If it faces west, $\psi = 90°$ *etc.*
- The direction of the planimetric shift between DSMs is given by an angle β, with the same conventions as the azimuth. The magnitude of the shift is B.
- dh refers to measured local variations of elevation, and $\tilde{dh}$ is the global elevation shift between DSMs.

The relation linking all those parameters is the following [Nuth and Kääb, 2011]:

$$dh = B\cos(\psi - \beta)\tan(sl) + \tilde{dh} \tag{6.5}$$

The three unknowns are B, β and $\tilde{dh}$, as the slope parameters can be computed by any GIS software from the DSMs. For instance, the slope of DSM_{true} is computed as:

$$sl(x,y) = \frac{1}{8}\sqrt{\left(\frac{\text{DSM}_{true} * k_x(x,y)}{r_x}\right)^2 + \left(\frac{\text{DSM}_{true} * k_y(x,y)}{r_y}\right)^2} \tag{6.6}$$

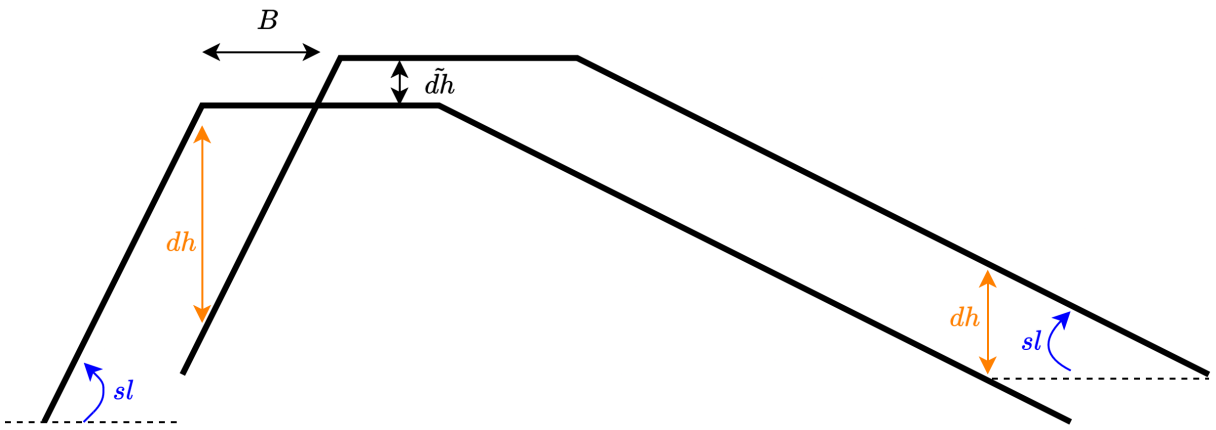


Figure 6.16: Planimetric shift of magnitude B and altimetric shift $\tilde{dh}$ between two DSMs. We can see that local variations $\tilde{dh}$ vary depending on the slope sl. This diagram is in 2D, the angle of the planimetric shift and the azimuth of the slope are therefore not represented.

where $*$ denotes a convolution with two kernels $k_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$, $k_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$, and r_x, r_y are the resolution in x and y. We will use the slope in the evaluation of the different metrics.

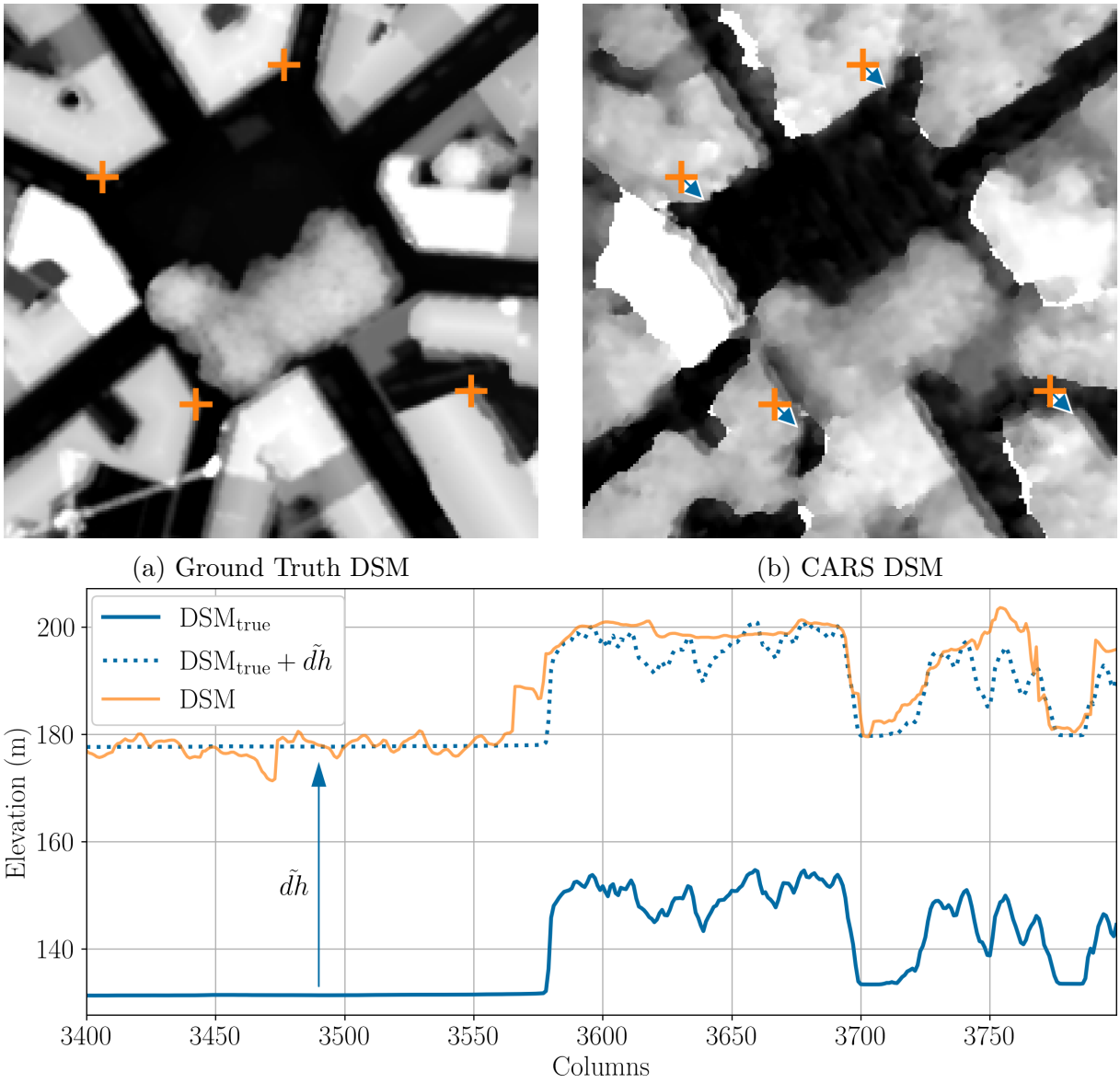
The unknowns B, β and $\tilde{dh}$ are determined by a least square optimization problem. Because the DSM is not expressed analytically, the optimization is not guaranteed to be exact. Multiple iterations of the planimetric and altimetric shift estimation lead to a better final result. An illustration of the co-registration process is presented in Figure 6.17.

The problem we encounter is that the DSM obtained from photogrammetry already possess some errors. In the least squared minimization problem, the residuals computed from the difference between the ground truth DSM and the photogrammetry DSM therefore also contain those errors. This deteriorates the quality of the co-registration. However, it remains the best solution for co-registering DSMs. We apply this co-registration to our data, before computing any metric. The different metrics we consider will be presented in Section 6.3, while the following sections details the parameters used in our stereo pipeline.

6.2.4 Configuration of the Photogrammetry Pipeline

This section provides technical information about the configuration of the CARS pipeline used to process the satellite images.

We used the Copernicus DEM with a 30m resolution as the reference altitude, as the SRTM elevation models were not available for high latitudes, such as Norway for instance.



(a) Ground Truth DSM

(b) CARS DSM

(c) Elevation along a row of the DSM

Figure 6.17: Planimetric shift and altimetric shift from the co-registration step over the city of Toulouse. In (a) and (b), reference points in orange are located at the same row and columns in the two DSMs. The planimetric shift is indicated with blue arrows. (c) presents the altimetric shift between the two DSMs.

We used the same range of considered disparity for every image: [-50pix, 50pix]. This range is quite high, especially for acquisition with a high altitude ratio r_{alt}, but ensures we are not limited to a restrictive range of possible elevations.

Although we developed a method for creating disparity intervals which works for both the CENSUS and MC-CNN cost functions, we will only present results using the CENSUS cost function. Two reasons motivate this choice. First, due to the quantity of scenes and the variety of analyses completed, it would quickly become quite overwhelming to present each result for both cost functions, especially when there is no major difference between the two. Secondly, we observed in the previous chapter that disparity intervals obtained using CENSUS were less accurate than those using MC-CNN. As we favor accuracy above interval size, we chose to use the CENSUS cost function throughout our results, as validating the accuracy requirements for CENSUS ensures that they are also validated using MC-CNN. We verified that there was no unexpected behavior for MC-CNN intervals across scenes, and since nothing notable appeared, we did not delve into so many details as we did for the CENSUS intervals.

We detail here the list of the parameters used in the pipeline. The CENSUS cost function was computed using a 5×5 window. The SGM penalties used for regularizing the CENSUS cost volume were $P_1 = 8$ and $P_2 = 32$. Disparity intervals were produced using a possibility threshold of 0.9. Intervals were regularized in low confidence areas for which the confidence from ambiguity was below 0.6, after being minimized by a minitive kernel of shape $(1, 2 \cdot k_{amb} + 1) = (1, 5)$. For each regularized pixel, we took the 90^{th} upper and 10^{th} lower quantiles over a low confidence area that extended $n_{\mathfrak{N}} = 3$ rows above and below. A V-fit refinement step was used to get sub-pixel disparities, and a median filter of shape (3×3) was applied to the resulting disparity map. Pixels that did not validate the cross-checking criterion from Equation (1.14) were not considered for triangulation. Triangulated 3D points were filtered using Equation (1.23) with $k = 5$ and $N = 50$, and Equation (1.24) with $D_{max} = 3$ m and $N_{min} = 50$. The rasterization from Equation (1.25) used $\sigma = 0.3$ m and $r = 3$ m.

6.3 Metrics for Evaluating Elevation Intervals

We introduce here the metrics used to evaluate the performance of elevation intervals. In order to stay consistent with Section 5.2.1, we consider the metrics introduced for disparity intervals and adapt them to elevation intervals.

We will refer to the true elevation as DSM_{true}, the predicted elevation as DSM and the elevation intervals as [$\underline{\text{DSM}}$, $\overline{\text{DSM}}$].

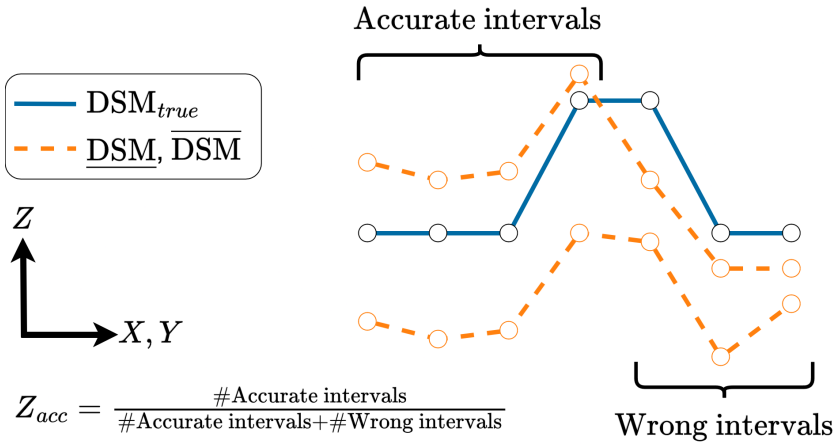


Figure 6.18: Schematic representation of Z_{acc}

Elevation Accuracy Metric

Similarly to Section 5.2.1, the first metric is the proportion of correct intervals, *i.e.* the proportion of intervals containing the ground truth. We call this metric the elevation accuracy Z_{acc}:

$$Z_{acc} = \frac{\#\{\text{DSM}_{true} \mid s.t.\ \text{DSM}_{true} \in [\underline{\text{DSM}},\ \overline{\text{DSM}}]\}}{\#\{\text{DSM}_{true}\}} \tag{6.7}$$

Figure 6.18 illustrates what this metric represents. Z_{acc} is similar to acc defined in Equation (5.15) from Section 5.2.1, we thus want to maximize Z_{acc}. We also keep the objective of 90% accuracy for our method, which was validated for the disparity intervals.

Residual Elevation Error Metric

We are also interested in evaluating the magnitude of the errors. We thus define the residual elevation error Z_ε for all intervals that do not contain the ground truth as:

$$Z_\varepsilon = \frac{1}{r_{alt}} \cdot \text{median}\left(\min(|\text{DSM}_{true} - \overline{\text{DSM}},\ |\text{DSM}_{true} - \underline{\text{DSM}}|)\right) \tag{6.8}$$

where r_{alt} is the disparity to altitude ratio (or altimetric ratio), *i.e.* the elevation difference resulting from a shift of one disparity. Figure 6.19 illustrates what this metric represents. We want Z_ε error to be as close to 0 as possible. r_{alt} has been computed alongside the epipolar grids (Section 1.3.2). Dividing by the ratio r_{alt} effectively converts $\min(|\text{DSM}_{true} - \overline{\text{DSM}}|, |\text{DSM}_{true} - \underline{\text{DSM}}|)$ from meters to pixels. As an error of one disparity pixel in the dense matching step can result in an elevation error of 1m in a scene, and 2m in another, we divide by r_{alt} in order to be able to compare stereo pairs with different convergence angles. Table 6.3 contains the different altimetric ratios for the data we consider. Z_ε is similar to ε defined in Equation (5.16) from Section 5.2.1.

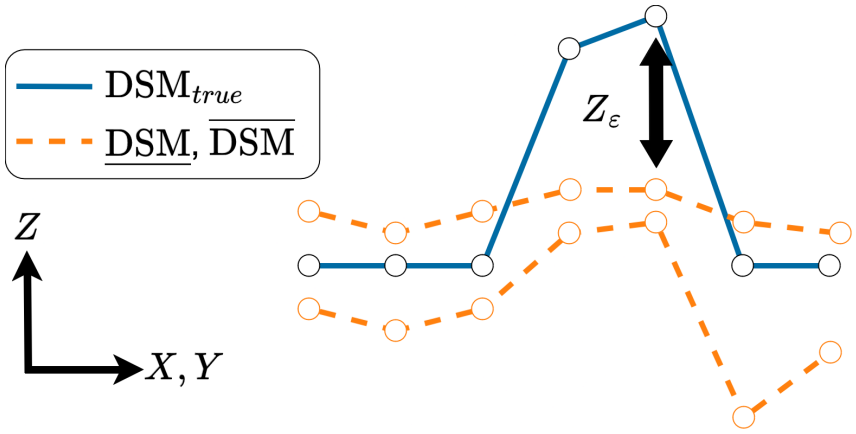


Figure 6.19: Schematic diagram of the quantity represented by Z_ε

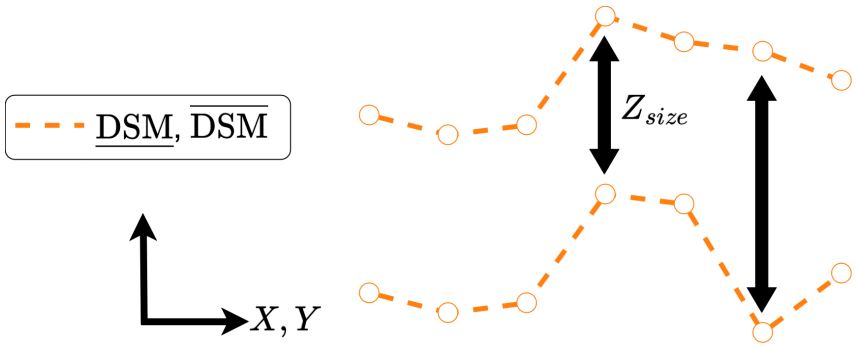


Figure 6.20: Schematic diagram of the quantity represented by Z_{size}

Relative Elevation Size Metric

Those two previous metrics allow measuring the accuracy and errors of elevation intervals. To evaluate the size of intervals, we use the relative elevation size Z_{size} defined as :

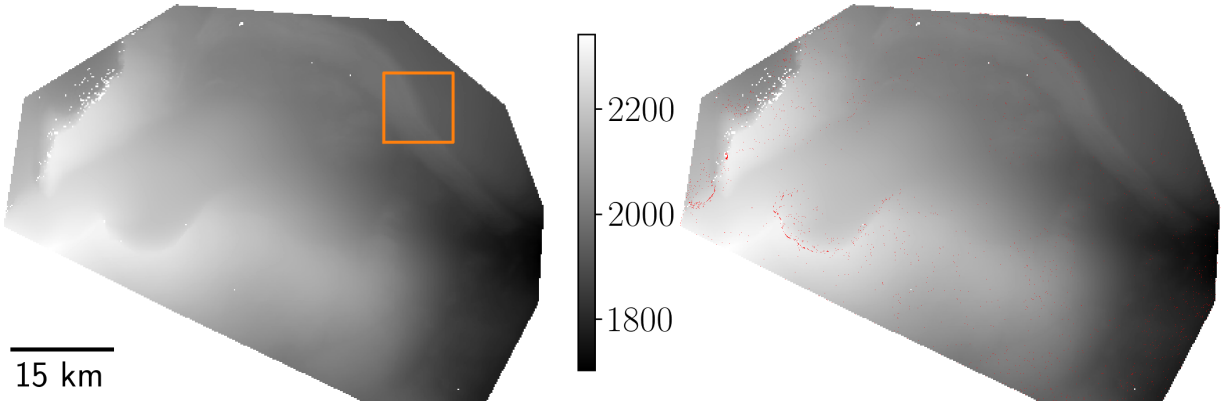
$$Z_{size} = \frac{1}{r_{alt}} \text{median} \left(\overline{\text{DSM}} - \underline{\text{DSM}} \right) \tag{6.9}$$

We divide by the altimetric ratio for the same reasons as in Z_ε, *i.e.* to compare stereo pairs with different convergence angles. Figure 6.20 illustrates what this metric represents. We want the relative elevation size to be as close to zero as possible. Z_{size} is similar to s_{rel} defined in Equation (5.17) from Section 5.2.1.

> **Remark:** In the rasterization process, 3D points from low confidence area are mixed with points from high confidence areas. We therefore are unable to define interval regularization areas in the final DSM. The relative over-estimation o_{rel} has therefore no equivalent for elevation intervals.

Scene	Z_{acc}	Z_{ε} (pix)	Z_{size} (pix)	r_{alt} (m/pix)	invalid
Bordeaux	89.3%	0.56	4.18	1.24	21.5%
Graasubreen	99.7%	0.12	1.96	1.32	27.5%
Hellmem	98.8%	2.76	2.02	1.32	34.7%
Grenoble	93.1%	0.87	3.99	1.06	6.4%
Langfjordjøkelen	99.4%	0.30	2.16	1.35	13.9%
Monaco	90.3%	0.57	3.76	0.99	41.9%
Montpellier	89.1%	0.28	2.05	3.64	1.9%
Paris	84.6%	0.46	2.01	5.78	3.0%
Peyto	98.9%	0.18	2.08	1.66	18.3%
Pic du midi	98.1%	0.20	2.41	1.01	7.5%
Toulouse	92.0%	0.38	2.04	2.43	6.9%

Table 6.3: Elevation metrics for the different stereo pairs. The last column indicates the proportion of invalid pixels in the considered DSMs.



(a) Photogrammetry DSM. Orange area is detailed in Figure 6.22

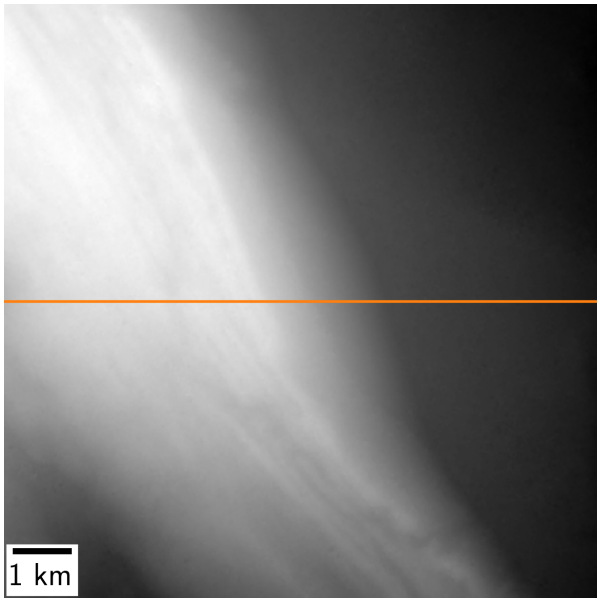
(b) DSM with wrong intervals in red.

Figure 6.21: DSM without and with wrong intervals over Graasubreen scene

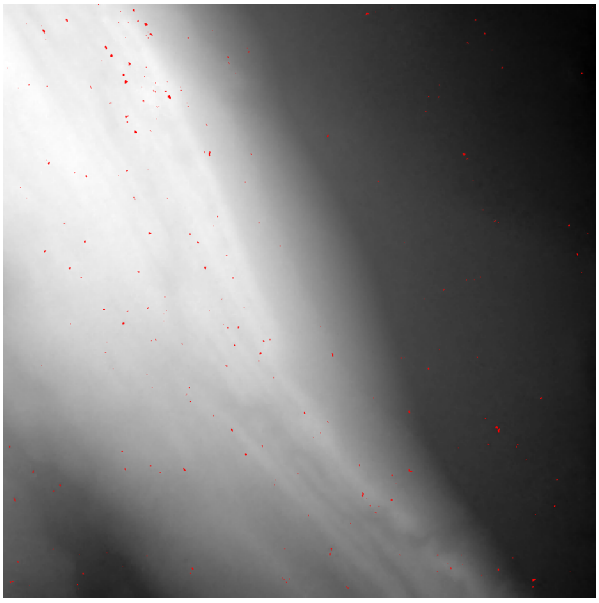
6.4 Elevation Intervals Results

The different metrics have been evaluated between the photogrammetry DSM and the associated ground truth DSM. Results are presented in Table 6.3, and are commented in the following sections. We also displayed for informational purposes the proportion of invalid pixels of each scene, in the last column of the table. A pixel is invalid if the ground truth DSM or predicted DSM does not contain any data for this pixel, or if it was masked by a water mask. Invalid pixels are therefore pixels that were not considered when computing the metrics.

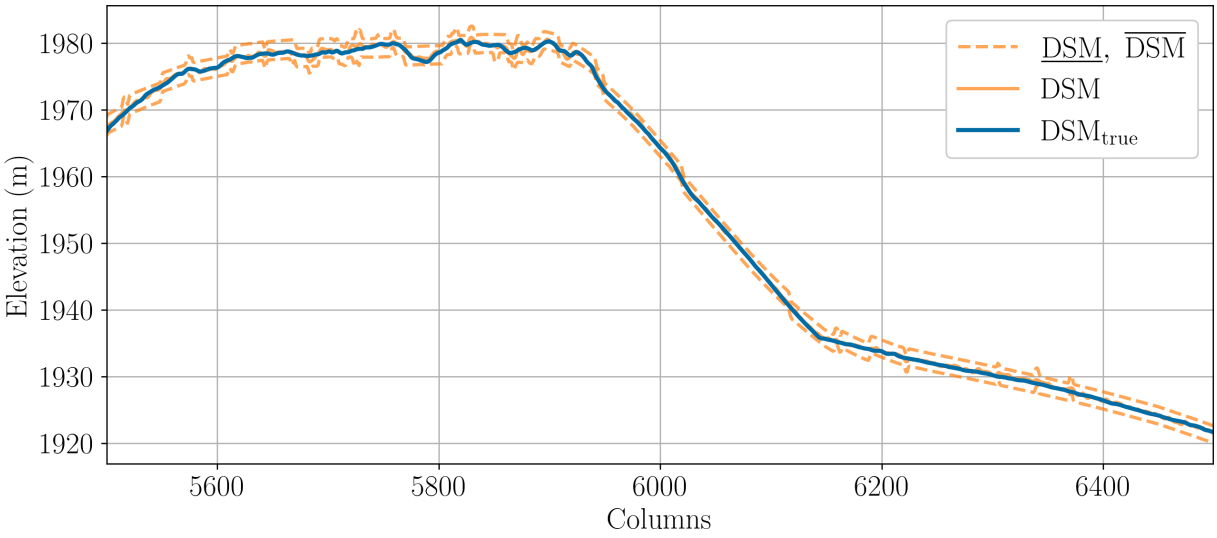
Figures 6.21 to 6.27 allow to qualitatively evaluate the performance of the elevation intervals in different scenes. We can see that intervals seem to have a better accuracy performance in rural scenes such as glaciers than in urban scenes. For instance, we can clearly see wrong intervals of the Toulouse scene in Figure 6.23(b). They are harder to



(a) Photogrammetry DSM. Orange line is detailed in (c)

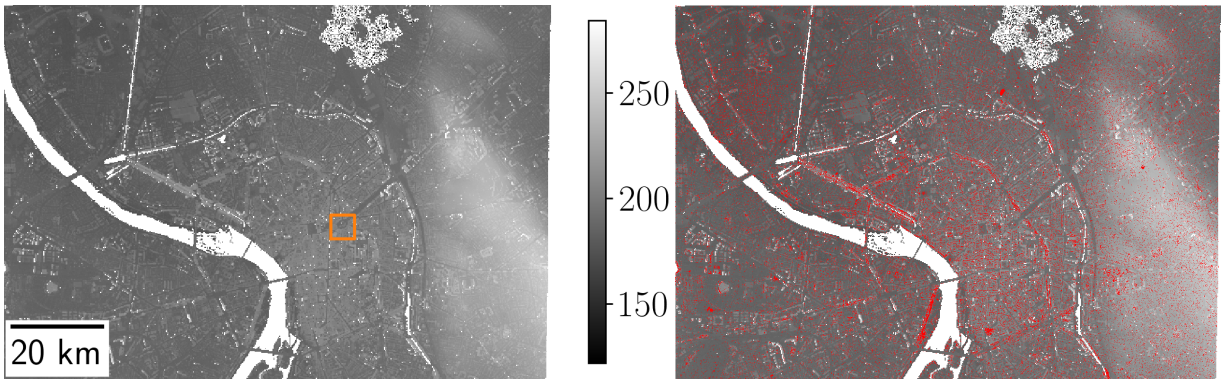


(b) DSM with wrong intervals in red.



(c) DSM, ground truth and elevation intervals along the orange line of (a)

Figure 6.22: Details of the Graasubreen DSM. This area corresponds to the orange square from Figure 6.21(a).



(a) Photogrammetry DSM. Orange area is detailed in Figure 6.24

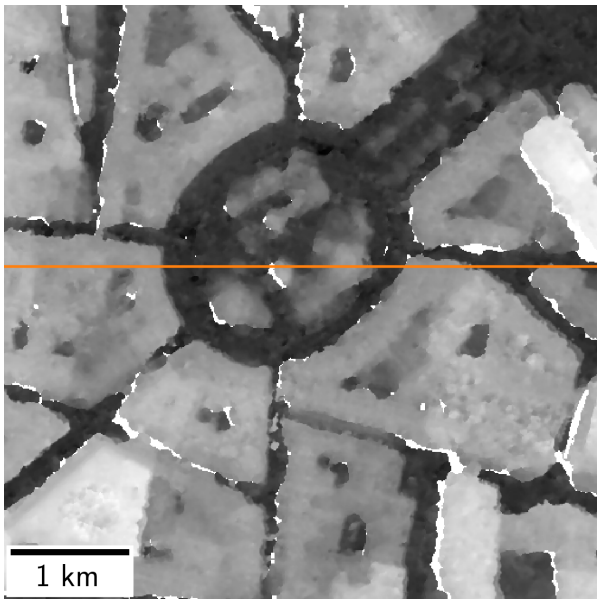
(b) DSM with wrong intervals in red.

Figure 6.23: DSM without and with wrong intervals over Toulouse scene

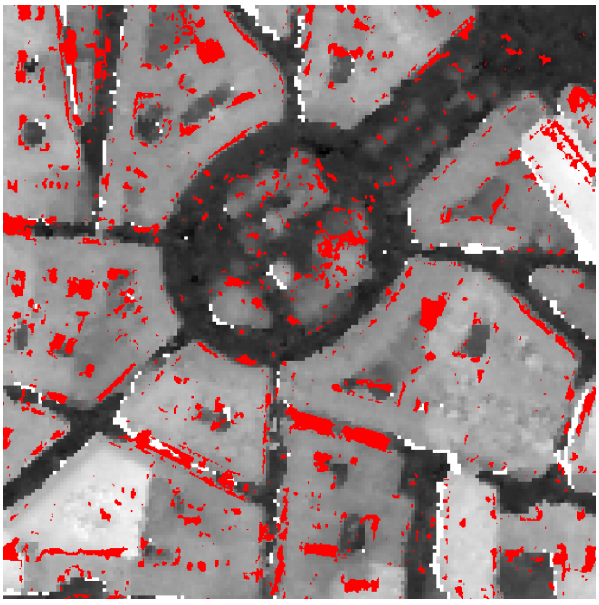
distinguish in the Graasubreen scene in Figures 6.21(b) and 6.22(b). Another notable observation is that elevation intervals of the Graasubreen glacier in Figure 6.22(c) are very small and do not present large size fluctuations along the considered cross-section. This is not the case for the Toulouse scene (Figure 6.24(c)), where intervals present large variations in size. They are relatively small from columns 7000 to 7100, but their size increase from column 7400 to 7500. We can see on the right of Figure 6.24(c), near column 7450, that their size allow to correctly contain the ground truth despite the difference between DSM and DSM_{true}. However, there are some very large intervals, sometimes reaching 80 m in size, near columns 7150, or 7300. Those intervals can occur because the range of considered disparities in the dense matching step is $[-50, 50]$, for every scene. Converted into altitude, elevations intervals have a maximal size of $[-50 \cdot r_{alt}, 50 \cdot r_{alt}]$, centered around the reference altitude (which can vary across the scene). This means that we can have a maximal elevation interval of size 243m in the case of Toulouse, where $r_{alt} = 2.43$. The previously observed intervals of size 80m in Figure 6.24(c) thus originates from a disparity interval of 33 pixels in size, *i.e.* a third of the disparity intervals. Other examples of large intervals include the Monaco scene where $r_{alt} = 0.99$ and for which the largest interval has a size of 99 m, or the Paris scene where $r_{alt} = 5.78$ and the largest interval has a size of 578m. We will see in section 6.8 that we can modify our method to filter out points leading to those intervals. For now, we will evaluate the method as it stands. The following sections are dedicated to discussing each metric individually.

6.4.1 Elevation Accuracy

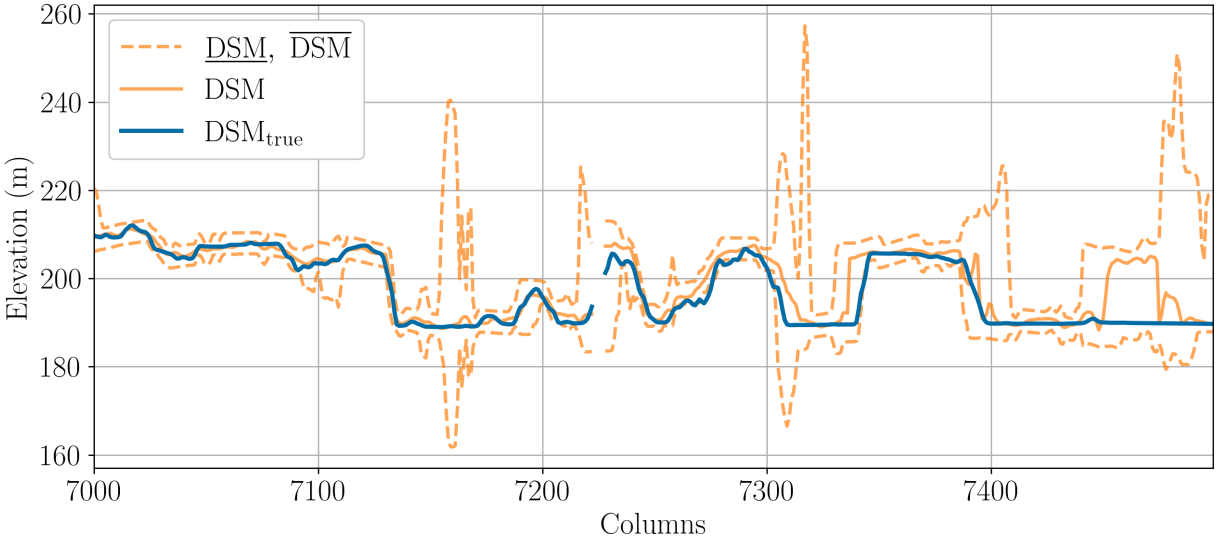
The elevation accuracy Z_{acc}, defined in Equation (6.7), measures the proportions of intervals which contains the ground truth elevation, which we want to be as high as possible. The first observation is that the 90% accuracy objective is validated on 8 of the 11 scenes we considered. In particular, the different glaciers as well as the Pic du Midi elevation



(a) Photogrammetry DSM. Orange line is detailed in Figure 6.24(c)

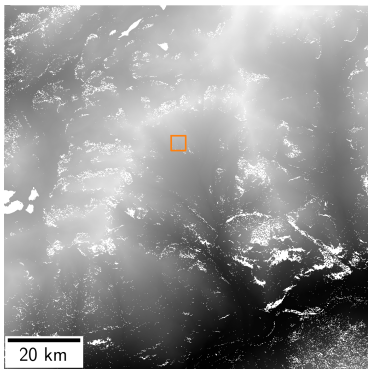


(b) DSM with wrong intervals in red.

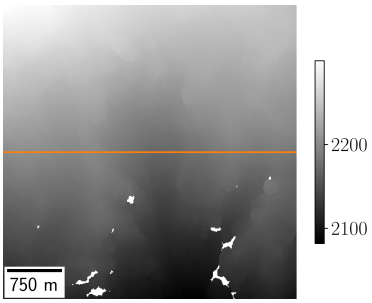


(c) DSM, ground truth and elevation intervals along the orange line of Figure 6.24(a)

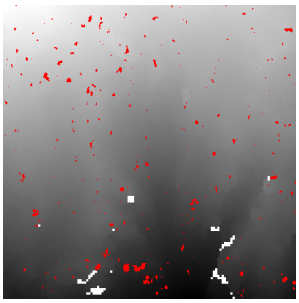
Figure 6.24: Detailed DSM over Wilson square, Toulouse. This area corresponds to the orange square from Figure 6.23(a).



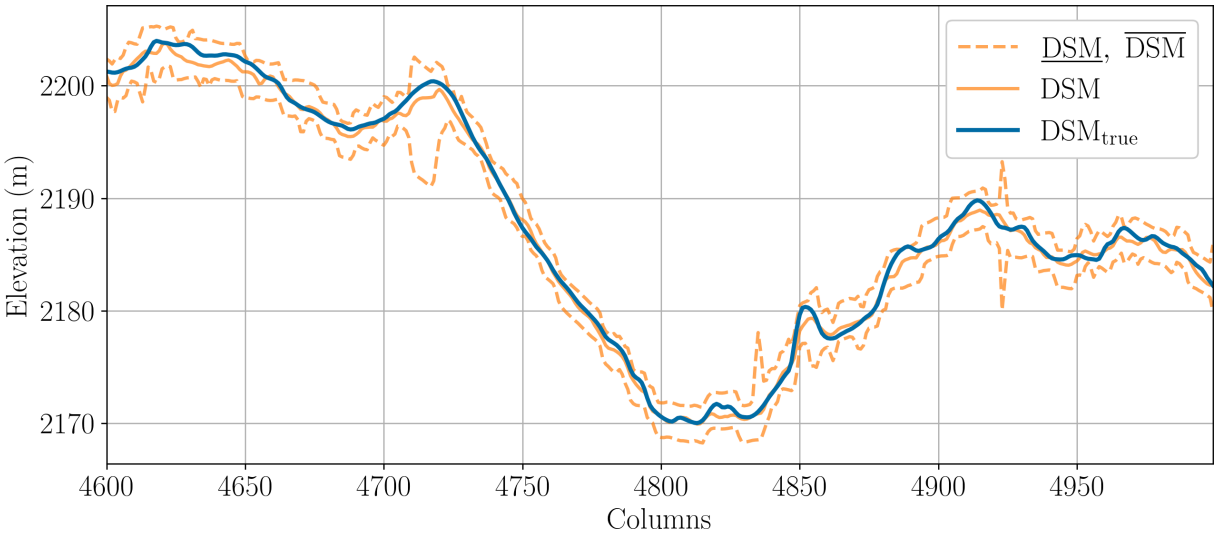
(a) Photogrammetry DSM. Orange square is detailed in (a)



(b) Photogrammetry DSM zoom. Orange line is detailed in (d)

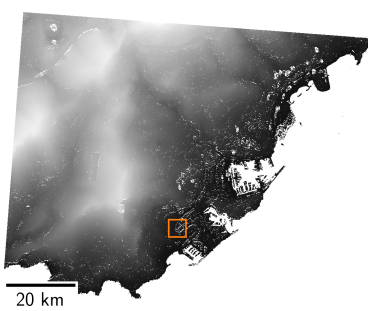


(c) DSM with wrong intervals in red.

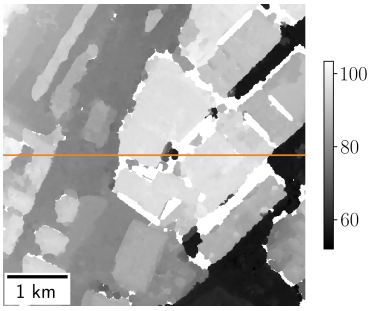


(d) DSM, ground truth and elevation intervals along the orange line of Figure 6.25(b)

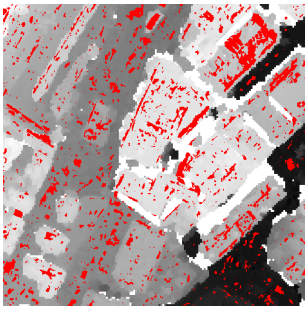
Figure 6.25: Detailed DSM over the mountainous region near Grenoble.



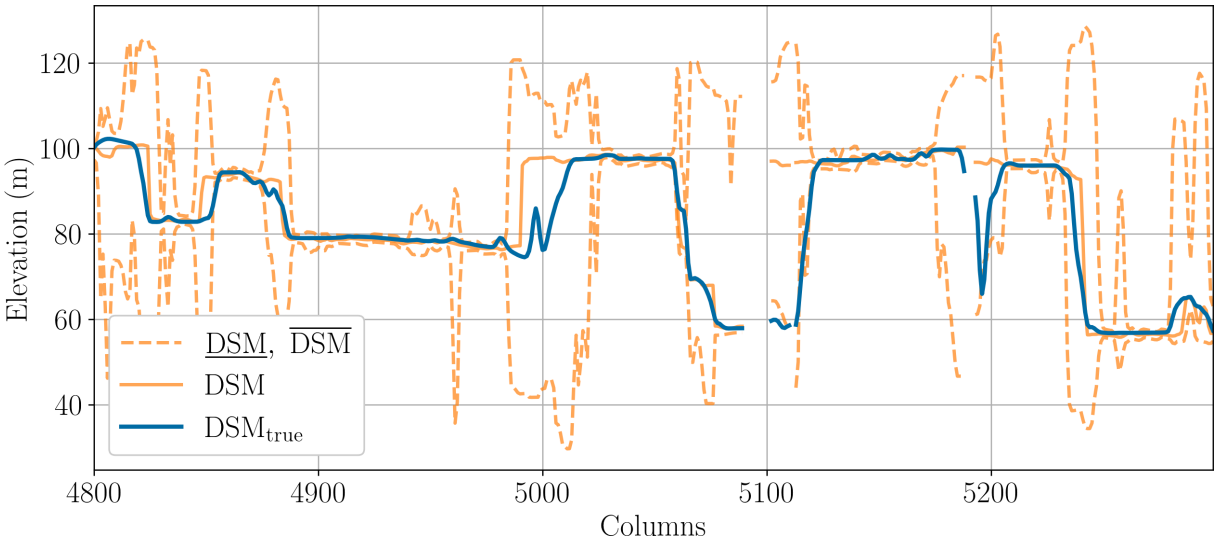
(a) Photogrammetry DSM. Orange square is detailed in (a)



(b) Photogrammetry DSM zoom. Orange line is detailed in (d)

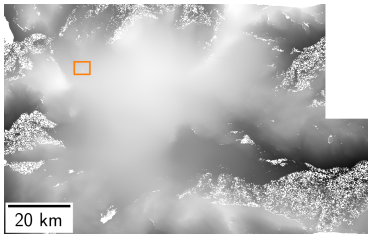


(c) DSM with wrong intervals in red.

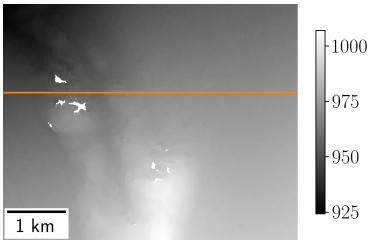


(d) DSM, ground truth and elevation intervals along the orange line of (b)

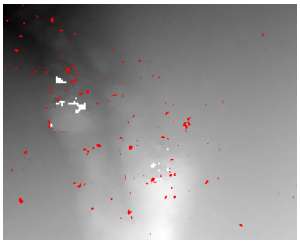
Figure 6.26: Detailed DSM over the city of Monaco.



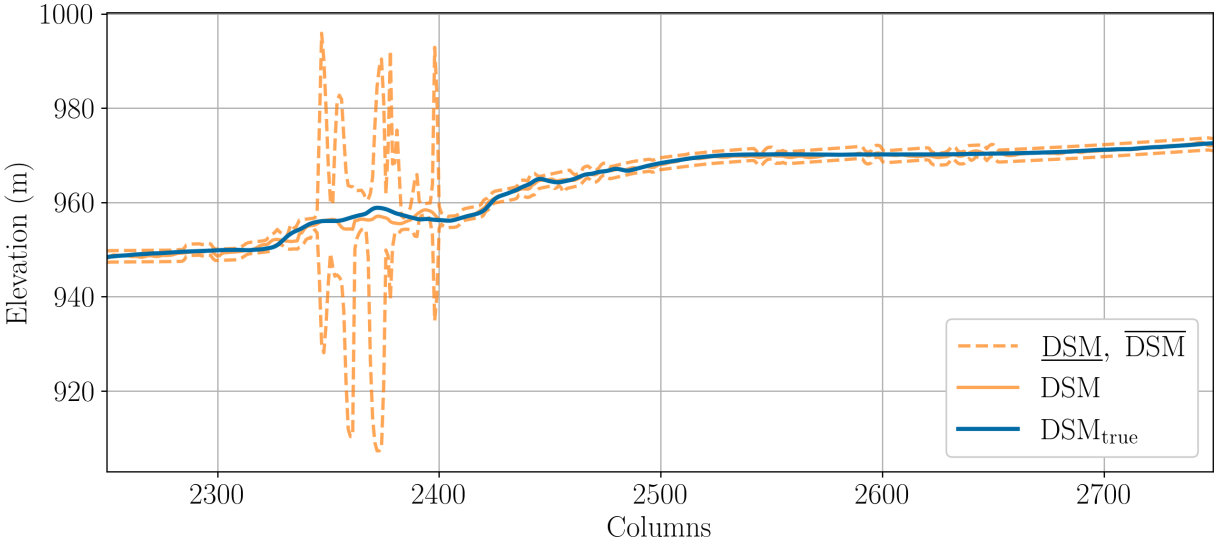
(a) Photogrammetry DSM. Orange square is detailed in (b)



(b) Photogrammetry DSM zoom. Orange line is detailed in (d)



(c) DSM with wrong intervals in red.



(d) DSM, ground truth and elevation intervals along the orange line of (b)

Figure 6.27: Detailed DSM over the Langfjordjøkelen glacier.

intervals present a great accuracy, between 98.1% and 99.7%. Scenes that do not verify the 90% objective are the urban stereo pairs over Bordeaux, Montpellier, and Paris. In general, it seems that urban DSMs have a lower accuracy than rural ones, which was to be expected as there are more steep variations of elevation in cities due to the presence of buildings. The Montpellier DSM and Bordeaux DSM are close to the 90% objective, but intervals over Paris present an elevation accuracy of only 84.6% . We will see in Section 6.7 that some other sources of errors can explain why $Z_{acc} < 90\%$ on those scenes, such as vibration of the satellite during images acquisition or ground truth non-synchronicity issues. Provided that those errors are not present, we can claim that elevation intervals are accurate enough. They correctly estimate the error committed during the dense matching step, and the propagation of disparity intervals to elevation intervals is properly carried out.

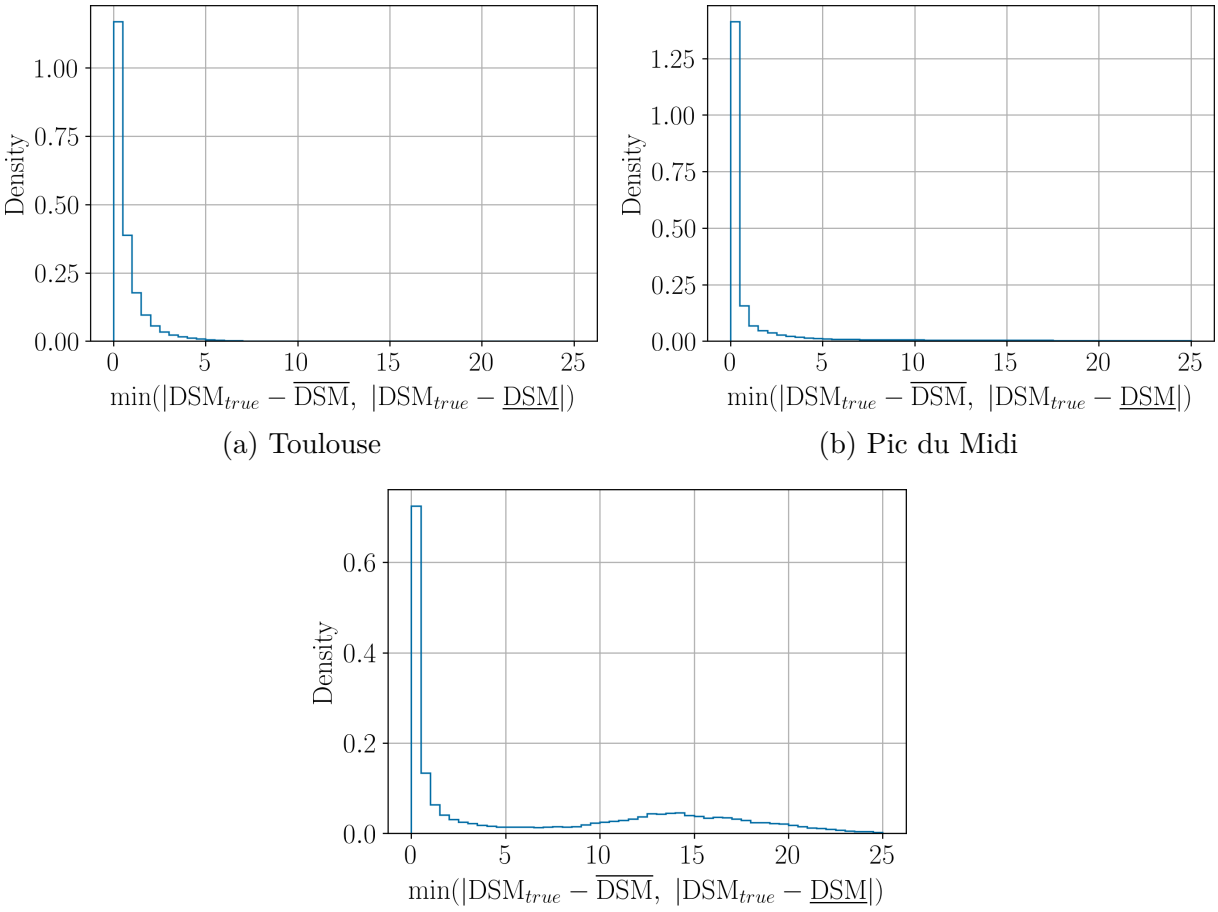
6.4.2 Residual Elevation Error

The residual elevation error Z_ε, defined in Equation (6.8), estimates the gap between intervals and the ground truth, when intervals do not contain the ground truth. We want it to be as small as possible. Z_ε is less than, or almost equal to, half a pixel for the majority of scenes, which is also around 0.5% of the disparity range. This is really low, especially when compared to disparity intervals of Table 5.1 where ε was typically around 3% of the disparity range. Because Z_ε is the median of error, it means that half of wrong intervals are less than Z_ε pixels away from one of the bounds of the elevation intervals. Consequently, extending intervals by Z_ε would divide by two the number of wrong intervals. For instance, on the Monaco scene, $Z_\varepsilon = 0.57$pix, $r_{alt} = 0.99$ m/pix and $Z_{acc} = 90.3\%$. Defining the extended intervals as:

$$[\underline{\mathrm{DSM}} - Z_\varepsilon \cdot r_{alt}, \overline{\mathrm{DSM}} + Z_\varepsilon \cdot r_{alt}] \approx [\underline{\mathrm{DSM}} - 0.56, \overline{\mathrm{DSM}} + 0.56]$$

As the accuracy is 90.3%, it means that 9.7% of pixels are not accurate. Using $[\underline{\mathrm{DSM}} - 0.56, \overline{\mathrm{DSM}} + 0.56]$ would lead to a new accuracy of $90.3 + \dfrac{9.7}{2} = 95.15\%$. For every scene, validating the 90% accuracy, Z_ε provides an easy extension for obtaining intervals with 95% accuracy.

Figure 6.28 displays histograms of the distribution of errors, from which Z_ε is the median. For both urban and rural scenes, errors are mostly of a few pixels. The shape of distributions on other scenes are similar to the Toulouse or Pic du Midi scenes. There is one Z_ε value that stands out, for the Hellmem scene, where it equals 2.76 pixels. Figure 6.28(c) displays the distribution of errors on the Hellmem scene, where the majority of the distribution is still located on the first peak of the distribution. There is however the presence of a tail on the distributions, that was not present on other scenes. We have



(a) Toulouse (b) Pic du Midi

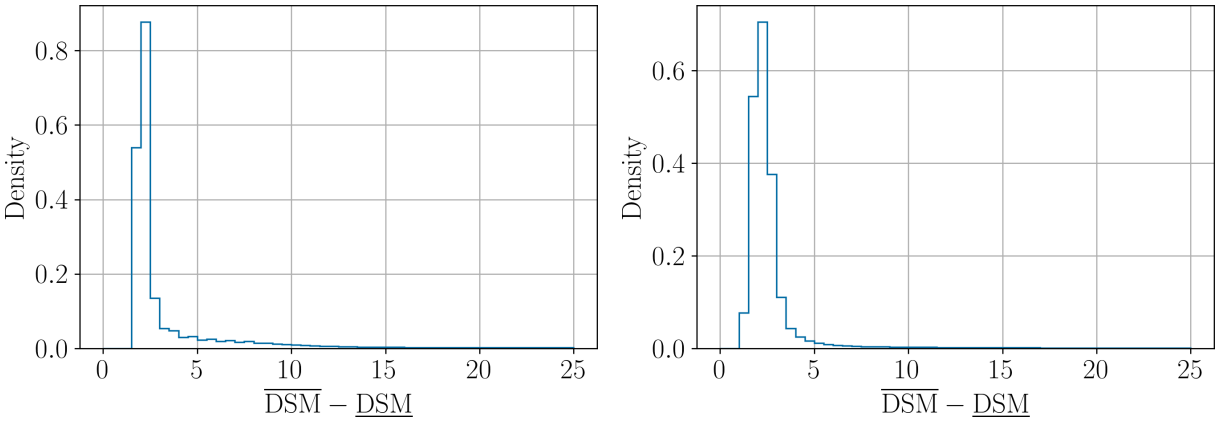
(c) Hellmem

Figure 6.28: Histograms of the residual error over the Toulouse, Pic du Midi and Hellmem scenes. Z_ε is defined as the median of those distributions.

to keep in mind that the accuracy for this scene is 98.8%, so that Z_ε is only computed on 1.2% of valid intervals. $Z_\varepsilon = 2.76$ is therefore not necessarily a concerning observation, as it concerns very few pixels. It is more surprising that it equals 0.12 for the Graasubreen DSM which already has the best accuracy (99.7%) out of all scenes. This translates the great accuracy performance of the elevation intervals estimation on this area.

6.4.3 Relative Elevation Size

The relative elevation size Z_{size}, defined in Equation (6.9), estimates the size of intervals, both correct and incorrect. Z_{size} is usually around 2 pixels wide. The Bordeaux, Grenoble and Monaco datasets are the only exceptions, where Z_{size} is closer to 4 pixels in size. It does not seem to be correlated neither to the accuracy, nor to the altimetric ratio or acquisition angles of the satellites. In any case, 4 pixels is not ideal, but it is not excessively large either, especially as the considered disparity range is $[-50, 50]$ in pixels. We can deduce from those values of Z_{size} that the predicted DSM is close to the true elevation.



(a) Toulouse (b) Pic Du Midi

Figure 6.29: Histograms of the relative size over the Toulouse and Pic du Midi scenes. Z_{size} is defined as the median of those distributions.

Figure 6.29 presents the distributions of relative sizes over the Toulouse and Hellmem scenes. It provides a more complete overview of the distribution of sizes than simply the indicator Z_{size} which is defined as the median of those distributions. The distributions on other scenes have a similar shape.

6.4.4 Altimetric Ratio and Invalid Pixels

Looking at Table 6.3, it seems that high values of the altimetric ratio r_{alt} are correlated to "low" values of accuracy. This correlation does not imply a causal relationship. Indeed, scenes with low accuracy are urban scenes, on which the 3D reconstruction is harder due to the strong variations of elevation. In order to limit the number of occlusions in urban scenes, we often reduce the convergence angle of satellites. A low convergence angle results in a high altimetric ratio, as seen in Figure 6.2. This explains why we observe a high altimetric ratio for scenes with low accuracy.

The proportion of invalid pixels that are not considered in the metrics can sometimes be quite high, as it is the case for Monaco (41%) or Hellmem (34.7%). This is due to the presence of water in large part of the image (Figure 6.9(a)) or simply the absence of data from the provided ground truth (Figure 6.7(b))

6.5 Comparison with "Naive" Intervals

The fact that most intervals have a relative size Z_{size} of around 2 pixels may raise the following question: would "naive" intervals be accurate ? We define naive intervals $[\underline{\text{DSM}}_{naive}, \overline{\text{DSM}}_{naive}]$, as intervals centered on the predicted DSM with a size of 2 times

the altimetric ratio r_{alt}:

$$[\underline{\mathrm{DSM}}_{naive},\ \overline{\mathrm{DSM}}_{naive}] = [\mathrm{DSM} - r_{alt}, \mathrm{DSM} + r_{alt}] \tag{6.10}$$

Table 6.4 allows comparing the accuracy of the naive intervals with that of our interval method. The table also contains the mean and median errors of the DSM defined as:

$$\varepsilon_{\text{mean}} = \text{mean}\,|\mathrm{DSM} - \mathrm{DSM}_{true}| \qquad \varepsilon_{\text{median}} = \text{median}\,|\mathrm{DSM} - \mathrm{DSM}_{true}| \tag{6.11}$$

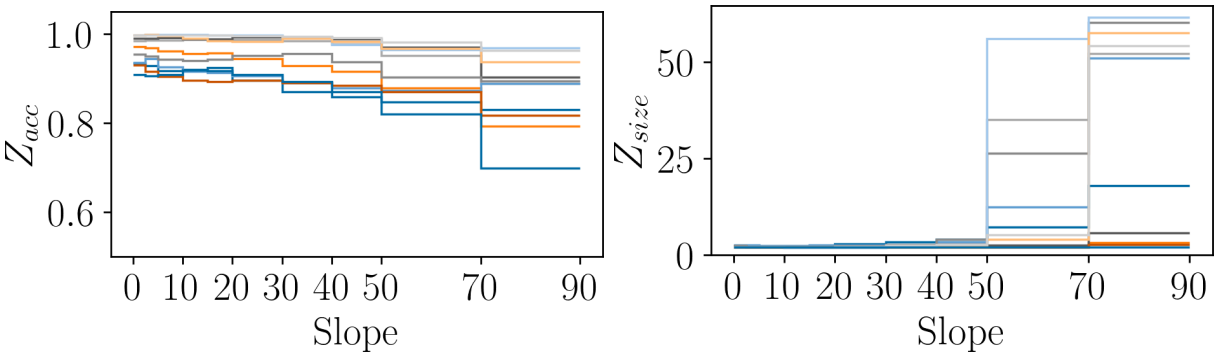
Scene	Z_{acc} (pix)	"Naive" Z_{acc} (pix)	$\varepsilon_{\text{median}}$ (m)	$\varepsilon_{\text{mean}}$ (m)
Bordeaux	89.3%	62.9%	0.86	1.93
Graasubreen	99.7%	98.8%	0.19	0.28
Hellmem	98.8%	96.4%	0.22	0.75
Grenoble	93.1%	65.7%	0.61	2.08
Langfjordjøkelen	99.4%	88.1%	0.26	1.31
Monaco	90.3%	61.2%	0.69	1.75
Montpellier	89.1%	79.4%	1.81	2.52
Paris	84.6%	82.4%	2.35	3.59
Peyto	98.9%	95.6%	0.32	0.56
Pic du midi	98.1%	86.1%	0.31	1.08
Toulouse	92.0%	83.2%	0.92	1.79

Table 6.4: Comparison of "naive" intervals accuracy with our method for elevation intervals. The median and mean error of the predicted DSM are also indicated for reference.

Table 6.4 indicates that all "naive" intervals have a lower accuracy than intervals defined using our method. The difference is sometimes quite substantial, as for the Bordeaux, Grenoble, or Monaco scenes where the accuracy drops from around 30% between the two methods. It also corresponds to scenes with a relative elevation size Z_{size} which is higher than on the others scenes. This confirms that our method correctly adapts the size of elevation intervals to each scene individually. It also proves that our method can provide good accuracy performance even when the predicted DSM is far from the ground truth (indicated by high values of $\varepsilon_{\text{median}}$ and $\varepsilon_{\text{mean}}$).

6.6 Influence of Slope on the Metrics

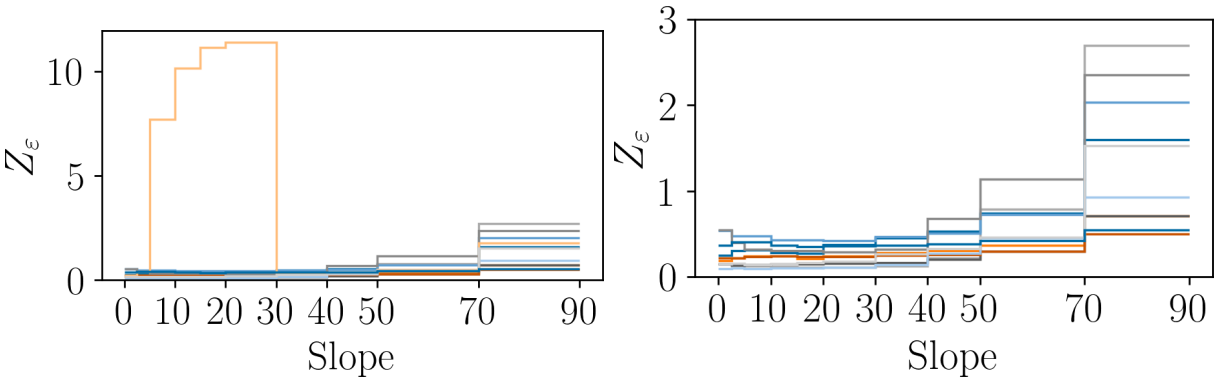
Using Equation (6.6), it is possible to compute the slope of the scene from the ground truth DSM. It would be interesting to take a deeper look at the behavior of elevation metrics depending on the slope. To do so, we compute the slope sl for each pixel and divided the ranges of slopes in different sections. Those sections are delimited by the



(a) Accuracy Z_{acc}

(b) Relative size Z_{size}

Figure 6.30: Accuracy Z_{acc} and relative size Z_{size} values depending on the slope (in degree). Each curve corresponds to a different scene.



(a) Residual error Z_ε

(b) Residual error Z_ε without Hellmem yellow curve

Figure 6.31: Residual error Z_ε values depending on the slope (in degree). Each curve corresponds to a different scene. In (b), we removed the Hellmem yellow curve from (a) for more visibility.

following values, from [Hugonnet et al., 2022]:

$$[0°,\ 2.5°,\ 5°,\ 10°,\ 15°,\ 20°,\ 30°,\ 40°,\ 50°,\ 70°,\ 90°]$$

We then evaluated the metrics on each slope range separately. Results are presented in Figures 6.30(b) and 6.31. Figures 6.31(a) and 6.31(b) are the same, except we removed one scene, Hellmem, in the right figure. As mentioned earlier, this scene has very few pixels in error, and the median is thus not a robust indicator on this scene. Those figures illustrate the fact that elevation intervals have better accuracy and smaller size on flat slopes than on steep slopes.

Regarding the accuracy of intervals, it tends to drop with steeper slopes. This can be explained due to the fact that the dense matching algorithm has better accuracy performances on smooth surfaces, provided that they are not textureless. This is in part due to the SGM regularization which penalizes changes in disparity, and therefore in

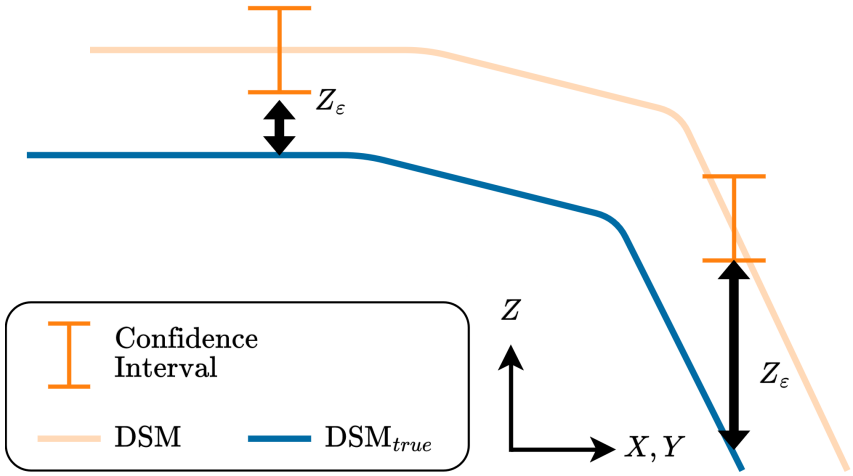


Figure 6.32: Influence of slope and planimetric error on the accuracy of intervals. The planimetric error increases the residual error on steeper slope

elevation.

The size of intervals is greater on steep slopes. This is due to the fact that we purposefully extended the disparity intervals near disparity fluctuations, which correspond to steeper slopes. Because disparity changes indicate a variation in elevation, elevation intervals obtained from disparity intervals are consequently larger on steeper slopes.

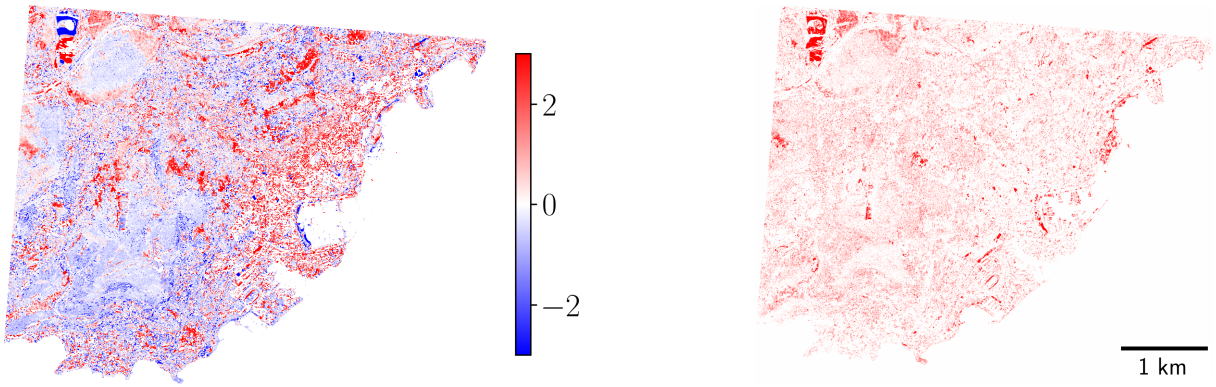
Finally, the residual error also increases with steeper slope when a planimetric error is present between the ground truth and the predicted DSM. This can also be understood using Figure 6.32, as a steeper slope on the right interval leads to a greater residual error.

The next section will investigate sources of errors independent of our method, that can affect the evaluation of metrics.

6.7 Other Sources of Error

We saw in Section 6.4 that elevation confidence intervals present high accuracy and low relative size. There were however three scenes that did not meet the accuracy objective of 90%: Bordeaux, Montpellier and Paris. We will see in this section that there are different factors which can explain why the accuracy objective was not verified on those scenes. Those reasons are:

- Asynchronicity of the ground truth and satellite acquisitions
- Rasterization of LiDAR data over vegetation
- Vibrations of the satellite during the images acquisition



(a) Error DSM − DSM_{true} (b) Position of wrong intervals, in red.

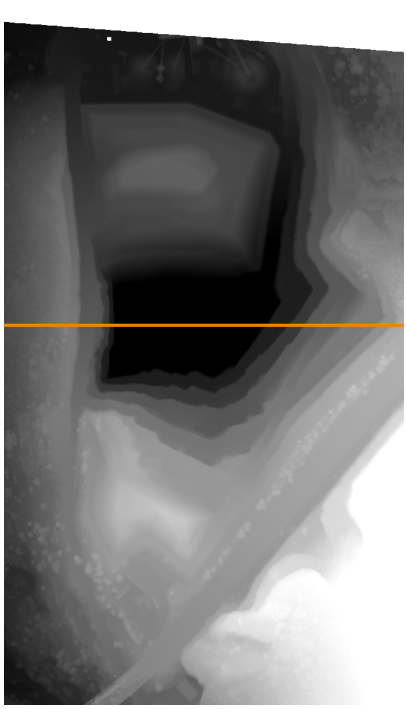
Figure 6.33: Errors and position of wrong intervals on the Monaco scene

6.7.1 Asynchronicity of Sources

The dates of acquisition of LiDAR data and Pléiades images were presented in Table 6.1. In some scenes, large periods of time separate the LiDAR data from the satellite image, partly because we also tried to minimize seasonality changes. Specifically, the time between acquisitions for Bordeaux, Grenoble, Monaco, and Montpellier ranges between 5 and 13 months. This can lead to major changes of elevation in those scenes. A remarkable example of this can be found in the Monaco scene. If we take a look at the errors on this scene in Figure 6.33, we can see that there is a strong concentration of errors in the top left corner of the image.

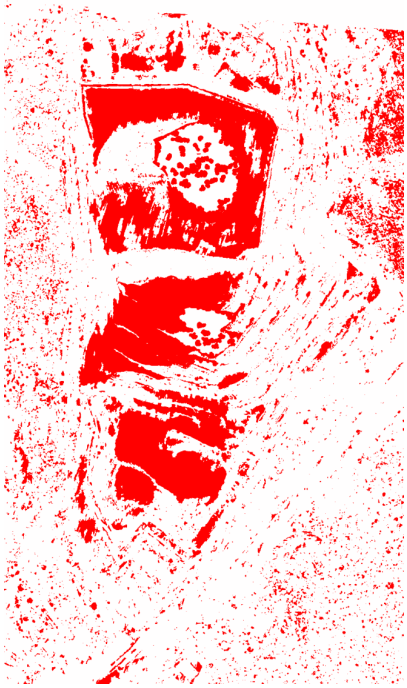
Figure 6.34 is a zoom on this area. By looking at the differences between the ground truth DSM in Figure 6.34(a) and the RGB image in Figure 6.34(b), we can see that there are some differences between the ground truth DSM and the satellite image. Most notably, the bottom left of the quarry levels are more constricted on the ground truth than on the RGB image. We can confidently state that the quarry was excavated between the images' acquisition and the LiDAR acquisition that occurred a year later. This explains why so many intervals are wrong in the area, in comparison with the rest of the scene. This can also be observed on a cross-section of the DSMs presented in Figure 6.34(d). Those intervals are thus false negatives, and lower the global accuracy of the scene, as they approximately represent around 1.5% of the valid data on this scene.

This type of false negative can be found on a smaller scale in most urban scenes due to asynchronicity between satellite and LiDAR acquisitions. We did not manually detect and correct every building that was destroyed or built in-between acquisitions, as it was too time-consuming. The accuracy results could therefore be further improved in urban scenes. It is possible, although very unlikely, that asynchronicity causes false positives during the accuracy computation.

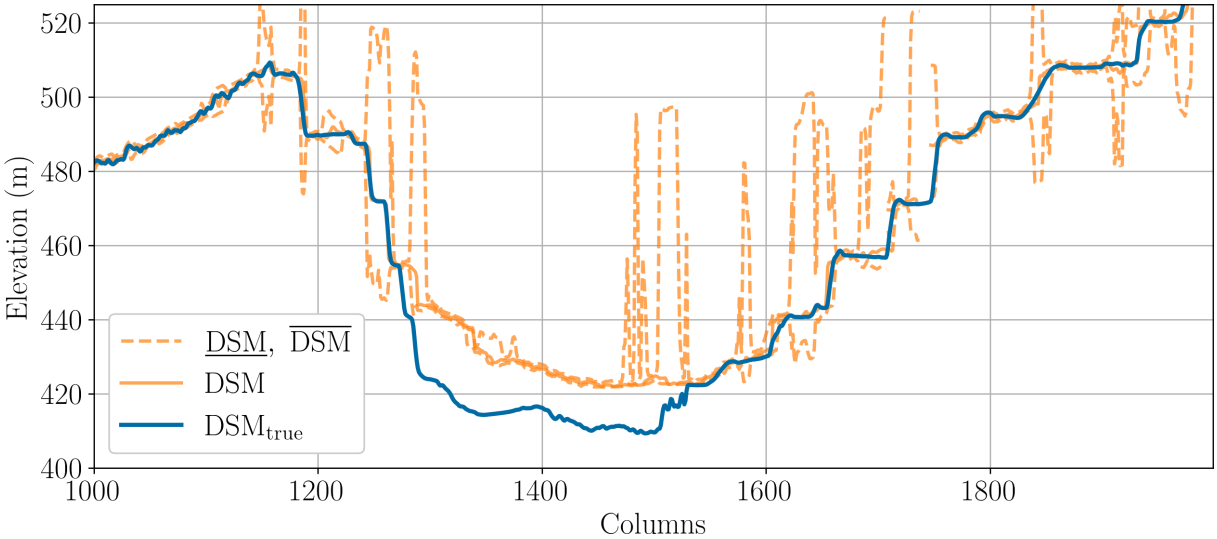


(a) Ground truth DSM. Orange line is detailed in (d)

(b) RGB image



(c) Position of wrong intervals, in red.



(d) DSM, ground truth and elevation intervals along the orange line of (a).

Figure 6.34: Detail over the La Turbie quarry, in the top left corner of the Monaco scene.

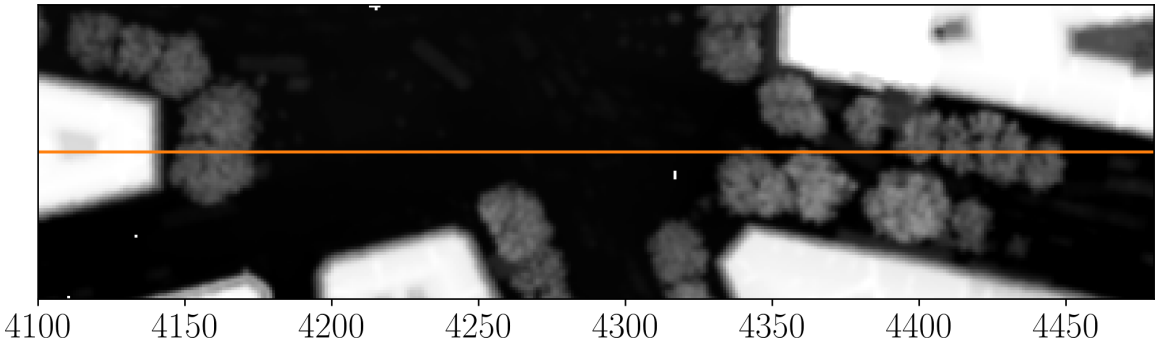
6.7.2 LiDAR Data Over Vegetation

Intervals computed over the Paris scene have an accuracy of 84.6%, which is the lowest of all scenes. When trying to understand why intervals did not seem to perform correctly, we noticed that many pixels in error were representing vegetation. This can be observed in Figure 6.35, where many intervals are wrong near trees. By looking at the intervals in Figures 6.35 and 6.36, we can see that the photogrammetry DSM over-estimates the height of the canopy: columns 4150 to 4170 and 4345 to 4450 of the Paris scene in Figure 6.35(d), and columns 3400 to 3800 of the Bordeaux scene in Figure 6.36(d). Other buildings, or the ground, do not present such error. One probable hypothesis is that because the LiDAR HD over Paris was acquired in early March, and the satellite images in the last day of May, tree foliage had major differences. Firstly, trees seem bigger on the RGB image than on the LiDAR DSM, which is probably due to the increase of foliage in-between acquisitions. Also, the LiDAR probably acquired points both on the top of trees and on the ground, through the sparse foliage. When rasterizing the LiDAR point cloud using a Gaussian mean, we averaged both points on top of the trees and on the ground, resulting in a mix between the two elevations. Due to its resolution, the photogrammetry DSM does not create multiple points at both the top and the base of the same tree. It therefore only predicts the elevation of the tree canopy, resulting in the observed error when comparing with the LiDAR DSM. We tried to mask the vegetation pixels, using a mask computed from the Normalized Difference Vegetation Index [Gao, 1996], but the quality of the masks were not sufficiently accurate, and tend to mask many pixels that were not vegetation. We therefore decided not to use them. Also, this effect was particularly present on the Paris scene, which is probably because the LiDAR was acquired in early March, while the LiDAR on other scenes was acquired from June to September, when the vegetation is denser. However, we will see in the next section that there is a greater source of error occurring on that scene, which can also be the source of errors leading to a low accuracy.

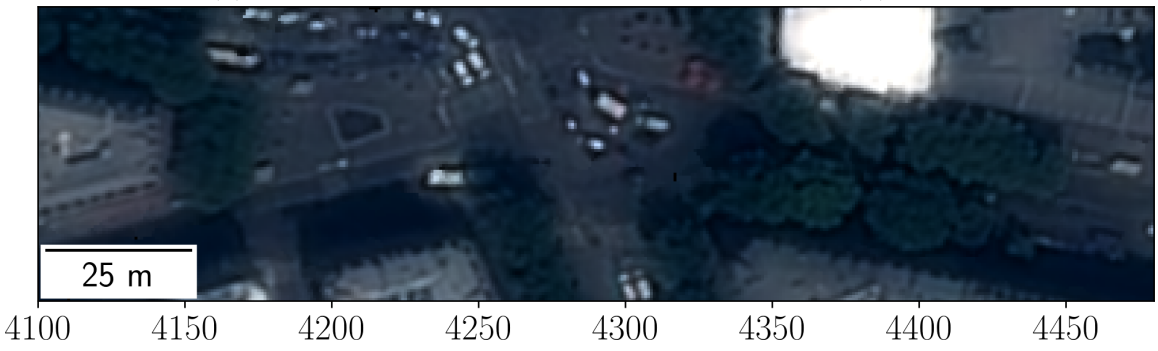
6.7.3 Vibration of the Satellite

The previous sources of errors concerned the quality of the ground truth DSM. However, there is one source of errors that we did not take into consideration in our methods: the satellite vibration during acquisition. During the acquisition of images, push-broom captors may experience vibrations which are not modeled in the RPC model. The vibrations occur on the pitch angle, whose axis is perpendicular to the direction of flight of the satellite, as depicted in Figure 6.37. This translates into low frequency along-track biases on the DSM, sometimes called undulations [Hugonnet et al., 2022]. This can be detected by looking at the difference between the photogrammetry and the ground truth DSMs:

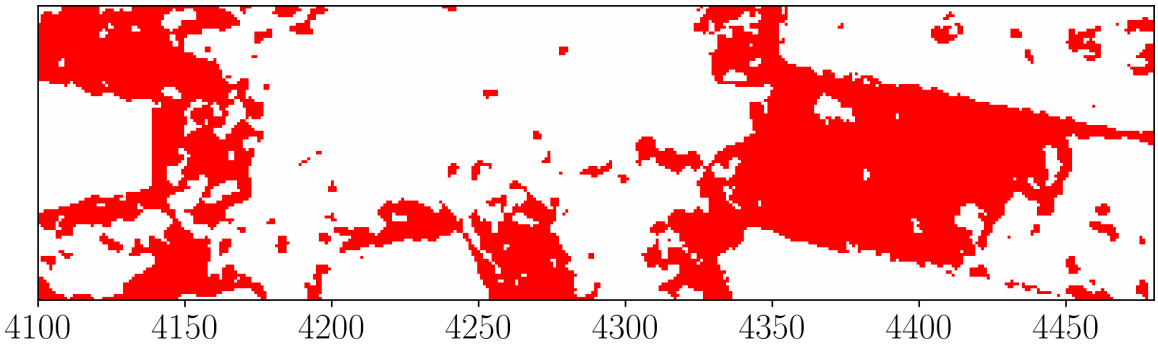
$$\mathrm{DSM}(row,\ col) - \mathrm{DSM}_{true}(row,\ col) \tag{6.12}$$



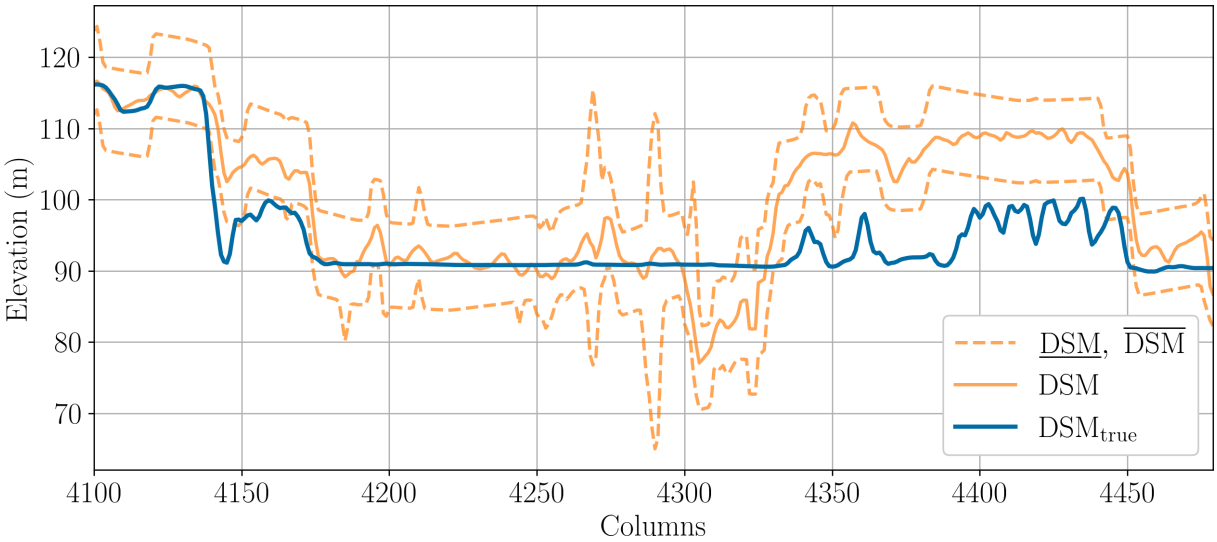
(a) Ground truth DSM. Orange line is detailed in (d)



(b) RGB image

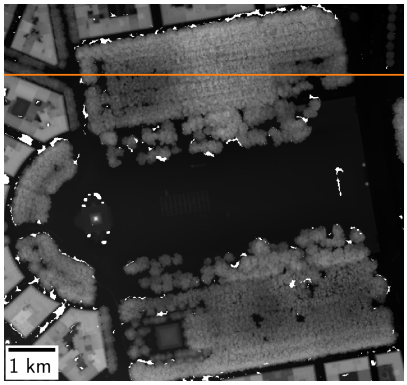


(c) Position of wrong intervals, in red.



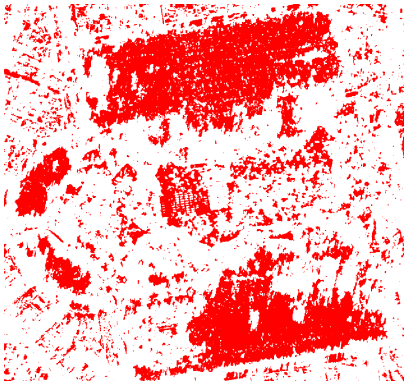
(d) DSM, ground truth and elevation intervals along the orange line of (a)

Figure 6.35: Zoom near Saint-Augustin, in the Paris scene

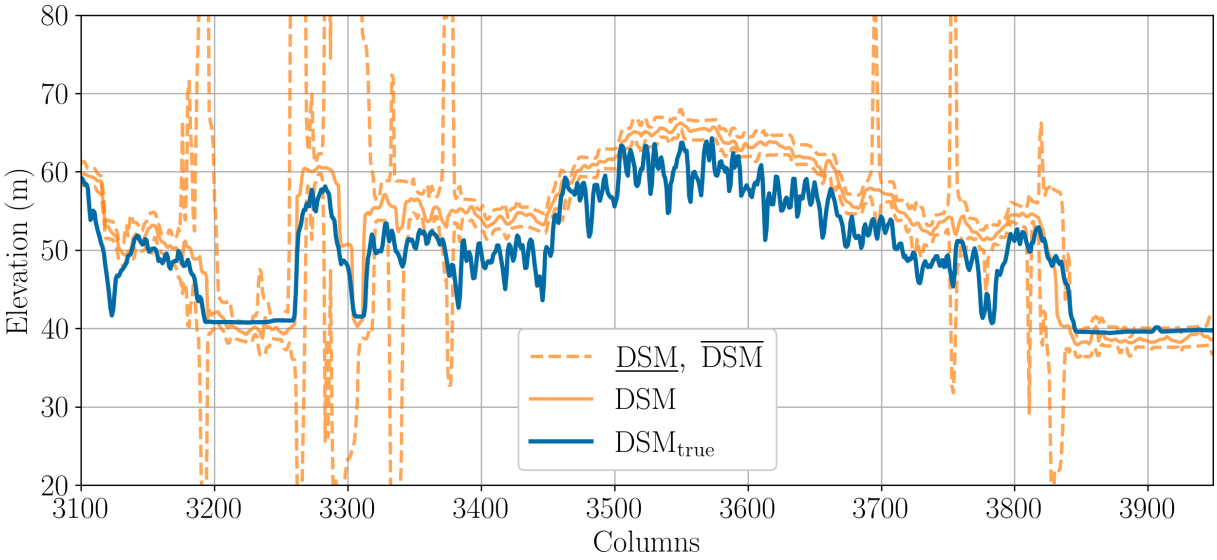


(a) Ground truth DSM. Orange line is detailed in (d)

(b) RGB image



(c) Position of wrong intervals, in red.



(d) DSM, ground truth and elevation intervals along the orange line of (a).

Figure 6.36: Detail over the Quinconces square, in the Monaco scene.

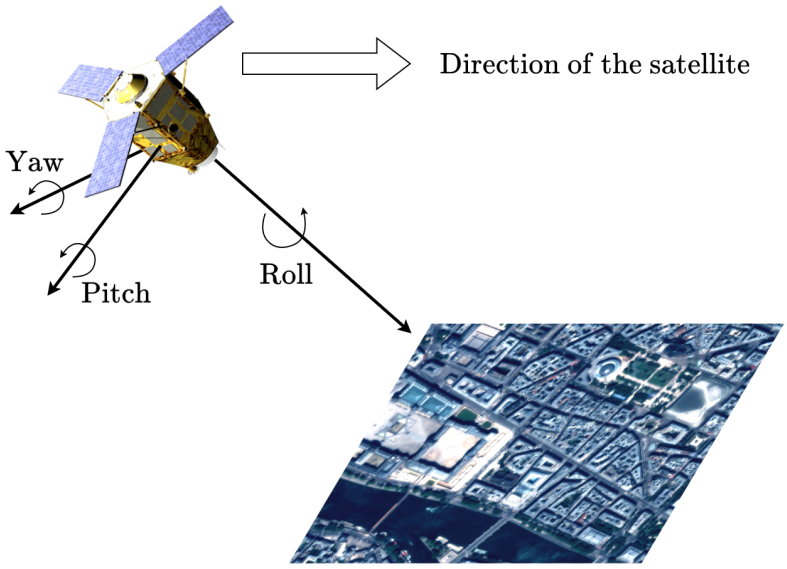
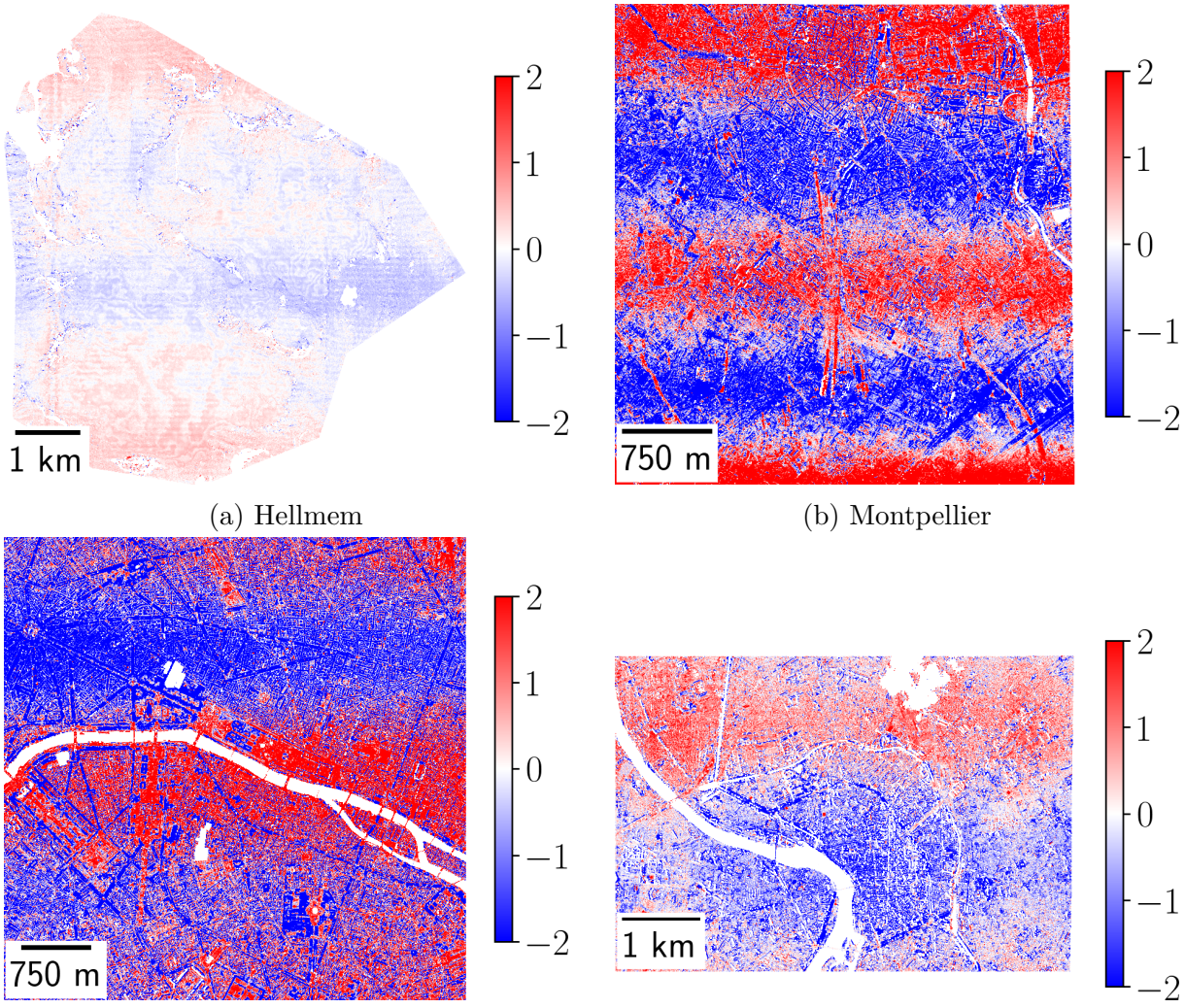


Figure 6.37: Angles of a satellite. Vibrations of the satellite occur on the pitch axis.

Without biases, this difference (or residual error), should not be correlated with the position *row*, *col* in the image. Figure 6.38 presents all scenes for which we detected a bias. The bias is small on Figure 6.38(a) but is hard to miss on Figures 6.38(b) to 6.38(d). Figure 6.39 indicates that those biases can reach 2.5m of amplitude.

When computing elevation confidence intervals, we did not consider potential errors of this magnitude for the RPC models. Our method principally aimed to model and propagate the errors stemming from the dense matching step, which we considered to be the part of the pipeline where the largest errors could occur. Biases observed in Figure 6.38 indicate that only considering the dense matching errors is not sufficient when producing DSMs with Pléiades images. The CO3D satellites will not use a push-broom sensor, but rather a CCD Bayer matrix, which can potentially reduce vibration issues. However, the satellite will still need to stabilize itself during acquisition, therefore some vibrations can still be expected.

In the scope of this thesis, we did not try to model the uncertainty associated with potential vibrations of the satellite. It would however be interesting to determine if Montpellier and Paris scenes could validate the 90% accuracy objective without the effect of vibrations. We therefore propose a simple correction of the vibration effect, which is not meant to perfectly solve the issue, but rather to sufficiently reduce it for a better evaluation of the elevation interval metrics. In order to do so, we computed the residual error ($\mathrm{DSM} - \mathrm{DSM}_{true}$), and attempted to model the observed bias using a least square approach. We assume that the bias depends only on the row of the image and propose to



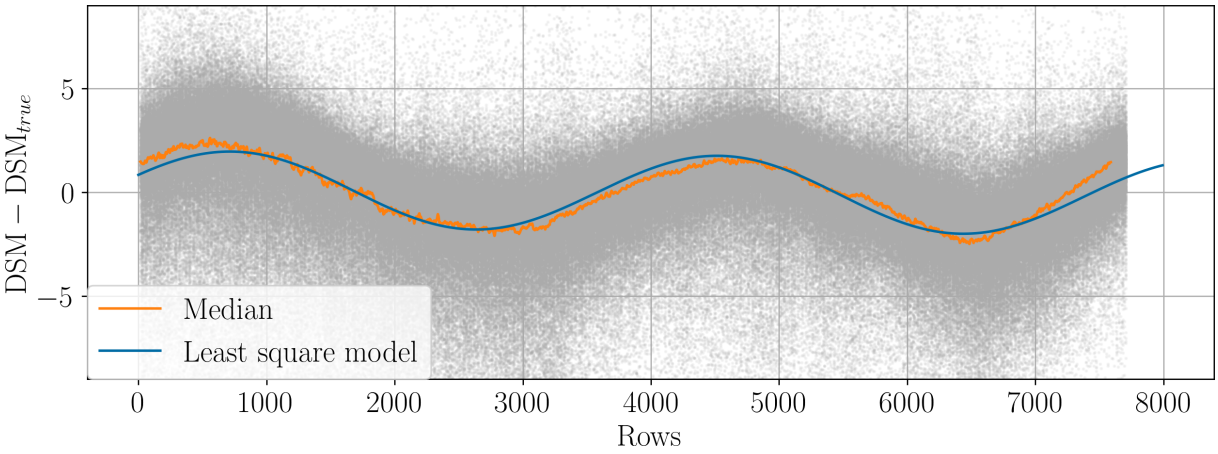
(a) Hellmem

(b) Montpellier

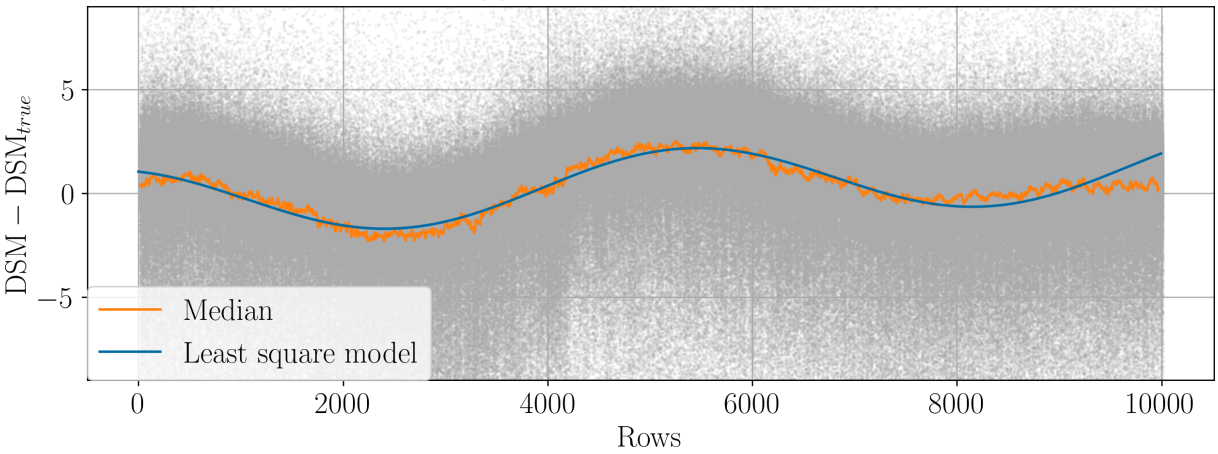
(c) Paris

(d) Toulouse

Figure 6.38: Difference (DSM − DSM_{true}) for different scenes. A bias along rows (along-track) appears, suggesting vibrations during the image acquisition. Biases on the Hellmem DSM are smaller that for the other three scenes.



(a) Montpellier scene



(b) Paris scene

Figure 6.39: Scatter plot where gray points are the residual error (DSM − DSM$_{true}$) of a pixel. The x-axis represents the row of each pixel. As an indicator, we also computed the median residual error for each row, appearing in orange. The blue line is the optimized model from Equation (6.13).

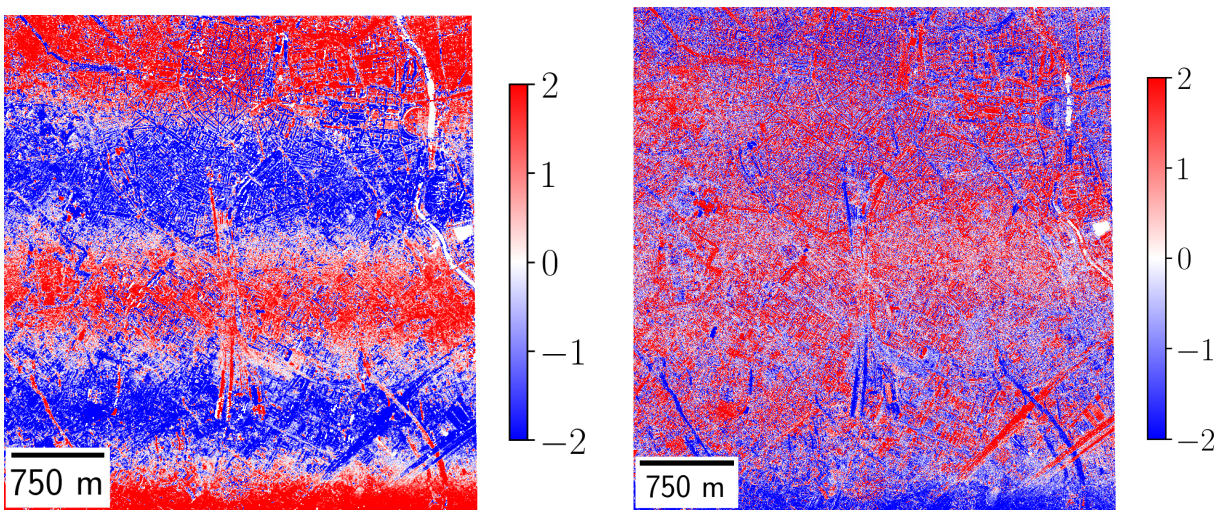
simply model it by a cosine function carried by a linear function of the row:

$$bias(row) = A_1 \cos(A_2 \cdot row + A_3) + A_4 \cdot row + A_5 \tag{6.13}$$

where $(A_1, \ldots, A_5)$ are the parameters of our model. Figure 6.39 shows scatter plots of the residual error (DSM − DSM$_{true}$) where the x-axis indicates image rows. We also computed the median of the residual error for each row as an indicator of the distribution of errors, and the optimized model from Equation (6.13).

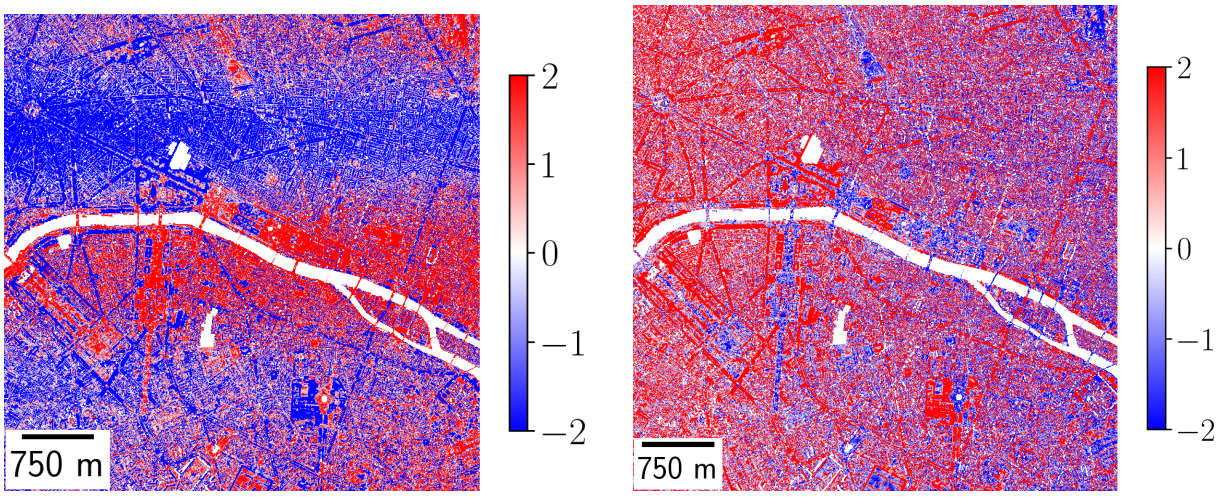
Having estimated the bias, we can subtract it from the DSM and its confidence intervals. This will not change the relative size of the intervals, as we apply the same elevation shift to both bounds. The accuracy and residual error are however impacted by this bias rectification. Here are the improvements:

- For the Montpellier scene, the accuracy Z_{acc} increases 89.1% to 92.6%. The residual



(a) Residual error with bias (b) Residual error without bias

Figure 6.40: Residual error (DSM − DSM_{true}) with and without the bias estimated in Figure 6.39 for the Montpellier scene.



(a) Residual error with bias (b) Residual error without bias

Figure 6.41: Residual error (DSM − DSM_{true}) with and without the bias estimated in Figure 6.39, for the Paris scene.

error Z_ε also increases from 0.28 pixels to 0.32 pixels.

- For the Paris scene, the accuracy Z_{acc} increases from 84.6% to 88.1%. The residual error Z_ε changes from 0.46 pixels to 0.43 pixels.

The fact that the residual error slightly varies is due to the fact that as the accuracy increases, the set on which Z_ε is computed changes, thus changing its median.

Remark: In this section, we used the ground truth to correct the bias from the vibrations of the satellite. We used the ground truth because our objective was to verify if the biases were indeed the source of the missing accuracy of elevation intervals, so that we could safely assume our method was efficient, granted that no vibrations occurred during the acquisition.

Our aim was not to propose a general method for modeling or correcting the errors due to vibrations, in which case we could not have used the ground truth, as it is not available everywhere. However, if such were our intents, using a low resolution DSM such as the Copernicus DSM at 30m resolution could suffice to detect the presence of vibrations. This DSM was already used in the CARS pipeline as our initial elevation. By plotting the differences between the low resolution DSM and the predicted photogrammetry DSM, we can indeed observe the presence of vibrations, as seen in the following figure:

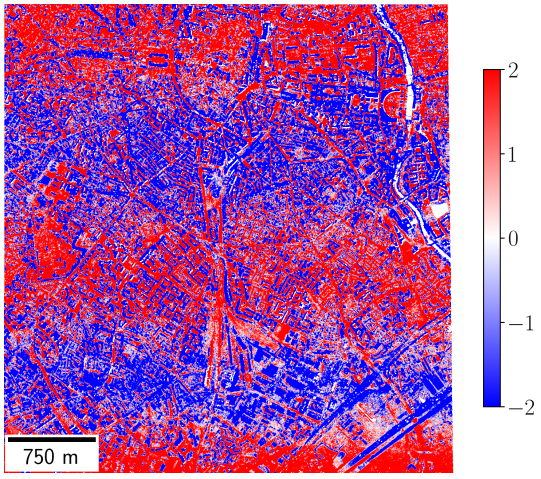


Figure 6.42: Difference DSM − DSM_{low} between the photogrammetry model DSM and the low resolution model DSM_{low} over the Montpellier scene.

The Paris scene still does not reach the 90% accuracy objective, while being very close to it. Looking at Figure 6.41(b), there still seems to be a bias, horizontally this time, as residuals are positive on the left side of the image and negative on the right side of the image. We can also distinguish that streets seem to have more positive residuals,

which is once again probably due to the vegetation issue discussed in Section 6.7.2. As the accuracy is very close to the 90% objective, and we have good reasons to think there are false negative in our test, we consider that this scene confirms the aimed accuracy performance of our method for creating elevation confidence intervals.

6.8 Unexplored Leads

In the previous sections, we evaluated the accuracy and size performance of elevation intervals. Those intervals were obtained by triangulation of disparity intervals, yielding an interval along a line of sight, which was then rasterized. As they stand, intervals are a reflection of the potential error committed during the dense matching step. During our investigations, we came up with other interesting ideas that we did not have time to get to the bottom of. In order to conclude this chapter, we will quickly go over them.

We saw in Figures 6.24(c), 6.34(d) and 6.36(d) that elevation intervals could sometimes reach very high values, like 80 m in the case of Toulouse. The presence of those intervals reflect the fact that bounds of corresponding disparity intervals have a high possibility degree, and that the difference between choosing one bound or the other is not obvious. It could be wise to filter out the 3D points leading to such large intervals. After all, a large elevation interval suggests that we are not certain about the position of the point, so removing it seems natural. Additionally, we ignored the planimetric error ΔXY during the rasterization as presented in Figure 6.3, but the hypothesis that ΔXY is small is probably not valid for large intervals. For acquisitions with incidence angles, around 11°, an elevation interval of 50m would mean that the planimetric uncertainty is around $50 \tan(11°) \approx 20$ m. For the Peyto scene, where the incidence is 21.3°, this uncertainty reaches 46 m. Not only are we unsure about the elevation of the point, but there is much uncertainty regarding which cell it falls into during the rasterization process. We can either filter those points directly after the dense matching step, based on the size of the intervals, or after they have been triangulated, based on the resulting planimetric/altimetric shift. The latter takes into account the acquisition angles of images, and is not limited to the information contained in the disparity map.

When triangulating the disparity, we consider line of sights to be 3D lines. However, they represent the position of a pixel, which is not a point but a surface (determined by the resolution of images). Lines of sights are therefore not exactly lines, but rather a cone or a cylinder. We could even go further with this model, by taking into consideration the accuracy of lines of sights into the definition of those cylinders. Intersecting "cylinders" of sights would then yield a 3D volume. We would then have to processed differently the obtained volumes during the rasterization step.

The rasterization step uses a Gaussian mean based on the distance of 3D points to the center of a cell. We could additionally use the size of intervals to adapt the weights in the rasterization steps. Doing so, we would favor points with small intervals, *i.e.* for which we are confident about their position, to improve the final DSM. If we reason with 3D volumes, we could compute a density on each volume and project those densities on the final raster.

The uncertainty information contained in the different intervals (disparity or elevation) could also be used to improve the final DSM. As it stands, the computation of intervals does not modify the final DSM.

Conclusion

In this final chapter, we propagated the uncertainty associated with the disparity from Chapter 5 into elevation intervals at the end of the photogrammetry pipeline. Elevation intervals are a result of the uncertainty due to the dense matching step, which is the most complex part of the pipeline, and therefore the most important source of uncertainty. Intervals have an accuracy of 90%, and their size is proportionate to the potential errors between the predicted DSM and the true elevation. Intervals have a better performance in terms of accuracy and size in natural landscape than in urban ones, as those scenes contain more elevation differences and steeper slopes. The methodology developed for computing elevation confidence intervals aims to satisfy requirements expressed by DSM users. They can also provide a solution for the future CO3D mission requirements, as a performance map will need to be provided alongside the DSM.

Conclusion

Context

Usage of elevation data for environmental applications, urban planning, risk assessment, *etc.* requires the massive production of Digital Surface Model (DSM)s. DSMs at a global scale can be produced using satellites orbiting the Earth, and using different techniques such as LiDAR measures, radar interferometry or optical photogrammetry. In particular, optical sensors, which have become relatively low-cost, allow producing dense DSMs with sub-meter resolution. In this context, CNES and Airbus are preparing the launch of the CO3D mission, consisting in two pairs of low-cost optical satellites dedicated to the production of high-resolution DSMs across the globe using stereophotogrammetry.

Photogrammetry is a complex task, consisting in multiple successive algorithms to process images and extract the elevation information contained through the parallax effect. It can be divided into the following main steps:

- Resampling of stereo images into a convenient geometry.
- Dense matching of pixels between images.
- Triangulation of a 3D point for each match.
- Rasterization of the point cloud on a regular grid.

Along these steps, many uncertainties arise, which can lead to errors of varying magnitude. The objective of this thesis was to quantify and propagate the uncertainty alongside a photogrammetry pipeline, in preparation of the CO3D mission. In particular, we focused on the CARS pipeline which will be used to process the CO3D data. Furthermore, one of the requirements of the CO3D mission is to produce a performance map alongside DSMs. Many DSM users also seek to know the quality of the predicted DSM, usually characterized by confidence intervals. The main contribution of this thesis is precisely the development of a method for computing confidence intervals, which can also be used as a performance map for the CO3D mission.

Contributions

In this thesis, we used other uncertainty models than well-known probability distributions. We considered *imprecise* probabilities and more specifically possibility distributions, more adapted to model epistemic uncertainty, *i.e.* arising from a lack of knowledge, by opposition to the uncertainty due to a purely random process. Those models define credal sets, *i.e.* convex sets of probability distributions.

When propagating multiple sources of uncertainty, it is required to compute a multivariate model of uncertainty accounting for the different dependencies between uncertain sources. We proposed to use dependency models known as copulas to construct multivariate credal sets. We introduced three methods for aggregating marginal credal sets into multivariate credal sets using copulas, namely $\mathfrak{M}_{robust}$, $\mathfrak{M}_{mass}$ and $\mathfrak{M}_{agg}$. Those models are not equivalent, and their computation presents varying degrees of complexity. We investigated the relations between those sets depending on the copula used to join them, as well as the specific models used to define marginal credal sets, such as the aforementioned possibility distributions.

We used the previous results concerning multivariate credal sets to propagate the uncertainty in a specific part of the photogrammetry pipeline: the dense matching step. More specifically, we consider the computation of the matching cost between every pixel of the stereo images. We showed that the models could correctly estimate the propagated uncertainty regarding the matching cost on a real pair of stereo images.

The cost volume is used to compute the disparity map, encoding the pairing of pixels to be triangulated. Computing a cost volume allowing to correctly estimate the disparity is the hardest part of the stereo pipeline. Correctly estimating the uncertainty on the predicted disparity is therefore crucial for the rest of the pipeline. However, only considering the propagated uncertainty on the cost volume, as we previously did, is not sufficient for the correct uncertainty estimation of the disparity map. We therefore proposed to use possibility distributions to model the epistemic uncertainty regarding the choice of each disparity. For each considered pixel, we used those possibility distributions to determine a disparity confidence interval. We evaluated the accuracy of the intervals using real stereo images, and observed that intervals contain the true disparity at least 90% of the time. This method for creating intervals can be applied to a wide range of stereo algorithms and is not restricted to satellite photogrammetry. To the best of our knowledge, it is also the first time such disparity confidence intervals are computed.

We then propagated those disparity confidence intervals all the way to the end of the pipeline, where they take the form of elevation confidence intervals associated with the predicted DSM. We evaluated the performance of elevation intervals on real satellite im-

ages, for which we possess a reference high quality DSM. Intervals are once again accurate 90% of the time, validating the performances of our new method. We also implemented this method for estimating disparity and elevation confidence intervals into the CARS (`https://github.com/CNES/cars`) and Pandora (`https://github.com/CNES/Pandora`) software, developed by CNES. They are already publicly available, and can be used to produce elevation confidence intervals for DSMs.

Limitations and Perspectives

We demonstrated that possibility distributions and copulas could be used to propagate uncertainty in a problem such as the evaluation of a cost volume. Implementing this propagation remains a difficult challenge. It requires using simple cost functions and other simplifying assumptions. It also requires large processing capacities to be carried out efficiently, as we joined the uncertainty of thousands of different pixels in our experiments.

We evaluated our method for producing elevation intervals using high resolution DSMs obtained from LiDAR data. However, we only had access to DSMs provided by glaciologists, or high resolution point clouds from the LiDAR HD program. This means that we only observed landscapes which either contained glaciers, or were located in France. Extending the evaluation of intervals to a broader diversity of locations would be valuable, such as deserts or American cities.

Another limitation of our method is that it does not take into consideration the potential errors occurring *before* the dense matching step. Those errors could occur when defining the epipolar geometry for instance, or on the localization model itself, for instance caused by vibration of the satellite as seen at the end of Chapter 6. Combining uncertainty models of the sensor itself or on its geolocation model could lead to a better estimation of the overall uncertainty of the DSM.

Different future perspective regarding our work can be considered. The method we developed for computing confidence intervals is carried out alongside the processing of the main DSM, without influencing the values it contains. However, we could imagine using the information contained in the confidence intervals to facilitate or improve the different disparity or elevation predictions. Here are a few interesting leads that could be explored:

- Disparity intervals could be computed a first time before any SGM regularization. Then, during the SGM regularization, disparities that are not contained inside disparity intervals are ignored, which could greatly reduce the amount of computation necessary.
- Another approach would be to use matching cost possibilities to apply a different

strategy when computing the disparity map. Currently, disparities are determined using a *winner-takes-all* strategy, meaning that for each pixel, the selected disparity is the one minimizing its cost curve. We then do the same for the cost volume of the right image, and remove matches that do not verify the cross-checking test of Equation (1.14). However, we could see the choice of a disparity map as a stable marriage problem [Irving, 1998], where possibilities are interpreted as degrees of preferences. This could improve the quality of the disparity map, as the choice of each disparity would consider more information than a single cost curve. It would also provide an alternative to the *winner takes all* strategy, which accounts for the vast majority of strategies used in dense matching.

- We currently extend intervals in low confidence areas using quantiles computed over a set of neighboring intervals. We could consider using instead other cost functions which present better performances near depth discontinuities, or using values of the cost volume before SGM regularization to better process intervals in those low confidence areas.

- When triangulating disparity intervals, we could take into consideration the limited resolution of line of sights. We could for instance reason with uncertain 3D volumes when computing their intersection, which would be a more realistic model than the precise lines of sight currently considered.

- Before the rasterization step, we could discard points for which the interval size is too large, as the potential error committed by considering this point is too high.

- During rasterization, we could use information from intervals to modify the weights of each point in the final value of the DSM. Points with small elevation intervals would be granted more importance in the final product than those with large intervals.

Ideas developed in this thesis for 1D matching could be extended to 2D matching, for instance in the Pandora2D tool (`https://github.com/CNES/Pandora2D`). Apart from photogrammetry, ideas developed in this thesis could be used to improve confidence criteria of the alignment of image bands for multi-spectral images, for instance in the TRISHNA mission [Lagouarde et al., 2019], or alignment of multi-temporal images, such as Sentinel-2 [Yan et al., 2018]. Taking a step back from imagery, we showed in this thesis that using other models of uncertainty than well-known probabilities can lead to new methods for estimating and characterizing potential errors. Many methods using incomplete or imperfect data for diverse applications, such as clustering, classification or decision-making, can also benefit from using imprecise probabilities.

7. Annex

This annex contains results and experiments that we consider secondary in relation to the main content. Sections 7.1 and 7.2 presents additional results concerning copulas presented in Section 2.4 and used in Section 3.2.3. Section 7.3 presents results on the median filter used in Section 5.1.3. Section 7.4 contains ablation studies of the method presented in Chapter 5.

7.1 Directional Convexity/Concavity for Copulas

This section will investigate a property shared by some copulas called directional convexity/concavity. This is a theoretical contribution, as we do not exploit them in the applications to stereophotogrammetry in Chapters 4 and 5. However, we will see that those properties are shared by many common families of copulas. The main result of this section is proposition 12, which was used to prove a specific relationship between multivariate uncertainty models in Sections 3.2.3 and 3.2.4.

Definition 22: D-convexity, D-concavity

A copula C is called directionally convex (D-convex) [Alvoni et al., 2007] if for every $(u_1, \ldots, u_n) \in [0,1]^n$, $(v_1, \ldots, v_n) \in [0,1]^n$, $i \in [\![1, n]\!]$ and $t \in [0,1]$ it verifies:

$$\begin{aligned} C(u_1, \ldots, tu_i + (1-t)v_i, \ldots, u_n) \leqslant\ & t \cdot C(u_1, \ldots, u_i, \ldots, u_n) \\ & + (1-t) \cdot C(u_1, \ldots, v_i, \ldots, u_n) \end{aligned} \tag{7.1}$$

In other words, the copula is convex when fixing all but one of its variables. A copula is called directionally concave (D-concave) if the inequality is reversed.

D-convexity/D-concavity is quite common in known families of copulas. The following paragraphs detail this property for copulas presented in Table 2.1, in the case $n = 2$. As the copulas presented are symmetric regarding their variables, D-convexity/D-concavity is

only detailed for u_1. We assume that θ always belong to the domain of definition detailed in Table 2.1, and that u_1, u_2 are in $[0, 1]$. Regarding the Clayton and Gumbel families, the copula is defined by continuous extension in cases $u_1 = 0$ and $u_2 = 0$. Finally, if the restriction of a copula C to one of its variable u_i is twice differentiable, then proving its D-convexity for this variable can be done by proving that $\dfrac{\partial^2 C}{\partial {u_i}^2} \geqslant 0$.

Ali-Mikhail-Haq copula This copula is twice differentiable, and its second order partial derivative is

$$\frac{\partial^2 C}{\partial {u_1}^2} = u_2(1 - \theta(1 - u_2))\frac{-2\theta(1 - u_2)}{(1 - \theta(1 - u_1)(1 - u_2))^3}$$

Thus the Ali-Mikhail-Haq copula is D-convex for $\theta \in [-1, 0]$ and D-concave for $\theta \in [0, 1)$.

Clayton copula This copula is not always differentiable on all of its domain, depending on which value is retained by the maximum function. It is however continuous as it is the maximum of two continuous function. For convenience, we work with (u_1, u_2) in $\mathbb{I}^2$, where $\mathbb{I}$ is the open unit interval (the closed unit interval is then covered by continuity). Let $u_2 \in \mathbb{I}$. We split the possible range $\mathbb{I}$ of u_1 in two:

- the first domain $\mathcal{D}_1^{\theta,u_2}$ is where $u_1^{-\theta} + u_2^{-\theta} - 1 \leqslant 0$, and thus $C(u_1, u_2) = 0$. Here, $\frac{\partial^2 C}{\partial {u_1}^2} = 0$ and the copula is both D-convex and D-concave.
- the second domain $\mathcal{D}_2^{\theta,u_2}$ is where $u_1^{-\theta} + u_2^{-\theta} - 1 > 0$ and thus $C(u_1, u_2) \geqslant 0$. Here it holds that

$$\frac{\partial^2 C}{\partial {u_1}^2} = (1 + \theta)(1 - u_2^{-\theta})u_1^{-2-\theta}(u_1^{-\theta} + u_2^{-\theta} - 1)^{-2-\frac{1}{\theta}}$$

  Because of the definition of $\mathcal{D}_2^{\theta,u_2}$, the sign of $\frac{\partial^2 C}{\partial {u_1}^2}$ on $\mathcal{D}_2^{\theta,u_2}$ is that of $1 - u_2^{-\theta}$.

If $\theta > 0$, then $D_2^\theta = \mathbb{I}$ and $\frac{\partial^2 C}{\partial {u_1}^2} \leqslant 0$ which means that the copula is D-concave on all of its domain.

The case where $\theta < 0$ is less straightforward. In that case, $\frac{\partial^2 C}{\partial {u_1}^2} \geqslant 0$ on $\mathcal{D}_2^{\theta,u_2}$. The restrictions of the copula to $\mathcal{D}_1^{\theta,u_2}$ and $\mathcal{D}_2^{\theta,u_2}$ are both D-convex, but we need to prove that it is still true on their union. Let $u_1 \in \mathcal{D}_1^{\theta,u_2}$, $v_1 \in \mathcal{D}_2^{\theta,u_2}$ and $t \in [0, 1]$. We note $w_1 = (1 - u_2^{-\theta})^{-\frac{1}{\theta}}$, such that $\mathcal{D}_1^{\theta,u_2} =]0, w_1]$ and $\mathcal{D}_2^{\theta,u_2} =]w_1, 1[$. By continuity, C is

D-convex on $\mathcal{D}_2^{\theta,u_2} \bigcup \{w_1\}$. Because $u_1, w_1 \in \mathcal{D}_1^{\theta,u_2}$, it holds that:

$$\begin{aligned} tC(u_1,\ u_2) + (1-t)C(v_1,\ u_2) &= tC(w_1,\ u_2) + (1-t)C(v_1,\ u_2) \\ &\geqslant C(tw_1 + (1-t)v_1,\ u_2) \\ &\quad \text{by convexity of } C \text{ on } \mathcal{D}_2^{\theta,u_2} \bigcup \{w_1\} \\ &\geqslant C(tu_1 + (1-t)v_1,\ u_2) \\ &\quad \text{because } C \text{ is component-wise increasing} \end{aligned}$$

which, by definition 22, proves that C is D-convex on $\mathcal{D}_1^{\theta,u_2} \bigcup \mathcal{D}_2^{\theta,u_2}$. By continuity, C is D-convex on all of its domain.

Frank copula This copula is twice differentiable, and its second order partial derivative is

$$\frac{\partial^2 C}{\partial {u_1}^2} = \frac{(e^{-\theta u_2} - 1)e^{-\theta u_1}\theta(e^{-\theta u_2} - e^{-\theta})}{(e^{-\theta} - 1 + (e^{-\theta u_1} - 1)(e^{-\theta u_2} - 1))^2}$$

If $\theta \geqslant 0$ then $\frac{\partial^2 C}{\partial {u_1}^2} \leqslant 0$ and C is D-concave. If $\theta \leqslant 0$ then $\frac{\partial^2 C}{\partial {u_1}^2} \geqslant 0$ and C is D-convex.

Gumbel copula This copula is twice differentiable on $]0,1]^2$, and its second order partial derivative is

$$\frac{\partial^2 C}{\partial {u_1}^2} = -\theta \frac{u_2}{u_1} \ln(u_2)(1 - \theta \ln(u_2)) e^{-\theta \ln(u_1) \ln(u_2)}$$

It holds that for all $\theta \in (0,1]$, $\frac{\partial^2 C}{\partial {u_1}^2} \geqslant 0$. By continuity, C is always D-convex.

As there is no explicit formula for the family of multivariate Gaussian copulas, it is difficult to prove its D-concavity or D-convexity. However, numerical approximations in the case $n = 2$ seem to indicate that a Gaussian copula would be D-convex if its marginals are positively correlated, and D-concave if they are negatively correlated. Figure 7.1 present those observations, with solid lines representing positive correlation, and dashed lines representing negative correlations. In the case $n > 2$, the copula can be neither D-convex nor D-concave, depending on the value of the correlation matrix. An example of this statement is provided in Figure 7.2.

The rest of this section will present different results regarding D-convex and D-concave copulas, leading to the main result of this section presented in proposition 12.

> **Remark:** All D-convex copulas C are dominated by the product copula. Similarly, all D-concave copulas dominate the product copula.

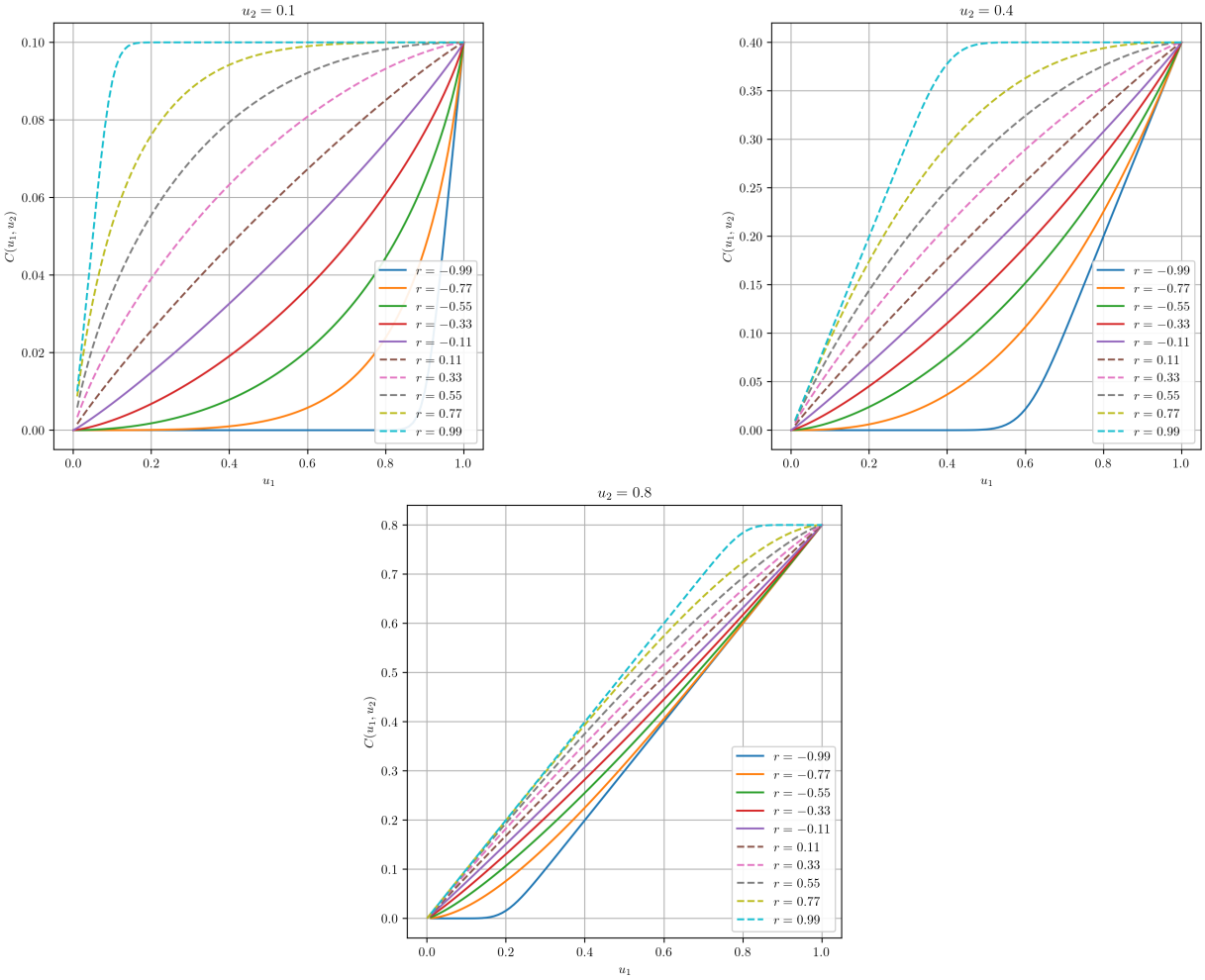


Figure 7.1: Gaussian 2-copulas in the direction u_1 for different u_2. Each figure present different plots for correlations r between u_1 and u_2 ranging in $[-1, 1]$. Solid lines represent negative correlation, while dashed lines represent positive correlations.

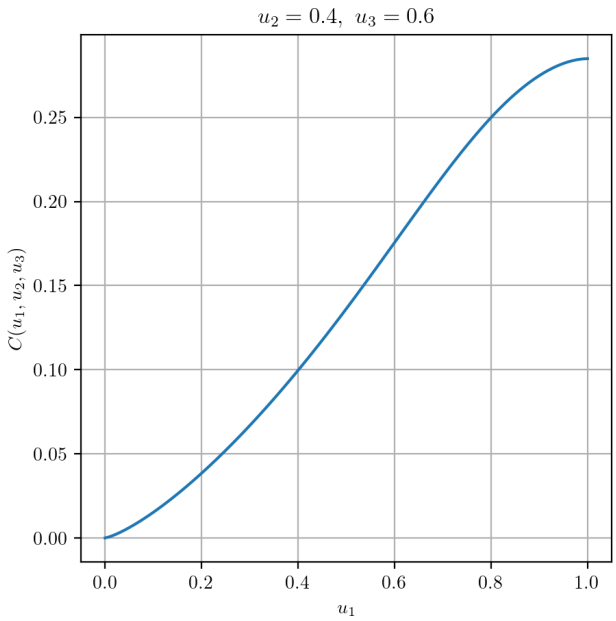


Figure 7.2: Directional cut of a Gaussian 3-copula with in the direction u_1, with $R = \begin{bmatrix} 1 & -0.4 & 0.7 \\ -0.4 & 1 & 0.3 \\ 0.7 & 0.3 & 1 \end{bmatrix}$, $u_2 = 0.4$ and $u_3 = 0.6$. The copula is neither D-convex nor D-concave

Consider a D-convex copula C, and let $(u_1, \ldots, u_n) \in [0,1]^n$.

$$\begin{aligned} C(u_1, \ldots, u_n) &= C((1-u_1)\cdot 0 + u_1 \cdot 1, u_2, \ldots, u_n) \\ &\leqslant (1-u_1)C(0, u_2, \ldots, u_n) + u_1 C(1, u_2, \ldots, u_n) \\ &\leqslant u_1 C(1, u_2, \ldots, u_n) \end{aligned}$$

Repeating this for $u_2, \ldots, u_n$ yields:

$$C(u_1, \ldots, u_n) \leqslant u_1 \ldots u_n C(1, \ldots, 1) = u_1 \ldots u_n = C_\Pi(u_1, \ldots, u_n)$$

The proof for D-convexity is similar.

Proposition 11: If C is a D-convex copula, then it verifies for all $(u_1, \ldots, u_n) \in [0,1]^n$, $(v_1, \ldots, v_n) \in [0,1]^n$ *s.t.* $\forall i \in [\![1, n]\!]$, $u_i + v_i \leqslant 1$:

$$\begin{aligned} C(u_1, \ldots, u_i + v_i, \ldots, u_n) \geqslant & \quad C(u_1, \ldots, u_i, \ldots, u_n) \\ & + C(u_1, \ldots, v_i, \ldots, u_n) \end{aligned} \tag{7.2}$$

Similarly, if $\forall i \in [\![1, n]\!]$, $u_i - v_i \geqslant 0$, it verifies:

$$\begin{aligned} C(u_1, \ldots, u_i - v_i, \ldots, u_n) \leqslant & \quad C(u_1, \ldots, u_i, \ldots, u_n) \\ & - \quad C(u_1, \ldots, v_i, \ldots, u_n) \end{aligned} \tag{7.3}$$

The inequalities are reversed for D-concave copulas.

Proof: Let $(u_1, \ldots, u_n) \in [0,1]^n$, $(v_1, \ldots, v_n) \in [0,1]^n$ *s.t.* $\forall i \in [\![1, n]\!]$, $u_i + v_i \leqslant 1$. Let $i \in [\![1, n]\!]$. Applying the definition of convexity (7.1) with $v_i = 0$ yields:

$$\forall t \in [0,1],\ C(u_1, \ldots, tu_i, \ldots, u_n) \leqslant tC(u_1, \ldots, u_n)$$

Let $w_i = u_i + v_i \in]0, 1]$ (the case where $u_i = 0$ or $v_i = 0$ is trivial). It is possible to write $u_i = tw_i$ and $v_i = (1-t)w_i$, with $t = \frac{u_i}{w_i} \in [0,1]$. Then it holds that:

$$\begin{aligned} C(u_1, \ldots, u_i, \ldots, u_n) = & \quad C(u_1, \ldots, t \cdot w_i, \ldots, u_n) \\ \leqslant & \quad t \cdot C(u_1, \ldots, w_i, \ldots, u_n) \\ C(u_1, \ldots, v_i, \ldots, u_n) = & \quad C(u_1, \ldots, (1-t) \cdot w_i, \ldots, u_n) \\ \leqslant & \quad (1-t) \cdot C(u_1, \ldots, w_i, \ldots, u_n) \end{aligned}$$

Summing the above equations yields:

$$\begin{aligned} C(u_1, \ldots, u_i, \ldots, u_n) + C(u_1, \ldots, v_i, \ldots, u_n) & \leqslant C(u_1, \ldots, w_i, \ldots, u_n) \\ & \leqslant C(u_1, \ldots, u_i + v_i, \ldots, u_n) \end{aligned}$$

which proves (7.2).

Let $w_i = u_i - v_i \in [0,1]$, clearly $w_i + v_i \leqslant 1$. Using proposition 11, it holds that:

$$\begin{aligned} & C(u_1, \ldots, w_i, \ldots, u_n) + \\ & C(u_1, \ldots, v_i, \ldots, u_n) && \leqslant C(u_1, \ldots, w_i + v_i, \ldots, u_n) \\ \Leftrightarrow\ & C(u_1, \ldots, w_i, \ldots, u_n) && \leqslant C(u_1, \ldots, w_i + v_i, \ldots, u_n) \\ & && \quad - C(u_1, \ldots, v_i, \ldots, u_n) \\ \Leftrightarrow\ & C(u_1, \ldots, u_i - v_i, \ldots, u_n) && \leqslant C(u_1, \ldots, u_i, \ldots, u_n) \\ & && \quad - C(u_1, \ldots, v_i, \ldots, u_n) \end{aligned}$$

which proves (7.3). □

Proposition 12: If C is a D-convex copula, then it verifies for all $(u_1,\ \ldots,\ u_n) \in [0,1]^n$, $(v_1,\ \ldots,\ v_n) \in [0,1]^n$ *s.t.* $\forall i \in [\![1,n]\!]$, $u_i - v_i \geqslant 0$:

$$C(u_1 - v_1,\ \ldots,\ u_i - v_i,\ \ldots,\ u_n - v_n) \leqslant \quad H_{v_1,\ \ldots,\ v_i,\ \ldots,\ v_n}^{u_1,\ \ldots,\ u_i,\ \ldots,\ u_n} \tag{7.4}$$

where H is the H-volume of C. The inequality is reversed for D-concave copulas.

Proof: The result is straightforward by induction using proposition 11. □

7.2 Joining P-boxes Using D-Convex/Concave Copulas

This section contains the proof of Proposition 8 from Chapter 3. We remind here the proposition:

Proposition 13: When joining marginals represented by p-boxes using the natural ordering from (3.18) with a copula C, it holds that:

- if C is D-convex, then $\mathfrak{M}_{mass} \subseteq \mathfrak{M}_{agg}$.
- if C is D-concave, then $\mathfrak{M}_{agg} \subseteq \mathfrak{M}_{mass}$.

Proof: Consider n p-boxes $[\underline{F}_1,\ \overline{F}_1],\ \ldots,\ [\underline{F}_n,\ \overline{F}_n]$, a D-convex copula C and the natural order on focal sets $(a_k^i)_{1 \leqslant k \leqslant N_i}$ of each marginal p-box $[\underline{F}_i,\ \overline{F}_i]$. We will denote m_C the joint mass functions obtained using Definition 19 and Bel_C its associated belief function. We will also refer to $\underline{P}$ as the lower probability associated with $\mathfrak{M}_{agg}$ from Definition 21.

When considering the natural order on focal sets of a p-box (3.18), it holds that for every focal set a_p^i, the set $\{k \mid a_k^i \subseteq a_p^i\}$ is composed of consecutive integers. In the following, we denote by $\underline{p}$ and $\overline{p}$ the lowest and highest indices of the focal sets included in a_p^i. This means that $\{a_{\underline{p}}^i,\ \ldots,\ a_{\overline{p}}^i\}$ is the set of all focal sets included in a_p^i.

Let $a_{p_1}^1,\ \ldots,\ a_{p_n}^n$ be focal sets of $m_1,\ \ldots,\ m_n$. We note $u_p^i = \sum_{k=1}^{p} m_i(a_k^i)$. It

then holds that:

$$\begin{aligned}\mathrm{Bel}_C(a^1_{p_1},\ \ldots,\ a^n_{p_n}) &= \sum_{a^1_{k_1}\subseteq a^1_{p_1}} \cdots \sum_{a^n_{k_n}\subseteq a^n_{p_n}} m_C(a^1_{k_1},\ \ldots,\ a^n_{k_n}) \\ &= \sum_{k_1=\underline{p}_1}^{\overline{p}_1} \cdots \sum_{k_n=\underline{p}_n}^{\overline{p}_n} m_C(a^1_{k_1},\ \ldots,\ a^n_{k_n}) \\ &= \sum_{k_1=\underline{p}_1}^{\overline{p}_1} \cdots \sum_{k_n=\underline{p}_n}^{\overline{p}_n} H^{u^1_{k_1},\ \ldots,\ u^n_{k_n}}_{u^1_{k_1-1},\ \ldots,\ u^n_{k_n-1}}\end{aligned}$$

As the H-volume is computed over a partitioning of $[u^1_{\underline{p}_1-1}, u^1_{\overline{p}_1}] \times \cdots \times [u^n_{\underline{p}_n-1}, u^n_{\overline{p}_n}]$, it is possible to greatly simplify the sums. The proof is the same as the proof of (3.4) except that the CDF is computed over $[u^1_{\underline{p}_1-1}, u^1_{\overline{p}_1}] \times \cdots \times [u^n_{\underline{p}_n-1}, u^n_{\overline{p}_n}]$ and not $[0,1]^n$. This yields:

$$\mathrm{Bel}_C(a^1_{p_1},\ \ldots,\ a^n_{p_n}) = H^{u^1_{\overline{p}_1},\ \ldots,\ u^n_{\overline{p}_n}}_{u^1_{\underline{p}_1-1},\ \ldots,\ u^n_{\underline{p}_n-1}}$$

On the other hand, it holds that:

$$\begin{aligned}\underline{P}(a^1_{p_1},\ \ldots,\ a^n_{p_n}) &= C(\mathrm{Bel}_1(a^1_{p_1}),\ \ldots,\ \mathrm{Bel}_n(a^n_{p_n})) \\ &= C(\sum_{k_1=\underline{p}_1}^{\overline{p}_1} m_1(a^1_{k_1}),\ \ldots,\ \sum_{k_n=\underline{p}_n}^{\overline{p}_n} m_1(a^n_{k_n})) \\ &= C(u^1_{\overline{p}_1} - u^1_{\underline{p}_1-1},\ \ldots,\ u^n_{\overline{p}_n} - u^n_{\underline{p}_n-1})\end{aligned}$$

Using proposition 12 yields:

$$\mathrm{Bel}_C(a^1_{p_1},\ \ldots,\ a^n_{p_n}) \geqslant \underline{P}(a^1_{p_1},\ \ldots,\ a^n_{p_n})$$

The inequality is reversed if C is D-concave, which concludes the proof. □

Figure 7.3 illustrates the difference between Bel_C and $\underline{P}$ in the case $n = 2$.

7.3 Consistency of the Median Filtering

This section demonstrates a result used in Section 5.1.3. We define the median of a set of n sorted values $X = \{x_1,\ \ldots,\ x_n\}$ is defined as

$$\text{if } n = 2l+1, \qquad \text{median}\, X = x_{l+1} \tag{7.5}$$

$$\text{if } n = 2l, \qquad \text{median}\, X = \frac{x_l + x_{l+1}}{2} \tag{7.6}$$

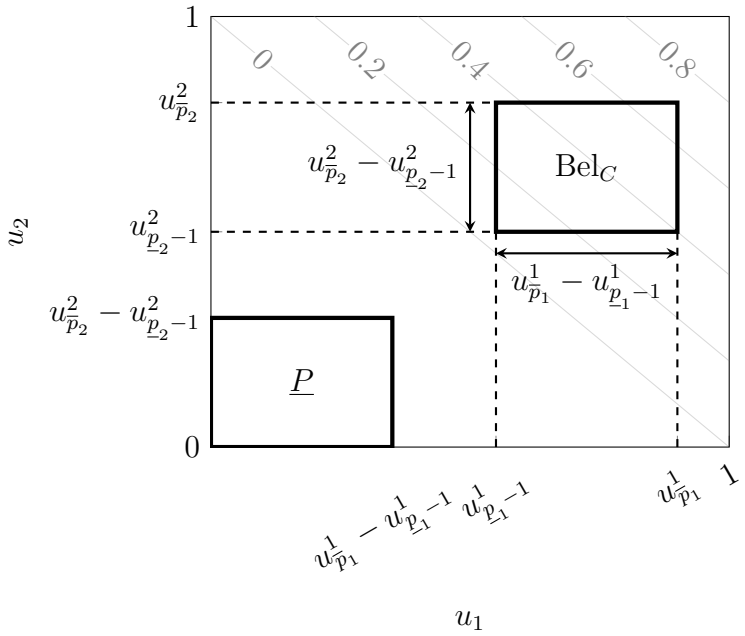


Figure 7.3: Bird's-eye view of the Łukaciewicz 2-copula C_L, where the gray lines are the isolines of the copula. Bel_C and $\underline{P}$ are represented in the case where the marginals are p-boxes. The thick rectangles represent the bounds on which to compute the H-volume. Numbers u^i_k use the notation of the proof of Proposition 8.

where l is an integer.

Proposition 14: Let $n \in \mathbb{N}^*$. Let $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_n\}$ be two sets of integers such that for all i in $[\![1, n]\!]$, $x_i \leqslant y_i$. Then:

$$\text{median}\, X \leqslant \text{median}\, Y \tag{7.7}$$

Proof: Let $\sigma_X : [\![1, n]\!] \to [\![1, n]\!]$ be a bijection sorting X, *i.e.*:

$$x_{\sigma_X(1)} \leqslant \cdots \leqslant x_{\sigma_X(i)} \leqslant \cdots \leqslant x_{\sigma_X(n)} \tag{7.8}$$

We define $\sigma_Y : [\![1, n]\!] \to [\![1, n]\!]$ similarly, this time sorting Y. Notice that it does not necessarily holds that $x_{\sigma_X(i)} \leqslant y_{\sigma_Y(i)}$, but only that $x_{\sigma_Y(i)} \leqslant y_{\sigma_Y(i)}$

Suppose that $n = 2l + 1$ where l is an integer. Then the median of X equals $x_{\sigma_X(l+1)}$ and the median of Y equals $y_{\sigma_Y(l+1)}$ It holds that for all $i \in [\![1, l + 1]\!]$:

$$y_{\sigma_Y(l+1)} \geqslant y_{\sigma_Y(i)} \geqslant x_{\sigma_Y(i)} \tag{7.9}$$

The median of Y is thus greater than at least $l + 1$ elements of X. Because the

$(l+1)^{\text{th}}$ smallest element of X is the median of X, then the median of Y is necessarily greater than the median of X.

The case $n = 2l$, is somehow similar. With the same arguments, we can say that $y_{\sigma_Y(l+1)}$ is greater than $l+1$ elements of X, thus it is greater than its $(l+1)^{\text{th}}$ smallest element $x_{\sigma_X(l+1)}$. Similarly, $y_{\sigma_Y(l)}$ is greater than l elements of X, thus it is greater than its l^{th} smallest element $x_{\sigma_X(l)}$. Therefore, it holds that:

$$\frac{y_{\sigma_Y(l)} + y_{\sigma_Y(l+1)}}{2} \geqslant \frac{x_{\sigma_X(l)} + x_{\sigma_X(l+1)}}{2} \tag{7.10}$$

Which also means that the median of Y is necessarily greater than the median of X. □

7.4 Ablation Studies for Disparity Confidence Intervals

This section will present ablation studies regarding the different parameters used to create disparity confidence intervals in Chapter 5. When not specified, the values of the different parameters are fixed to the same values used in our experiments, *i.e.*:

$\alpha = 0.9$ $\quad$ $q = 0.9$

$\tau_{amb} = 0.6$ $\quad$ $k_{amb} = 2$

$n_{\mathfrak{N}} = 2$

Those values were determined by evaluating different metrics on specific scenes of the Middlebury dataset. We present some results on the Middlebury Cones stereo images. In our experiments, we used the same values of parameters for every scene and for the two considered cost functions: CENSUS and MC-CNN. More in-depth analyses could lead to sets of parameter values specialized for each cost function. This is left as future work, considering that only the CENSUS cost function is used in Chapter 6.

7.4.1 Possibility Threshold

In Equation (5.5), we considered a parameter α used as a possibility threshold to compute a set of most possible disparities. We recall here the equations used in this section to compute confidence intervals:

$$D_\alpha = \{\ d \mid \pi_{row,\ col}(d) \geqslant \alpha\}$$
$$I_\alpha = [\underline{I}_\alpha,\ \overline{I}_\alpha] = [\min D_\alpha,\ \max D_\alpha]$$

Higher values of α will lead to smaller disparity sets, and thus to smaller intervals. In Chapters 5 and 6, we chose to use $\alpha = 0.9$, but other values could be considered. Figures 7.4 and 7.5 presents the evolution of metrics for different values of α. As expected, the accuracy decreases with higher values of α, as displayed in Figure 7.4(a).

Figure 7.4(b) displays the influence of α over the residual error ε. Variations of ε have a magnitude of around 2%, which indicates that α has very little influence over this metric. ε is not monotone with regard to α. This can be explained as follows: as α increases, the size of intervals decreases. There are thus more intervals that do not contain the ground truth, effectively modifying the set for which ε is the median. There is therefore no guarantee that ϵ is monotone with regard to α.

Figure 7.5(a) displays the influence of α over the relative size s_{rel} of intervals in high confidence areas. α does not have an influence over s_{rel} for the CENSUS cost function, and small influence for the MC-CNN cost function (fluctuations of 3% only).

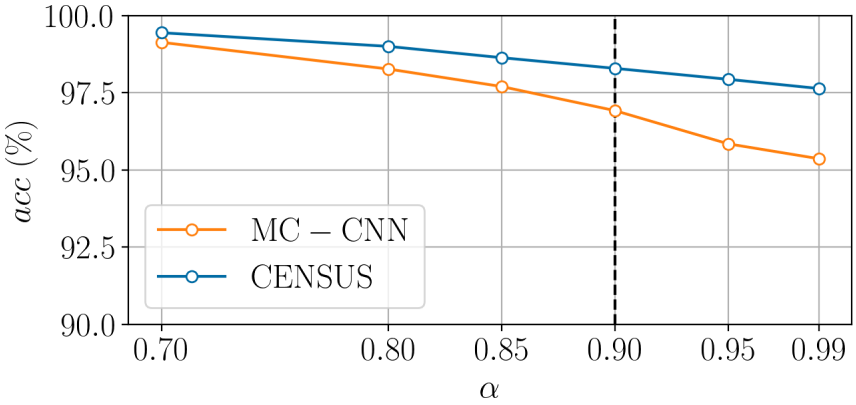
In Figure 7.5(b), we can observe that the relative over-estimation also decreases with higher values of α. For the CENSUS cost function, the relative over-estimation o_{rel} decreases less rapidly for values $\alpha \geqslant 0.9$. $\alpha = 0.9$ ensures that the relative over-estimation is less than 50% for both cost functions, which we consider a satisfying size for intervals in low confidence areas.

7.4.2 Low Confidence Areas

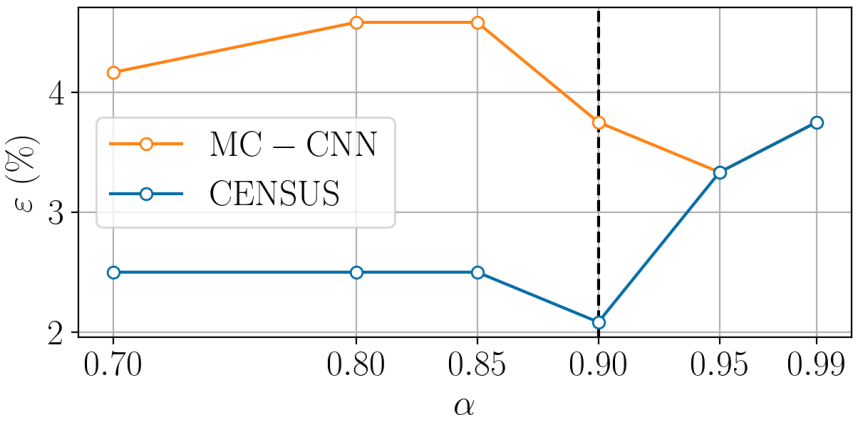
In Chapter 5, we noticed that intervals that did not contain the ground truth were usually located in low confidence areas. We thus proposed to detect low confidence areas in order to process the intervals differently there. We recall Equations (5.11) and (5.13) used to define low confidence areas. A pixel $(row,\ col)$ is considered to be in a low confidence area if it verifies:

$$\min c_{amb}(row,\ col + k) = \min_{-k_{amb} \leqslant k \leqslant k_{amb}} c_{amb}(row,\ col + k) \leqslant \tau_{amb}$$

This equation translates the fact that we use a minitive kernel to filter the confidence from ambiguity map, of size $(1, 2 \times k_{amb} + 1)$. High values of k_{amb} lead to lower values of the confidence from ambiguity, and consequently to more low confidence pixels. Similarly, higher values of the threshold τ_{amb} lead to more low confidence pixels. In Section 5.1.4, we also considered low confidence neighboring $\mathfrak{N}(row,\ col)$ of a low confidence pixel, defined

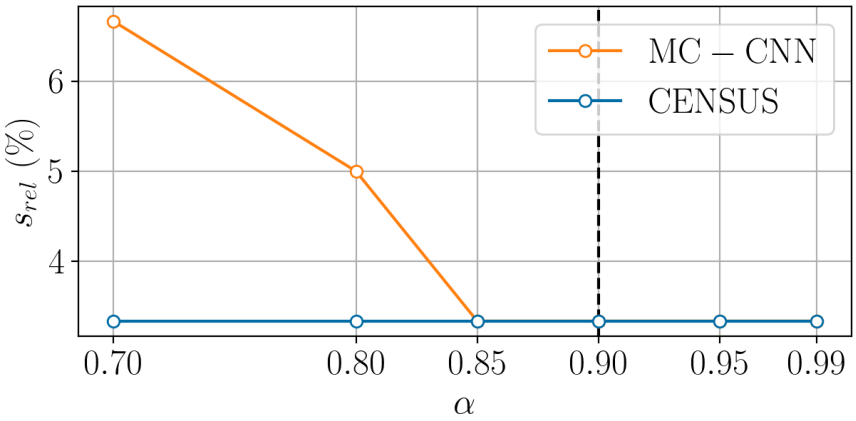


(a) Accuracy *acc* for different values of the parameter α for constructing intervals.

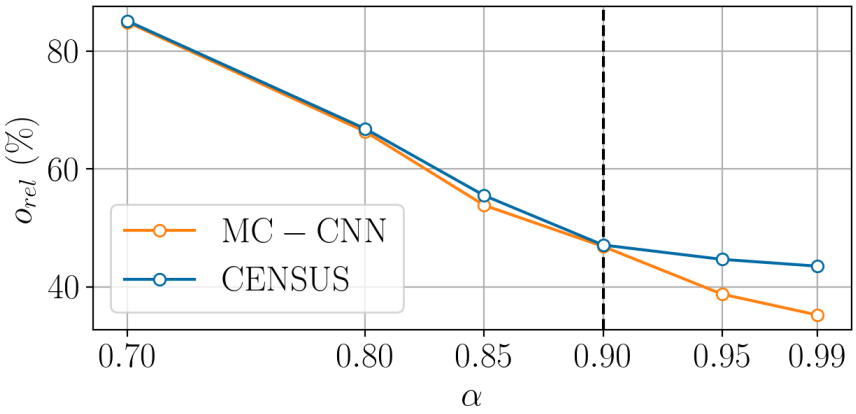


(b) Residual error ε for different values of the parameter α for constructing intervals.

Figure 7.4: Influence of the parameter α used for constructing intervals in Equation (5.5) for Middlebury Cones stereo images. Intervals were regularized in low confidence areas.



(a) Relative size s_{rel} for different values of the parameter α for constructing intervals.



(b) Relative over-estimation o_rel for different values of the parameter α for constructing intervals.

Figure 7.5: Influence of the parameter α used for constructing intervals in Equation (5.5) for Middlebury Cones stereo images. Intervals were regularized in low confidence areas.

as:

$$\mathfrak{N}(row,\ col) = \{p \in \bigcup_{-n_{\mathfrak{N}} \leqslant k \leqslant n_{\mathfrak{N}}} S(row+k,\ col_k) \\ s.t.\ S(row+(k+1),\ col_{k+1}) \text{ is adjacent to } S(row+k,\ col_k)\}$$

where S is defined as:

$$S(row,\ col) = \{\ (row,\ col')\ s.t.\ \forall c \in [\![col, col']\!],\ \min c_{amb}(row,\ c+k) \leqslant \tau_{amb}\ \}$$

Given a low confidence pixel $(row,\ col)$, the parameter $n_{\mathfrak{N}}$ defines the number of rows above and below a pixel that will be explored when defining its neighboring. High values of $n_{\mathfrak{N}}$ will lead to neighboring with more low confidence pixels, but does not influence the total amount of low confidence pixels present in an image.

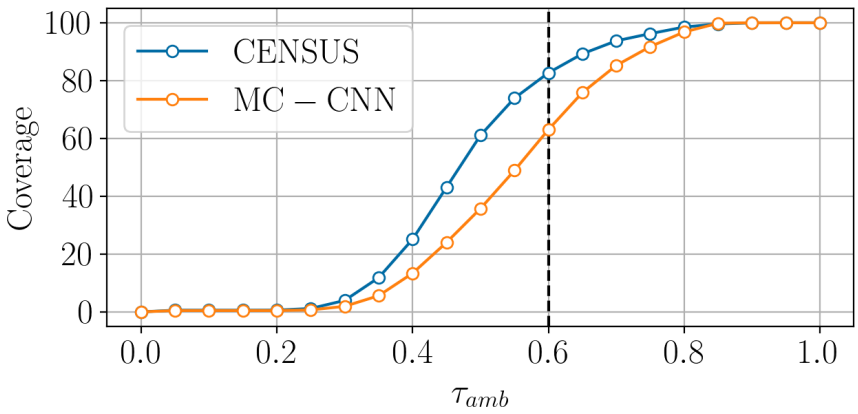
In Chapters 5 and 6, we chose to use $\tau_{amb} = 0.6$, $k_{amb} = 2$ and $n_{\mathfrak{N}} = 2$. We detail here the influence of those parameters.

We want low confidence areas to include as many wrong intervals as possible, but they should not be covering the entire image either. We also want to consider true intervals in the low confidence areas so that our quantile regularization is based on a sufficient pool of correct intervals. For this reason, computing the F1-score is not relevant. We therefore consider two metrics separately: the coverage and the proportion of low confidence pixels. The coverage is the proportion of wrong intervals that are also low confidence pixels:

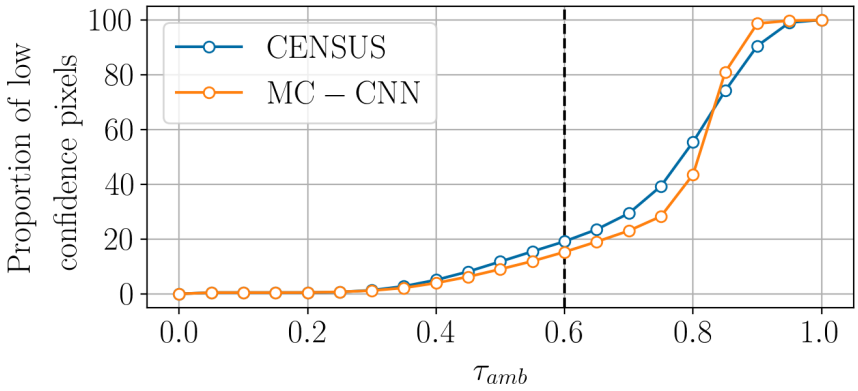
$$\text{Coverage} = \frac{\#\{(row,\ col)\ s.t.\ d_{true} \notin I_\alpha \text{ and } \min c_{amb}(row,\ col+k) \leqslant \tau_{amb}\}}{\#\{row,\ col)\ s.t.\ d_{true} \notin I_\alpha\}}$$

The proportion of low confidence pixels is simply the proportion of low confidence pixels in the entire left image. Figures 7.6(a) and 7.6(b) respectively display the evolution of the coverage and of the proportion of low confidence pixels for different values of τ_{amb}. Figures 7.7(a) and 7.7(b) display the same metrics, but for different values of k_{amb}. We can see that the value $\tau_{amb} = 0.6$ guarantees a coverage superior to 60% while maintaining a proportion of low confidence pixels smaller than 20% in the case of Middlebury Cones, for both considered cost functions. Similarly, $k_{amb} = 2$ guarantees a coverage superior to 60% while maintaining a proportion of low confidence pixels smaller than 20% in the case of Middlebury Cones, for both considered cost functions. In general, the coverage and proportion of low confidence pixels are more sensitive to a variation of 0.1 of τ_{amb} than a variation of 1 of the kernel size parameter k_{amb}.

Figure 7.8 displays the influence of $n_{\mathfrak{N}}$ over the accuracy, and over the number of pixels in low confidence neighboring $n_{\mathfrak{N}}$. Figure 7.8(a) shows that $n_{\mathfrak{N}}$ increases the global

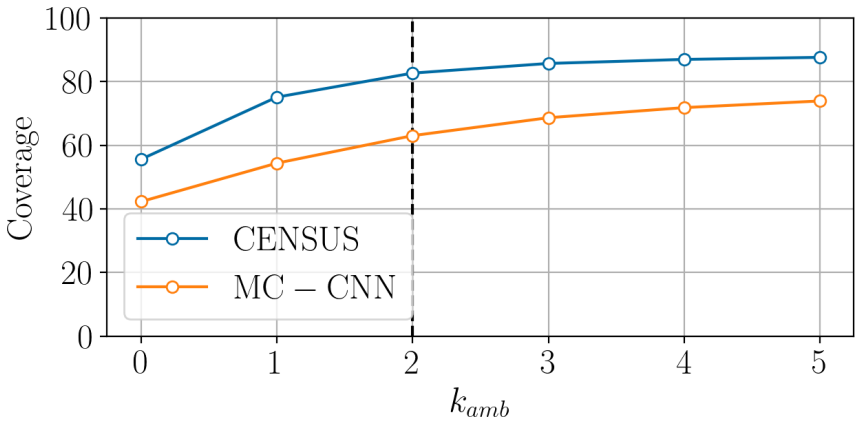


(a) Coverage for different values of the parameter τ_{amb} for detecting wrong intervals in low confidence areas.

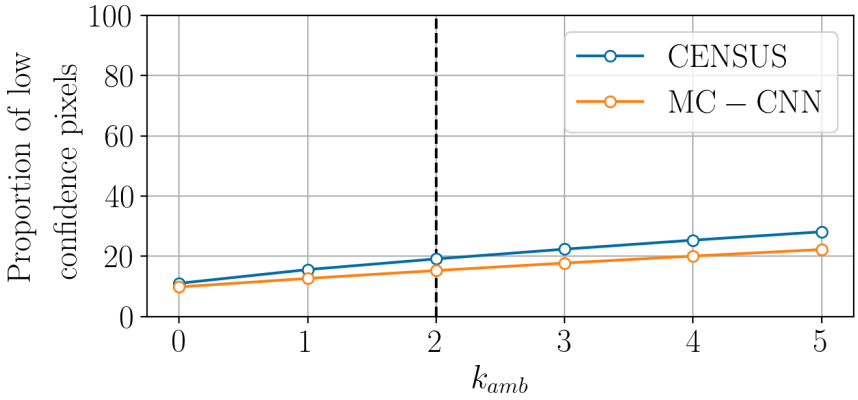


(b) Proportion of pixels detected as low confidence pixels depending on the value τ_{amb}

Figure 7.6: Influence of the parameter τ_{amb} used for interval regularization in Equation (5.11) for Middlebury Cones stereo images.



(a) Coverage for different values of the parameter k_{amb} for detecting wrong intervals in low confidence areas.



(b) Proportion of pixels detected as low confidence pixels depending on the value k_{amb}

Figure 7.7: Influence of the parameter k_{amb} used for interval regularization in Equation (5.11) for Middlebury Cones stereo images.

accuracy, but it is less sensitive to variations of $n_{\mathfrak{N}}$ than it is to variation of α from Figure 7.4(a). Figures 7.8(b) and 7.8(c) shows that $n_{\mathfrak{N}} \geqslant 2$ strongly reduces the number of low confidence neighboring with less than 50 pixels. However, high values of $n_{\mathfrak{N}}$ increases the computation complexity for a small gain of accuracy. We therefore choose $n_{\mathfrak{N}} = 2$ as a trade-off.

7.4.3 Quantile Regularization

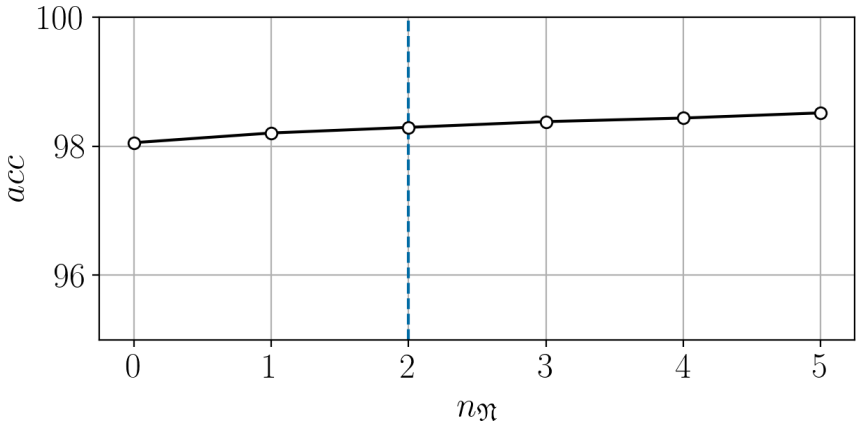
Once the neighboring $\mathfrak{N}$ of a low confidence pixel has been computed, we update the value of its confidence interval based on the distributions of intervals of its neighbors. We recall here Equation (5.14) used for the regularization of intervals:

$$\begin{aligned} I_{\alpha}^{reg} = [&\mathcal{Q}_{1-q}(\{\underline{I}_{\alpha}(r,\ c) \mid (r,\ c) \in \mathfrak{N}(row,\ col)\}), \\ &\mathcal{Q}_{q}(\{\overline{I}_{\alpha}(r,\ c) \mid (r,\ c) \in \mathfrak{N}(row,\ col)\})] \end{aligned}$$

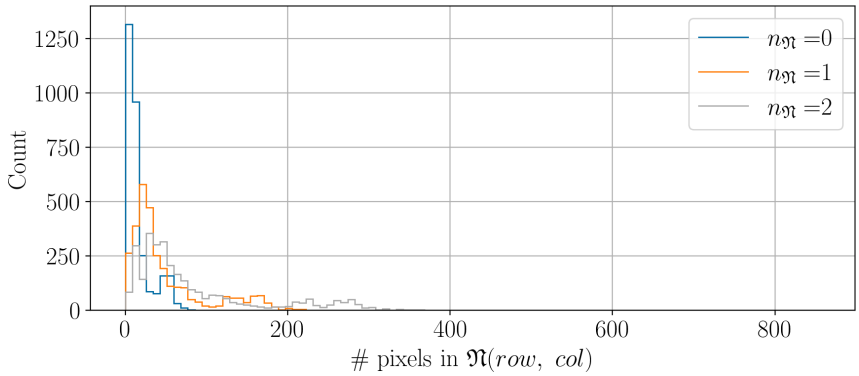
where $\mathcal{Q}_q$ refers to the q^{th} quantile of a set. In our experiments, we used $q = 0.9$.

Figure 7.9 displays the influence of the parameter q over the accuracy and relative over-estimation for the Middlebury Cones stereo images. As expected, both the accuracy and relative over-estimation increase with the value of q.

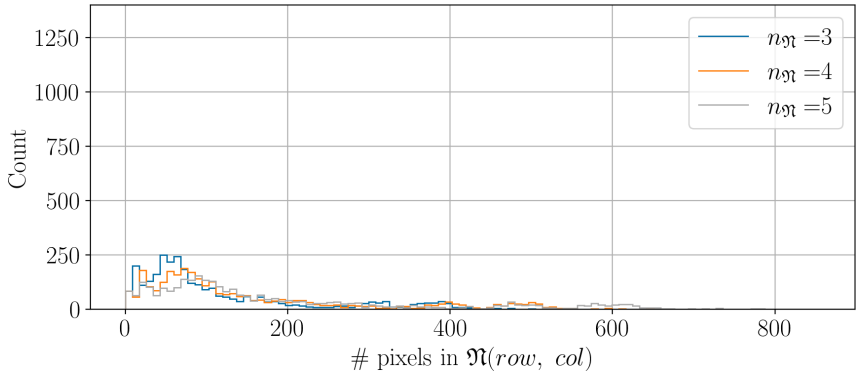
Figure 7.10 displays the accuracy with and without regularization for scenes of the Middlebury dataset. Without the regularization, many scenes do not reach an accuracy of 90%. This is especially true for the CENSUS cost function. This justifies the use of the regularization step from Section 5.1.4.



(a) Accuracy acc for different values of $n_{\mathfrak{N}}$

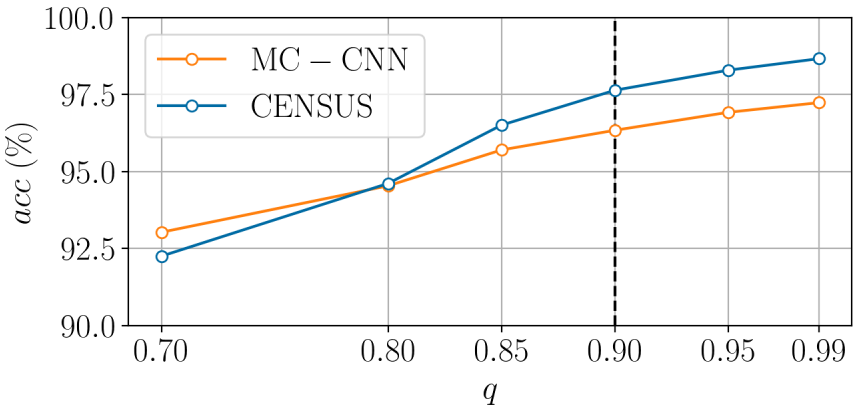


(b) Histograms of the number of pixels in low confidence areas $\mathfrak{N}(row,\ col)$ for $n_{\mathfrak{N}} \in \{1, 2, 3\}$.

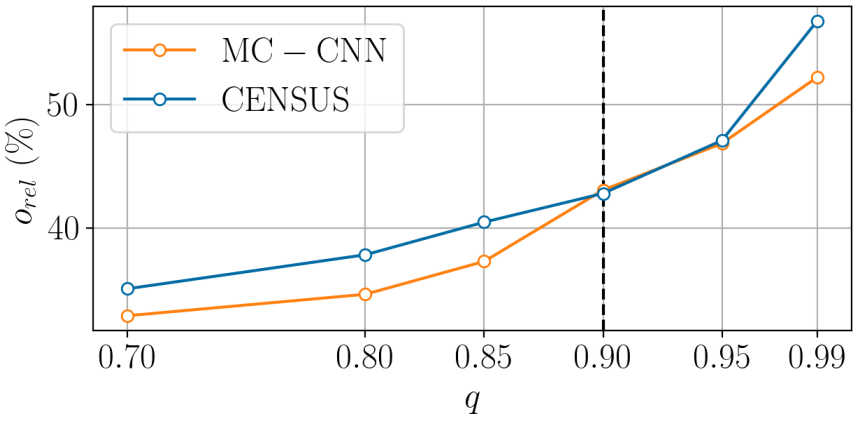


(c) Histograms of the number of pixels in low confidence areas $\mathfrak{N}(row,\ col)$ for $n_{\mathfrak{N}} \in \{4, 5, 6\}$.

Figure 7.8: Influence of the parameter $n_{\mathfrak{N}}$ used for interval regularization in Equation (5.13) using the CENSUS cost function on Middlebury cones.

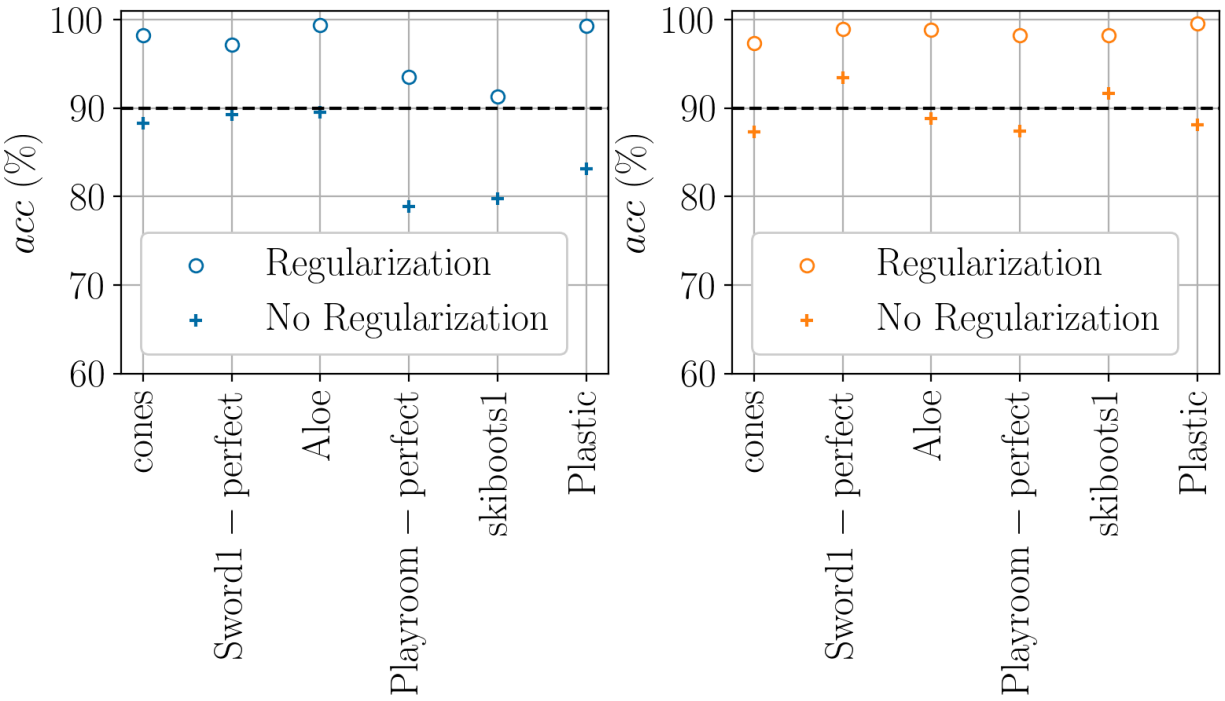


(a) Accuracy *acc* with regularization for different quantiles q



(b) Relative over-estimation o_{rel} with regularization for different quantiles q

Figure 7.9: Influence of the parameter q used for interval regularization in Equation (5.14) for Middlebury Cones stereo images.



(a) CENSUS cost function

(b) MC-CNN cost function

Figure 7.10: Accuracy with and without regularization of intervals in low confidence areas for different scenes of the Middlebury dataset.

Bibliography

[Alvoni et al., 2007] Alvoni, E., Durante, F., Papini, P. L., and Sempi, C. (2007). Different types of convexity and concavity for copulas. In *Proceedings of the 5th EUSFLAT Conference, Ostrava, Czech Republic*, volume 1, page 6.

[Augustin et al., 2014] Augustin, T., Coolen, F. P., de Cooman, G., and Troffaes, M. C. M. (2014). *Introduction to Imprecise Probabilities*. Wiley.

[Baltsavias and Stallmann, 1992] Baltsavias, E. P. and Stallmann, D. (1992). Metric Information Extraction from Spot Images and the Role of Polynomial Mapping Functions. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXIX.

[Bárdossy et al., 1995] Bárdossy, A., Bronsterts, A., and Merz, B. (1995). l-, 2- and 3-dimensional modeling of water movement in the unsaturated soil matrix using a fuzzy approach. *Advances in Water Resources*.

[Baudrit, 2005] Baudrit, C. (2005). *Représentation et propagation de connaissances imprécises et incertaines: Application à l'évaluation des risques liés aux sites et sols pollués*. math, Université Paul Sabatier - Toulouse III.

[Baudrit and Dubois, 2005] Baudrit, C. and Dubois, D. (2005). Comparing Methods for Joint Objective and Subjective Uncertainty Propagation with an Example in a Risk Assessment. In *4th International Symposium On Imprecise Probability: Theories And Applications (ISIPTA 2005)*, pages 31–40.

[Baudrit et al., 2007] Baudrit, C., Guyonnet, D., and Dubois, D. (2007). Joint propagation of variability and imprecision in assessing the risk of groundwater contamination. *Journal of Contaminant Hydrology*, 93(1-4):72–84.

[Beer et al., 2013] Beer, M., Ferson, S., and Kreinovich, V. (2013). Imprecise probabilities in engineering analyses. *Mechanical Systems and Signal Processing*, 37(1-2):4–29.

[Berthier et al., 2014] Berthier, E., Vincent, C., Magnússon, E., Gunnlaugsson, Á. Þ., Pitte, P., Le Meur, E., Masiokas, M., Ruiz, L., Pálsson, F., Belart, J. M., and Wagnon, P. (2014). Glacier topography and elevation changes derived from Pléiades sub-meter stereo images. *The Cryosphere*, 8(6):2275–2291.

[Bolles et al., 1987] Bolles, R. C., Baker, H. H., and Marimont, D. H. (1987). Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision*, 1(1):7–55.

[Bosch et al., 2018] Bosch, M., Foster, K., Christie, G., Wang, S., Hager, G. D., and Brown, M. (2018). Semantic Stereo for Incidental Satellite Images.

[Bosch et al., 2017] Bosch, M., Leichtman, A., Chilcott, D., Goldberg, H., and Brown, M. (2017). Metric Evaluation Pipeline for 3D Modeling of Urban Scenes. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-1/W1:239–246.

[Boudet, 2007] Boudet, L. (2007). *Auto-qualification de données géographiques 3D par appariement multi-image et classification supervisée : application au bâti en milieu urbain dense*. PhD Thesis, Université de Marne-la-Vallée.

[Boykov et al., 1998] Boykov, Y., Veksler, O., and Zabih, R. (1998). Markov random fields with efficient approximations. In *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231)*, pages 648–655, Santa Barbara, CA, USA. IEEE Comput. Soc.

[Buades and Facciolo, 2015] Buades, A. and Facciolo, G. (2015). Reliable Multiscale and Multiwindow Stereo Matching. *SIAM Journal on Imaging Sciences*, pages 888–915.

[Chebbi et al., 2023] Chebbi, M. A., Rupnik, E., Pierrot-Deseilligny, M., and Lopes, P. (2023). DeepSim-Nets: Deep Similarity Networks for Stereo Image Matching. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2097–2105, Vancouver, BC, Canada. IEEE.

[Chen et al., 2023] Chen, L., Wang, W., and Mordohai, P. (2023). Learning the Distribution of Errors in Stereo Matching for Joint Disparity and Uncertainty Estimation. arXiv:2304.00152 [cs].

[Cherubini et al., 2004] Cherubini, U., Luciano, E., and Vecchiato, W. (2004). *Copula Methods in Finance*. Wiley, 1 edition.

[CNES et al., 2008] CNES, ONERA, and IGN (2008). *Imagerie Spatiale. Des principes d'acquisition au traitement des images optiques pour l'observation de la Terre*. Editions Cépaduès.

[Coeurdevey and Gabriel-Robez, 2012] Coeurdevey, L. and Gabriel-Robez, C. (2012). Pléiades Imagery User Guide.

[Coffer et al., 2022] Coffer, M. M., Whitman, P. J., Schaeffer, B. A., Hill, V., Zimmerman, R. C., Salls, W. B., Lebrasse, M. C., and Graybill, D. D. (2022). Vertical artifacts in high-resolution WorldView-2 and WorldView-3 satellite imagery of aquatic systems. *International journal of remote sensing*, 43(4):1199–1225. Place: England.

[Cournet et al., 2020] Cournet, M., Sarrazin, E., Dumas, L., Michel, J., Guinet, J., Youssefi, D., Defonte, V., and Fardet, Q. (2020). Ground Truth Generation and Disparity Estimation for Optical Satellite Imagery. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B2-2020:127–134.

[Couso et al., 2001] Couso, I., Montes, S., and Gil, P. (2001). The Necessity Of The Strong α-Cuts Of A Fuzzy Set. *International Journal of Un-*

certainty, Fuzziness and Knowledge-Based Systems, 09(02):249–262. _eprint: https://doi.org/10.1142/S0218488501000788.

[Couso et al., 2000] Couso, I., Moral, S., and Walley, P. (2000). A survey of concepts of independence for imprecise probabilities. *Risk Decision and Policy*, 5(2):165–181.

[Czado and Nagler, 2022] Czado, C. and Nagler, T. (2022). Vine Copula Based Modeling. *Annual Review of Statistics and Its Application*, 9(1):453–477.

[de Cooman et al., 2010] de Cooman, G., Miranda, E., and Zaffalon, M. (2010). Factorisation Properties of the Strong Product. In Kacprzyk, J., Borgelt, C., González-Rodríguez, G., Trutschnig, W., Lubiano, M. A., Gil, M. Á., Grzegorzewski, P., and Hryniewicz, O., editors, *Combining Soft Computing and Statistical Methods in Data Analysis*, volume 77, pages 139–147. Springer Berlin Heidelberg, Berlin, Heidelberg. Series Title: Advances in Intelligent and Soft Computing.

[De Franchis et al., 2014] De Franchis, C., Meinhardt-Llopis, E., Michel, J., Morel, J.-M., and Facciolo, G. (2014). On stereo-rectification of pushbroom images. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 5447–5451, Paris, France. IEEE.

[de Joinville, 2001] de Joinville, O. (2001). *Evaluation de la qualité d'une cartographie urbaine à l'aide d'images aériennes à haute résolution*. PhD thesis, Ecole Nationale Supérieure des Télécommunications.

[Defonte et al., 2021] Defonte, V., Dumas, L., Cournet, M., and Sarrazin, E. (2021). Evaluation of MC-CNN Based Stereo Matching Pipeline for the CO3D Earth Observation Program. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pages 7670–7673.

[Delon and Rougé, 2007] Delon, J. and Rougé, B. (2007). Small Baseline Stereovision. *Journal of Mathematical Imaging and Vision*, 28(3):209–223.

[Delvit et al., 2006] Delvit, J.-M., Fave, P., and Gachet, R. (2006). The Geometric Supersite of Salon de Provence. In *ISPRS Marne-la-Vallée Workshop*.

[Dempster, 1967] Dempster, A. P. (1967). Upper and Lower Probabilities Induced by a Multivalued Mapping. *The Annals of Mathematical Statistics*.

[Deschamps-Berger, 2021] Deschamps-Berger, C. (2021). *Apport de la photogrammétrie satellite pour la modélisation du manteau neigeux*. PhD thesis, Université Toulouse 3 - Paul Sabatier.

[Destercke et al., 2008] Destercke, S., Dubois, D., and Chojnacki, E. (2008). Unifying practical uncertainty representations – I: Generalized p-boxes. *International Journal of Approximate Reasoning*, 49(3):649–663.

[Devika et al., 2006] Devika, S., Ramakrishnan S., S., Rao, S., and Murali Mohan, A. (2006). Developement and Implementation of Rational Polynomial Coefficient Algorithms for Georeferencing Cartosat-1 Data. In *ISPRS Archives*, volume XXXVI.

[Digne and De Franchis, 2017] Digne, J. and De Franchis, C. (2017). The Bilateral Filter

for Point Clouds. *Image Processing On Line*, 7:278–287.

[Dubois and Prade, 1992] Dubois, D. and Prade, H. (1992). When upper probabilities are possibility measures. *Fuzzy Sets and Systems*, 49(1):65–74.

[Dumas et al., 2022] Dumas, L., Defonte, V., Steux, Y., and Sarrazin, E. (2022). Improving Pairwise DSM With 3SGM: a Semantic Segmentation for SGM Using an Automatically Refined Neural Network. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, V-2-2022:167–175.

[Egnal et al., 2004] Egnal, G., Mintz, M., and Wildes, R. P. (2004). A stereo confidence metric using single view imagery with comparison to five alternative approaches. *Image and Vision Computing*, 22(12):943–957.

[Erdogan and Yilmaz, 2019] Erdogan, M. and Yilmaz, A. (2019). Detection of building damage caused by Van Earthquake using image and Digital Surface Model (DSM) difference. *International Journal of Remote Sensing*, 40(10):3772–3786. Publisher: Taylor & Francis.

[Ernst and Hirschmüller, 2008] Ernst, I. and Hirschmüller, H. (2008). Mutual Information Based Semi-Global Stereo Matching on the GPU. In Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Porikli, F., Peters, J., Klosowski, J., Arns, L., Chun, Y. K., Rhyne, T.-M., and Monroe, L., editors, *Advances in Visual Computing*, volume 5358, pages 228–239. Springer Berlin Heidelberg, Berlin, Heidelberg. Series Title: Lecture Notes in Computer Science.

[Facciolo et al., 2015] Facciolo, G., Franchis, C. d., and Meinhardt, E. (2015). MGM: A Significantly More Global Matching for Stereovision. In *Procedings of the British Machine Vision Conference 2015*, pages 90.1–90.12, Swansea. British Machine Vision Association.

[Farr et al., 2007] Farr, T. G., Rosen, P. A., Caro, E., Crippen, R., Duren, R., Hensley, S., Kobrick, M., Paller, M., Rodriguez, E., Roth, L., Seal, D., Shaffer, S., Shimada, J., Umland, J., Werner, M., Oskin, M., Burbank, D., and Alsdorf, D. (2007). The Shuttle Radar Topography Mission. *Reviews of Geophysics*, 45(2):2005RG000183.

[Ferson et al., 2004] Ferson, S., Oberkampf, W., Tucker, W., Zhang, J., Ginzburg, L., Berleant, D., Hajagos, J., and Nelsen, R. (2004). Dependence in probabilistic modeling, Dempster-Shafer theory, and probability bounds analysis. Technical Report SAND2004-3072, 919189, Sandia National Laboratories.

[Fouladinejad et al., 2019] Fouladinejad, F., Matkan, A., Hajeb, M., and Brakhasi, F. (2019). History and Applications of Space-Borne LiDARs. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-4/W18:407–414.

[Franchis et al., 2014] Franchis, C. d., Meinhardt, E., Michel, J., Morel, J.-M., and Facciolo, G. (2014). An automatic and modular stereo pipeline for pushbroom images. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3:49–56.

[Frank, 1999] Frank, M. V. (1999). Treatment of uncertainties in space nuclear risk as-

sessment with examples from Cassini mission applications. *Reliability Engineering & System Safety*, 66(3):203–221.

[Fua, 1991] Fua, P. (1991). Combining Stereo and Monocular Information to Compute Dense Depth Maps that Preserve Depth Discontinuities. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (II)*, pages 1292–1298.

[Ganci et al., 2022] Ganci, G., Cappello, A., and Neri, M. (2022). Data Fusion for Satellite-Derived Earth Surface: The 2021 Topographic Map of Etna Volcano. *Remote Sensing*, 15(1):198.

[Gao, 1996] Gao, B.-c. (1996). NDWI—A normalized difference water index for remote sensing of vegetation liquid water from space. *Remote Sensing of Environment*, 58(3):257–266.

[Gascoin et al., 2019] Gascoin, S., Grizonnet, M., Bouchet, M., Salgues, G., and Hagolle, O. (2019). Theia Snow collection: high-resolution operational snow cover maps from Sentinel-2 and Landsat-8 data. *Earth System Science Data*, 11(2):493–514.

[Geiger et al., 2012] Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? The KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, Providence, RI. IEEE.

[Geiger et al., 2013] Geiger, A., Lenz, Philip, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, 32(11):1231–1237.

[Geng, 1996] Geng, Z. J. (1996). Rainbow three-dimensional camera: new concept of high-speed three-dimensional vision systems. *Optical Engineering*, 35(2):376.

[Geudtner et al., 2014] Geudtner, D., Torres, R., Snoeij, P., Davidson, M., and Rommen, B. (2014). Sentinel-1 System capabilities and applications. *2014 IEEE Geoscience and Remote Sensing Symposium*, pages 1457–1460.

[Goldberg et al., 2002] Goldberg, S., Maimone, M., and Matthies, L. (2002). Stereo vision and rover navigation software for planetary exploration. In *Proceedings, IEEE Aerospace Conference*, volume 5, pages 5–5.

[Gray et al., 2021] Gray, A., Hose, D., de Angelis, M., Hanss, M., and Ferson, S. (2021). Dependent Possibilistic Arithmetic Using Copulas. In *Proceedings of the Twelveth International Symposium on Imprecise Probability: Theories and Applications*, Proceedings of Machine Learning Research, page 11. PMLR.

[Grodecki, 2001] Grodecki, J. (2001). IKONOS Stereo Feature Extraction - RPC Approach. *ASPRS Annual Convention*.

[Guo et al., 2024] Guo, X., Zhang, C., Lu, J., Wang, Y., Duan, Y., Yang, T., Zhu, Z., and Chen, L. (2024). OpenStereo: A Comprehensive Benchmark for Stereo Matching and Strong Baseline.

[Haala, 2014] Haala, N. (2014). Dense Image Matching Final Report. *European Spatial Data Research*, Official Publication No. 64.

[Haeusler et al., 2013] Haeusler, R., Nair, R., and Kondermann, D. (2013). Ensemble Learning for Confidence Measures in Stereo Vision. *2013 IEEE Conference on Computer Vision and Pattern Recognition*.

[Hagolle et al., 2017] Hagolle, O., Huc, M., Desjardins, C., Auer, S., and Richter, R. (2017). Maja Algorithm Theoretical Basis Document. *Zenodo*. Publisher: [object Object].

[Haller et al., 2010] Haller, I., Pantilie, C. D., Oniga, F., and Nedevschi, S. (2010). Real-time semi-global dense stereo solution with improved sub-pixel accuracy. In *2010 IEEE Intelligent Vehicles Symposium*, pages 369–376, La Jolla, CA, USA. IEEE.

[Hannah, 1994] Hannah, M. J. (1994). *Computer Matching of Areas in Stero Images*. PhD thesis, Stanford University Computer Science Department.

[Hariharan et al., 2014] Hariharan, B., Arbeláez, P. A., Girshick, R. B., and Malik, J. (2014). Hypercolumns for Object Segmentation and Fine-grained Localization. *CoRR*, abs/1411.5752. arXiv: 1411.5752.

[Hartley and Zisserman, 2004] Hartley, R. and Zisserman, A. (2004). *Multiple view geometry in computer vision*. Cambridge University Press, Cambridge, UK, second edition edition. OCLC: 171123855.

[Hirschmüller, 2005] Hirschmüller, H. (2005). Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 807–814, San Diego, CA, USA. IEEE.

[Hirschmuller, 2008] Hirschmuller, H. (2008). Stereo Processing by Semiglobal Matching and Mutual Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341.

[Hirschmüller et al., 2002] Hirschmüller, H., Innocent, P. R., and Garibaldi, J. (2002). Real-Time Correlation-Based Stereo Vision with Reduced Border Errors. *International Journal of Computer Vision*, page 18.

[Hirschmuller and Scharstein, 2007] Hirschmuller, H. and Scharstein, D. (2007). Evaluation of Cost Functions for Stereo Matching. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Minneapolis, MN, USA. IEEE.

[Hirschmuller and Scharstein, 2009] Hirschmuller, H. and Scharstein, D. (2009). Evaluation of Stereo Matching Costs on Images with Radiometric Differences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(9):1582–1599.

[Hora, 1996] Hora, S. C. (1996). Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management. *Reliability Engineering & System Safety*, 54(2):217–223.

[Hu and Mordohai, 2012] Hu, X. and Mordohai, P. (2012). A Quantitative Evaluation of Confidence Measures for Stereo Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2121–2133.

[Huang et al., 2022] Huang, X., Ren, L., Liu, C., Wang, Y., Yu, H., Schmitt, M., Hänsch, R., Sun, X., Huang, H., and Mayer, H. (2022). Urban Building Classification (UBC) - A Dataset for Individual Building Detection and Classification From Satellite Imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1413–1421.

[Hugonnet et al., 2022] Hugonnet, R., Brun, F., Berthier, E., Dehecq, A., Mannerfelt, E. S., Eckert, N., and Farinotti, D. (2022). Uncertainty Analysis of Digital Elevation Models by Spatial Inference From Stable Terrain. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:6456–6472.

[IGN, 1994] IGN (1994). BD TOPO® – Version 3.4 – Descriptif de contenu.

[IGN, 2013] IGN (2013). RGE ALTI® – Version 2.0 – Descriptif de contenu.

[IGN, 2022] IGN (2022). MicMac Documentation: MicMac, Apero, Pastis and Other Beverages in a Nutshell!

[IGN, 2024] IGN (2024). LiDAR HD – Version 1.0 – Descriptif de contenu.

[Irving, 1998] Irving, R. W. (1998). Matching Medical Students to Pairs of Hospitals: A New Variation on a Well-known Theme. In Bilardi, G., Italiano, G. F., Pietracaprina, A., and Pucci, G., editors, *Algorithms — ESA' 98*, pages 381–392, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Jacobsen et al., 2014] Jacobsen, K., Topan, H., Cam, A., Özendi, M., and Oruc, M. (2014). Radiometric and geometric characteristics of Pleiades images. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-1:173–177.

[Jasinski et al., 2020] Jasinski, M. F., Stoll, J. D., Hancock, D., Robbins, J., Nattala, J., Morison, J., Jones, B. M., Ondrusek, M. E., Pavelsky, T. M., and Parrish, C. (2020). *ATLAS/ICESat-2 L3A Inland Water Surface Height, Version 3, Boulder, Colorado USA*. National Snow and Ice Data Center. Publication Title: NASA National Snow and Ice Data Center Distributed Active Archive Center.

[Jaulin et al., 2001] Jaulin, L., Kieffer, M., Didrit, O., and Walter, É. (2001). *Applied Interval Analysis*. Springer London, London.

[Jenkins et al., 2023] Jenkins, L. T., Creed, M. J., Tarbali, K., Muthusamy, M., Trogrlić, R. Š., Phillips, J. C., Watson, C. S., Sinclair, H. D., Galasso, C., and McCloskey, J. (2023). Physics-based simulations of multiple natural hazards for risk-sensitive planning and decision making in expanding urban regions. *International Journal of Disaster Risk Reduction*, 84:103338.

[Ji et al., 2021] Ji, P., Li, J., Li, H., and Liu, X. (2021). Superpixel alpha-expansion and normal adjustment for stereo matching. *Journal of Visual Communication and Image Representation*, 79:103238.

[Jiang et al., 2024a] Jiang, L., Wang, F., Zhang, W., Li, P., You, H., and Xiang, Y. (2024a). Rethinking the Key Factors for the Generalization of Remote Sensing Stereo Matching Networks.

[Jiang et al., 2024b] Jiang, L., Xiang, Y., Wang, F., and You, H. (2024b). Unsupervised Stereo Matching Network For VHR Remote Sensing Images Based On Error Prediction.

[Joe, 1997] Joe, H. (1997). *Multivariate Models and Multivariate Dependence Concepts*. Chapman and Hall/CRC, New York, 1st edition edition.

[Kasser and Egels, 2001] Kasser, M. and Egels, Y. (2001). *Photogrammétrie numérique*. ENSG-IGN, ISSN 1628-4941. Paris: Hermès.

[Ke Zhang et al., 2009] Ke Zhang, Jiangbo Lu, and Lafruit, G. (2009). Cross-Based Local Stereo Matching Using Orthogonal Integral Images. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(7):1073–1079.

[Keselman et al., 2017] Keselman, L., Woodfill, J. I., Grunnet-Jepsen, A., and Bhowmik, A. (2017). Intel RealSense Stereoscopic Depth Cameras. Number: arXiv:1705.05548 arXiv:1705.05548 [cs].

[Khosravipour et al., 2016] Khosravipour, A., Skidmore, A. K., and Isenburg, M. (2016). Generating spike-free digital surface models using LiDAR raw point clouds: A new approach for forestry applications. *International Journal of Applied Earth Observation and Geoinformation*, 52:104–114.

[Kim et al., 2020] Kim, S., Min, D., Kim, S., and Sohn, K. (2020). Adversarial Confidence Estimation Networks for Robust Stereo Matching. *IEEE Transactions on Intelligent Transportation Systems*, PP:1–15.

[Kim et al., 2022] Kim, S., Poggi, M., Kim, S., Sohn, K., and Mattoccia, S. (2022). Meta-confidence estimation for stereo matching. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 10624–10631, Philadelphia, PA, USA. IEEE.

[Koh and Yang, 2016] Koh, J.-W. and Yang, H.-S. (2016). Unified piecewise epipolar resampling method for pushbroom satellite images. *EURASIP Journal on Image and Video Processing*, 2016(1):11.

[Kolmogorov and Zabih, 2001] Kolmogorov, V. and Zabih, R. (2001). Computing Visual Correspondence with Occlusions via Graph Cuts. *Proceedings of ICCV*, page 37.

[Krauß et al., 2013] Krauß, T., d'Angelo, P., Schneider, M., and Gstaiger, V. (2013). The Fully Automatic Optical Processing System Catena at DLR. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL.

[Krieger et al., 2007] Krieger, G., Moreira, A., Fiedler, H., Hajnsek, I., Werner, M., Younis, M., and Zink, M. (2007). TanDEM-X: A Satellite Formation for High-Resolution SAR Interferometry. *IEEE Transactions on Geoscience and Remote Sensing*, 45(11):3317–3341.

[Kuk-Jin Yoon and In-So Kweon, 2005] Kuk-Jin Yoon and In-So Kweon (2005). Locally Adaptive Support-Weight Approach for Visual Correspondence Search. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 924–931, San Diego, CA, USA. IEEE.

[Laga et al., 2022] Laga, H., Jospin, L. V., Boussaid, F., and Bennamoun, M. (2022).

A Survey on Deep Learning Techniques for Stereo-Based Depth Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(4):1738–1764.

[Lagouarde et al., 2019] Lagouarde, J.-P., Bhattacharya, B. K., Crebassol, P., Gamet, P., Adlakha, D., Murthy, C., Singh, S., Mishra, M., Nigam, R., Raju, P., Babu, S., Shukla, M., Pandya, M., Boulet, G., Briottet, X., Dadou, I., Dedieu, G., Gouhier, M., Hagolle, O., and Sarkar, S. (2019). Indo-French High-Resolution Thermal Infrared Space Mission for Earth Natural Resources Assessment and Monitoring – Concept and Definition of TRISHNA. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-3/W6:403–407.

[Lallement et al., 2022] Lallement, D., Lassalle, P., Ott, Y., Demortier, R., and Delvit, J. (2022). Bulldozer: An Automatic Self-Driven Large Scale DTM Extraction Method From Digital Surface Model. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XLIII-B2-2022, pages 409–415.

[Le Carrer and Ferson, 2021] Le Carrer, N. and Ferson, S. (2021). Beyond probabilities: A possibilistic framework to interpret ensemble predictions and fuse imperfect sources of information. *Quarterly Journal of the Royal Meteorological Society*, 147(739):3410–3433. _eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.4135.

[Le Saux et al., 2019] Le Saux, B., Yokoya, N., Hänsch, R., and Brown, M. (2019). Data Fusion Contest 2019 (DFC2019). *IEEE Dataport*. itemType: dataset.

[Lebègue et al., 2020] Lebègue, L., Cazala-Hourcade, E., Languille, F., Artigues, S., and Melet, O. (2020). CO3D, A Worldwide One-Meter Accuracy DEM For 2025. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B1-2020:299–304.

[Lebègue et al., 2015] Lebègue, L., Greslou, D., Blanchet, G., de Lussy, F., Fourest, S., Martin, V., Latry, C., Kubik, P., Delvit, J.-M., Dechoz, C., and Amberg, V. (2015). Pléiades-HR satellites image quality commissioning. *Revue Française de Photogrammétrie et de Télédétection*, 209:5–10.

[Lesniewska-Choquet, 2020] Lesniewska-Choquet, C. (2020). *Spécialité doctorale "STIC : Traitement de l'information"*. PhD thesis, Université Savoie-Mont-Blanc.

[Liu et al., 2024] Liu, C.-W., Chen, Q., and Fan, R. (2024). Playing to Vision Foundation Model's Strengths in Stereo Matching.

[Loghin et al., 2019] Loghin, A.-M., Otepka, J., Karel, W., Pöchtrager, M., and Pfeifer, N. (2019). Accuracy Analysis of Digital Elevation Models from very High Resolution Satellite Imagery. *Publ. Der DGPF*, 28:123–137.

[Loghin et al., 2020] Loghin, A.-M., Otepka-Schremmer, J., and Pfeifer, N. (2020). Potential of Pléiades and WorldView-3 Tri-Stereo DSMs to Represent Heights of Small Isolated Objects. *Sensors*, 20(9):2695.

[Loncan et al., 2015] Loncan, L., de Almeida, L. B., Bioucas-Dias, J. M., Briottet, X., Chanussot, J., Dobigeon, N., Fabre, S., Liao, W., Licciardi, G. A., Simoes, M., Tourneret, J.-Y., Veganzones, M. A., Vivone, G., Wei, Q., and Yokoya, N. (2015).

Hyperspectral Pansharpening: A Review. *IEEE Geoscience and Remote Sensing Magazine*, 3(3):27–46.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110.

[Malinowski and Destercke, 2022a] Malinowski, R. and Destercke, S. (2022a). Copulas, Lower Probabilities and Random Sets: How and When to Apply Them? In *Building Bridges between Soft and Statistical Methodologies for Data Science*, volume 1433, pages 271–278, Cham. Springer International Publishing. Series Title: Advances in Intelligent Systems and Computing.

[Malinowski and Destercke, 2022b] Malinowski, R. and Destercke, S. (2022b). Copules, probabilités inférieures et ensembles aléatoires : comment et quand les appliquer ? In *LFA 2022 - Rencontres francophones sur la Logique Floue et ses Applications*.

[Malinowski et al., 2023] Malinowski, R., Destercke, S., Dubois, E., Dumas, L., and Sarrazin, E. (2023). Uncertainty Propagation using Copulas in a 3D Stereo Matching Pipeline. In *Proceedings of Machine Learning Research*, volume 215, pages 288–298, Oviedo, Spain.

[Malinowski et al., 2024a] Malinowski, R., Destercke, S., Dumas, L., Dubois, E., and Sarrazin, E. (2024a). Uncertainty propagation in stereo matching using copulas. *International Journal of Approximate Reasoning*, 170:109191.

[Malinowski et al., 2024b] Malinowski, R., Sarrazin, E., Dubois, E., Dumas, L., and Destercke, S. (2024b). Robust confidence intervals for digital surface models using satellite photogrammetry. In *IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium*, pages 8741–8744.

[Malinowski et al., 2024c] Malinowski, R., Sarrazin, E., Dumas, L., Dubois, E., and Destercke, S. (2024c). Robust Confidence Intervals in Stereo Matching using Possibility Theory. arXiv:2404.06273 [cs].

[Marí et al., 2022] Marí, R., Ehret, T., and Facciolo, G. (2022). Disparity Estimation Networks for Aerial and High-Resolution Satellite Images: A Review. *Image Processing On Line*, 12:501–526.

[Marti et al., 2016] Marti, R., Gascoin, S., Berthier, E., de Pinel, M., Houet, T., and Laffly, D. (2016). Mapping snow depth in open alpine terrain from stereo satellite imagery. *The Cryosphere*, 10(4):1361–1380.

[Mehltretter and Heipke, 2019] Mehltretter, M. and Heipke, C. (2019). CNN-Based Cost Volume Analysis as Confidence Measure for Dense Matching. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 2070–2079, Seoul, Korea (South). IEEE.

[Melet et al., 2020] Melet, O., Youssefi, D., L'Helguen, C., Michel, J., Sarrazin, E., Languille, F., and Lebègue, L. (2020). CO3D Mission Digital Surface Model Production Pipeline. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B2-2020.

[Menze and Geiger, 2015] Menze, M. and Geiger, A. (2015). Object scene flow for autonomous vehicles. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3061–3070, Boston, MA, USA. IEEE.

[Mesa-Mingorance and Ariza-López, 2020] Mesa-Mingorance, J. L. and Ariza-López, F. J. (2020). Accuracy Assessment of Digital Elevation Models (DEMs): A Critical Review of Practices of the Past Three Decades. *Remote Sensing*, 12(16).

[Michel et al., 2020] Michel, J., Sarrazin, E., Youssefi, D., Cournet, M., Buffe, F., Delvit, J.-M., Emilien, A., Bosman, J., Melet, O., and L'Helguen, C. (2020). A New Satellite Imagery Stereo Pipeline Designed for Scalability, Robustness and Performance. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, V-2-2020:171–178.

[Miguez-Macho et al., 2007] Miguez-Macho, G., Fan, Y., Weaver, C. P., Walko, R., and Robock, A. (2007). Incorporating water table dynamics in climate modeling: 2. Formulation, validation, and soil moisture simulation. *Journal of Geophysical Research: Atmospheres*, 112(D13). _eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2006JD008112.

[Min-Gyu Park and Yoon, 2015] Min-Gyu Park and Yoon, K.-J. (2015). Leveraging stereo matching with learning-based confidence measures. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 101–109, Boston, MA, USA. IEEE.

[Monnet, 2023] Monnet, J.-M. (2023). *lidarHD: Utilities to download and manage files of France national airborne laser scanning cover (LIDAR HD)*. Institut National de la Recherche Agronomique.

[Montes et al., 2015] Montes, I., Miranda, E., Pelessoni, R., and Vicig, P. (2015). Sklar's theorem in an imprecise setting. *Fuzzy Sets and Systems*, 278:48–66.

[Morgan et al., 2004] Morgan, M., Kim, K., Jeong, S., and Habib, A. (2004). Epipolar Geometry of Linear Array Scanners Moving with Constant Velocity and Constant Attitude. In *XX ISPRS Congress*, volume XXXV.

[NASA, 2020] NASA (2020). Ames Stereo Pipeline documentation (version "3.4.0").

[Nelsen, 2006] Nelsen, R. B. (2006). *An introduction to copulas*. Springer series in statistics. Springer, New York Berlin Heidelberg, 2. ed edition.

[Noh and Howat, 2017] Noh, M.-J. and Howat, I. M. (2017). The Surface Extraction from TIN based Search-space Minimization (SETSM) algorithm. *ISPRS Journal of Photogrammetry and Remote Sensing*.

[Nuth and Kääb, 2011] Nuth, C. and Kääb, A. (2011). Co-registration and bias corrections of satellite elevation data sets for quantifying glacier thickness change. *The Cryosphere*, 5(1):271–290.

[Oberkampf et al., 2001] Oberkampf, W., Helton, J., and Sentz, K. (2001). Mathematical representation of uncertainty. In *19th AIAA Applied Aerodynamics Conference*. _eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2001-1645.

[Oh et al., 2010] Oh, J., Lee, W. H., Toth, C. K., Grejner-Brzezinska, D. A., and Lee, C. (2010). A Piecewise Approach to Epipolar Resampling of Pushbroom Satellite Images Based on RPC. *Photogrammetric Engineering & Remote Sensing*, 76(12):1353–1363.

[Oksanen, 2006] Oksanen, J. (2006). *Digital elevation model error in terrain analysis*. PhD thesis, University of Helsinki.

[Okutomi and Kanade, 1994] Okutomi, M. and Kanade, T. (1994). A stereo matching algorithm with an adaptive window: theory and experiment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):920–932.

[Panagiotakis et al., 2018] Panagiotakis, E., Chrysoulakis, N., Charalampopoulou, V., and Poursanidis, D. (2018). Validation of Pleiades Tri-Stereo DSM in Urban Areas. *ISPRS International Journal of Geo-Information*, 7(3):118.

[Park and Yoon, 2019] Park, M.-G. and Yoon, K.-J. (2019). Learning and Selecting Confidence Measures for Robust Stereo Matching. *IEEE transactions on pattern analysis and machine intelligence*, 41(6):1397–1411. Place: United States.

[Patelli et al., 2014] Patelli, E., Alvarez, D. A., Broggi, M., and de Angelis, M. (2014). An integrated and efficient numerical framework for uncertainty quantification: application to the NASA Langley multidisciplinary Uncertainty Quantification Challenge. In *16th AIAA Non-Deterministic Approaches Conference*, National Harbor, Maryland. American Institute of Aeronautics and Astronautics.

[Pekel et al., 2016] Pekel, J.-F., Cottam, A., Gorelick, N., and Belward, A. S. (2016). High-resolution mapping of global surface water and its long-term changes. *Nature*, 540(7633):418–422.

[Pelessoni et al., 2016] Pelessoni, R., Vicig, P., Montes, I., and Miranda, E. (2016). Bivariate p -boxes. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 24(02):229–263.

[Poggi et al., 2021] Poggi, M., Kim, S., Tosi, F., Kim, S., Aleotti, F., Min, D., Sohn, K., and Mattoccia, S. (2021). On the confidence of stereo matching in a deep-learning era: a quantitative evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1.

[Poggi and Mattoccia, 2016] Poggi, M. and Mattoccia, S. (2016). Learning a General-Purpose Confidence Measure Based on O(1) Features and a Smarter Aggregation Strategy for Semi Global Matching. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 509–518, Stanford, CA, USA. IEEE.

[Poggi et al., 2017] Poggi, M., Tosi, F., and Mattoccia, S. (2017). Quantitative Evaluation of Confidence Measures in a Machine Learning World. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5238–5247.

[Poli et al., 2015] Poli, D., Remondino, F., Angiuli, E., and Agugiaro, G. (2015). Radiometric and geometric evaluation of GeoEye-1, WorldView-2 and Pléiades-1A stereo images for 3D information extraction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 100:35–47.

[Qin, 2016] Qin, R. (2016). RPC Stereo Processor (RSP) – A Software Package for Digital Surface Model and Orthophoto Generation from Satellite Stereo Imagery. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, III.

[Qin, 2019] Qin, R. (2019). A critical analysis of satellite stereo pairs for digital surface model generation and a matching quality prediction model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 154:139–150.

[Qin et al., 2021] Qin, R., Gruen, A., and Fraser, C. (2021). Quality assessment of image matchers for DSM generation – a comparative study based on UAV images. _eprint: 2108.08369.

[Qin et al., 2022] Qin, R., Ling, X., Farella, E. M., and Remondino, F. (2022). Uncertainty-Guided Depth Fusion from Multi-View Satellite Images to Improve the Accuracy in Large-Scale DSM Generation. *Remote Sensing*, 14(6):1309.

[Qingxiong Yang, 2012] Qingxiong Yang (2012). A non-local cost aggregation method for stereo matching. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1402–1409, Providence, RI. IEEE.

[Quinio, 1992] Quinio, P. (1992). Random set approach to 3D scene reconstruction by stereoscopic vision. In Merritt, J. O. and Fisher, S. S., editors, *Stereoscopic Displays and Applications III*, pages 61–70, San Jose, CA.

[Quinio and Matsuyama, 1991] Quinio, P. and Matsuyama, T. (1991). Random closed sets: A unified approach to the representation of imprecision and uncertainty. In *Symbolic and Quantitative Approaches to Uncertainty*, volume 548, pages 282–286, Berlin, Heidelberg. Springer Berlin Heidelberg. Series Title: Lecture Notes in Computer Science.

[Richardson, 2006] Richardson, I. E. G. (2006). *H.264 and MPEG-4 Video compression: video coding for next-generation multimedia*. Wiley, Chichester, reprint edition.

[Rieg et al., 2018] Rieg, L., Klug, C., Nicholson, L., and Sailer, R. (2018). Pléiades Tri-Stereo Data for Glacier Investigations—Examples from the European Alps and the Khumbu Himal. *Remote Sensing*, 10(10):1563.

[Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *CoRR*, abs/1505.04597. arXiv: 1505.04597.

[Rupnik et al., 2017] Rupnik, E., Daakir, M., and Pierrot Deseilligny, M. (2017). MicMac – a free, open-source solution for photogrammetry. *Open Geospatial Data, Software and Standards*, 2(1):14.

[Sadeghi et al., 2016] Sadeghi, Y., St-Onge, B., Leblon, B., and Simard, M. (2016). Canopy Height Model (CHM) Derived From a TanDEM-X InSAR DSM and an Airborne Lidar DTM in Boreal Forest. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(1):381–397.

[Sarrazin et al., 2021] Sarrazin, E., Cournet, M., Dumas, L., Defonte, V., Fardet, Q., Steux, Y., Jimenez Diaz, N., Dubois, E., Youssefi, D., and Buffe, F. (2021). Ambiguity

Concept In Stereo Matching Pipeline. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B2-2021:383–390.

[Scharstein et al., 2014] Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nešić, N., Wang, X., and Westling, P. (2014). High-Resolution Stereo Datasets with Subpixel-Accurate Ground Truth. In Hornegger, J., Jiang, X., and Koch, R., editors, *Pattern Recognition*, volume 8753, pages 31–42. Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.

[Scharstein and Pal, 2007] Scharstein, D. and Pal, C. (2007). Learning Conditional Random Fields for Stereo. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Minneapolis, MN, USA. IEEE.

[Scharstein and Szeliski, 2003] Scharstein, D. and Szeliski, R. (2003). High-accuracy stereo depth maps using structured light. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, pages I–195–I–202, Madison, WI, USA. IEEE Comput. Soc.

[Scharstein et al., 2001] Scharstein, D., Szeliski, R., and Zabih, R. (2001). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In *Proceedings IEEE Workshop on Stereo and Multi-Baseline Vision (SMBV 2001)*, pages 131–140, Kauai, HI, USA. IEEE Comput. Soc.

[Schmelzer, 2012] Schmelzer, B. (2012). Characterizing joint distributions of random sets by multivariate capacities. *International Journal of Approximate Reasoning*, 53(8):1228–1247.

[Schmelzer, 2015a] Schmelzer, B. (2015a). Joint distributions of random sets and their relation to copulas. *International Journal of Approximate Reasoning*, 65:59–69.

[Schmelzer, 2015b] Schmelzer, B. (2015b). Sklar's theorem for minitive belief functions. *International Journal of Approximate Reasoning*, 63:48–61.

[Schmelzer, 2019] Schmelzer, B. (2019). Multivariate capacity functionals vs. capacity functionals on product spaces. *Fuzzy Sets and Systems*, 364:1–35.

[Schmelzer, 2023] Schmelzer, B. (2023). Random sets, copulas and related sets of probability measures. *International Journal of Approximate Reasoning*, 160.

[Shafer, 1976] Shafer, G. (1976). *A Mathematical Theory of Evidence*. Princeton University Press.

[Shean et al., 2016] Shean, D. E., Alexandrov, O., Moratto, Z. M., Smith, B. E., Joughin, I. R., Porter, C., and Morin, P. (2016). An automated, open-source pipeline for mass production of digital elevation models (DEMs) from very-high-resolution commercial stereo satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 116:101–117.

[Sklar, 1959] Sklar, M. (1959). *Fonctions de Répartition À N Dimensions Et Leurs Marges*. Université Paris 8.

[Sun et al., 2003] Sun, J., Zheng, N.-N., and Shum, H.-Y. (2003). Stereo Matching Us-

ing Belief Propagation. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 25(7):14.

[Tao and Hu, 2001] Tao, C. V. and Hu, Y. (2001). A Comprehensive Study of the Rational Function Model for Photogrammetric Processing. *PHOTOGRAMMETRIC ENGINEERING*.

[Tomasi and Manduchi, 1998] Tomasi, C. and Manduchi, R. (1998). Bilateral filtering for gray and color images. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 839–846, Bombay, India. Narosa Publishing House.

[Tombari et al., 2007] Tombari, F., Mattoccia, S., and Di Stefano, L. (2007). Segmentation-Based Adaptive Support for Accurate Stereo Correspondence. In Hutchison, D., Kanade, T., Kittler, J., Kleinberg, J. M., Mattern, F., Mitchell, J. C., Naor, M., Nierstrasz, O., Pandu Rangan, C., Steffen, B., Sudan, M., Terzopoulos, D., Tygar, D., Vardi, M. Y., Weikum, G., Mery, D., and Rueda, L., editors, *Advances in Image and Video Technology*, volume 4872, pages 427–438. Springer Berlin Heidelberg, Berlin, Heidelberg. Series Title: Lecture Notes in Computer Science.

[Tosi et al., 2024] Tosi, F., Bartolomei, L., and Poggi, M. (2024). A Survey on Deep Stereo Matching in the Twenties. arXiv:2407.07816 [cs].

[Tosi et al., 2018] Tosi, F., Poggi, M., Benincasa, A., and Mattoccia, S. (2018). Beyond local reasoning for stereo confidence estimation with deep learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

[Touloumis, 2015] Touloumis, A. (2015). Nonparametric Stein-type shrinkage covariance matrix estimators in high-dimensional settings. *Computational Statistics & Data Analysis*, 83:251–261. Publisher: Elsevier BV.

[Toutin et al., 2012] Toutin, T., Wang, H., Chomaz, P., and Pettier, E. (2012). Orthorectification of polarimetric Radarsat-2/RCM data with accurate lidar DSM. In *2012 IEEE International Geoscience and Remote Sensing Symposium*, pages 5880–5883.

[Troffaes, 2017] Troffaes, M. C. M. (2017). A Note on Imprecise Monte Carlo over Credal Sets via Importance Sampling. In Antonucci, A., Corani, G., Couso, I., and Destercke, S., editors, *Proceedings of the Tenth International Symposium on Imprecise Probability: Theories and Applications*, volume 62 of *Proceedings of Machine Learning Research*, pages 325–332. PMLR.

[Velazco, 2012] Velazco, E. (2012). 3D Digital Modeling for Urban Design + Planning. *City and Regional Planning*.

[Walley, 1991] Walley, P. (1991). *Statistical Reasoning with Imprecise Probabilities*. Springer US, Boston, MA.

[Wang et al., 2015] Wang, B., Shi, W., and Liu, E. (2015). Robust methods for assessing the accuracy of linear interpolated DEM. *International Journal of Applied Earth Observation and Geoinformation*, 34:198–206.

[Wang et al., 2022] Wang, C., Wang, X., Zhang, J., Zhang, L., Bai, X., Ning, X., Zhou, J., and Hancock, E. (2022). Uncertainty estimation for stereo matching based on evidential

deep learning. *Pattern Recognition*, 124:108498.

[Warth et al., 2019] Warth, G., Braun, A., Bödinger, A., Hochschild, V., and Bachofer, F. (2019). DSM-based identification of changes in highly dynamic urban agglomerations. *European Journal of Remote Sensing*, 52(1):322–334. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/22797254.2019.1604083.

[Wessel, 2018] Wessel, B. (2018). TanDEM-X Ground Segment – DEM Products Specification Document.

[Williamson, 2009] Williamson, J. (2009). PHILOSOPHIES OF PROBABILITY. In Irvine, A. D., editor, *Philosophy of Mathematics*, Handbook of the Philosophy of Science, pages 493–533. North-Holland, Amsterdam. ISSN: 18789846.

[Yamazaki et al., 2019] Yamazaki, D., Ikeshima, D., Sosa, J., Bates, P. D., Allen, G. H., and Pavelsky, T. M. (2019). MERIT Hydro: A High-Resolution Global Hydrography Map Based on Latest Topography Dataset. *Water Resources Research*, 55(6):5053–5073.

[Yamazaki et al., 2014] Yamazaki, D., Sato, T., Kanae, S., Hirabayashi, Y., and Bates, P. D. (2014). Regional flood dynamics in a bifurcating mega delta simulated in a global river model. *Geophysical Research Letters*, 41(9):3127–3135. _eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1002/2014GL059744.

[Yan et al., 2018] Yan, L., Roy, D. P., Li, Z., Zhang, H. K., and Huang, H. (2018). Sentinel-2A multi-temporal misregistration characterization and an orbit-based subpixel registration methodology. *Remote Sensing of Environment*, 215:495–506.

[Youssefi et al., 2024] Youssefi, D., Derksen, D., Migel-Arachchige, D., Siefert, J., Dumas, L., and Guinet, J. (2024). Geometrically guided and confidence-based point cloud denoising. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XLVIII-4/W12-2024, pages 149–155, Tartu, Estonia.

[Zabih and Woodfill, 1994] Zabih, R. and Woodfill, J. (1994). Non-parametric local transforms for computing visual correspondence. In Goos, G., Hartmanis, J., and Eklundh, J.-O., editors, *Computer Vision — ECCV '94*, volume 801, pages 151–158. Springer Berlin Heidelberg, Berlin, Heidelberg. Series Title: Lecture Notes in Computer Science.

[Zadeh, 1999] Zadeh, L. (1999). Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 100:9–34.

[Žbontar and LeCun, 2016] Žbontar, J. and LeCun, Y. (2016). Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. *Journal of Machine Learning Research 17*. arXiv: 1510.05970.

[Zureiki et al., 2008] Zureiki, A., Devy, M., and Chatil, R. (2008). Stereo Matching and Graph Cuts. In Bhatti, A., editor, *Stereo Vision*. InTech.