UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

THIAGO LOPES TRUGILLO DA SILVEIRA

Dense 3D Indoor Scene Reconstruction from Spherical Images

Thesis presented in partial fulfillment
of the requirements for the degree of
Doctor of Computer Science

Advisor: Prof. Dr. Cláudio Rosito Jung

Porto Alegre
September 2019

CIP — CATALOGING-IN-PUBLICATION

da Silveira, Thiago Lopes Trugillo

Dense 3D Indoor Scene Reconstruction from Spherical Images / Thiago Lopes Trugillo da Silveira. – Porto Alegre: PPGC da UFRGS, 2019.

133 f.: il.

Thesis (Ph.D.) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2019. Advisor: Cláudio Rosito Jung.

1. Spherical images. 2. 3D scene reconstruction. 3. Structure from motion. I. Jung, Cláudio Rosito. II. Título.

“An expert is a man who has made all the mistakes which can be made, in a narrow field.”

— NIELS BOHR

ACKNOWLEDGMENTS

My warmest thank to all those who contributed to this work. Primarily, I would like to thank my parents, Ilse Trindade Lopes and André Trugillo da Silveira, for their support during all my academic formation since 2010. I have also to thank my fiancé, Samara Dias Osorio, for being there always for me.

Also, I would like to thank all my professors and colleagues at the Universidade Federal do Rio Grande do Sul. Especially, I would like to acknowledge the insightful discussions that Dr. Cláudio Jung and I have during these almost four years. I sincerely thank Dr. Altamiro Susin, Dr. Anderson Maciel, Dr. Eduardo Gastal, and Dr. Leonardo Sacht for their contributions to this Dissertation.

Finally, I would like to thank Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) and Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) for the scholarships provided during my Doctorate course.

ABSTRACT

Image-based three-dimensional (3D) scene reconstruction approaches have been widely studied by the scientific community, with applications in archaeological and architectural modeling, infrastructure inspection, robot navigation, and autonomous driving systems, just to name a few. The vast majority of existing approaches deal with traditional pinhole cameras, which present a narrow field of view (FoV) and hence require several captures to model larger scenes. On the other hand, omnidirectional cameras present full 360 degrees FoV and have become popular in the past years with the release of consumer-grade devices. In this Dissertation, we address the problem of 3D indoor scene reconstruction based on multiple uncalibrated and unordered spherical images. In the initial part of this work, we show that the gold standard method for solving the relative five degrees of freedom (5-DoF) camera pose in the classic two-view problem, namely the eight-point algorithm (8-PA), is capable of producing more accurate estimates when using wide FoV image pairs compared to typical perspective/pinhole-based cameras. These results guide our main pipeline, allowing us to skip traditional but expensive approaches for non-linear refinement of both camera poses and depth estimates from two or more spherical input images. More precisely, our method uses sparse keypoint matching for initially derotating supporting images with respect to a preset reference image. Then, we use a large displacement optical flow algorithm for obtaining dense correspondences between the reference image and the others, and use them to estimate a confidence map that guides the depth estimation process. The proposed multi-view methodology generates a dense depth map fully registered to the reference color image, which allows us to enhance the estimated 3D structure by using image-guided filtering approaches. This representation also allows us to explore depth-image-based rendering techniques for generating novel views of the scene, with applications in 3-DoF+ navigation for augmented/mixed/virtual reality. We further investigate how to improve the recovered 3D geometry of the scenes by aggregating information in perceptually meaningful regions of the reference image. For this purpose, we adapt a superpixel algorithm to the spherical domain and use its output for guiding a spatially-constrained version of our calibrated reconstruction method. We also propose to use the segmented regions to select good scattered correspondences from the dense set of matchings for estimating the 6-DoF camera poses of the supporting images, going in the same direction as in the 8-PA analysis. As an additional contribution, we introduce a framework for inferring depth from a single spherical image, which can

be coupled to any existing and future monocular depth estimation algorithm suited for perspective images. We validate our approaches using both synthetic data and computer-generated imagery for which we have access to ground truth for pose and depth, showing competitive results concerning state-of-the-art methods.

Keywords: Spherical images. 3D scene reconstruction. structure from motion.

Reconstrução 3D Densa de Cenas Internas Através de Imagens Esféricas

RESUMO

Abordagens para reconstrução tridimensional (3D) de cenas baseadas em imagens têm sido amplamente estudadas pela comunidade científica, tendo aplicações em modelagem arqueológica e arquitetural, inspeção de infraestruturas, navegação de robôs e sistemas de nagevação autônomos, apenas para citar algumas. A vasta maioria das abordagens existentes lidam com as tradicionais câmeras *pinhole*, que apresentam um estreito campo de visão (FoV) e portanto requerem diversas capturas para modelar grandes cenas. Por outro lado, câmeras omnidirecionais apresentam um FoV de 360 graus, e têm se tornado populares em dispositivos de consumo nos últimos anos. Nesta Tese, nós abordamos o problema de reconstrução 3D de cenas internas baseada em múltiplas imagens esféricas não calibradas e não ordenadas. Na parte inicial deste trabalho, nós mostramos que o algoritmo padrão para resolver o problema dos cinco graus de liberdade (5-DoF) da pose relativa entre câmeras na configuração clássica de duas vistas, a saber o *eight-point algorithm* (8-PA), é capaz de produzir estimativas mais acuradas quando usando pares de imagens com amplo FoV se comparadas com as típicas câmeras *pinhole*/perspectiva. Esses resultados guiam nossa linha de trabalho principal, permitindo-nos evitar clássicas mas custosas abordagens para refinamento não-linear de ambas as estimativas de pose e profundidade a partir de duas ou mais imagens de entrada. Mais precisamente, nosso método usa casamento esparso de pontos-chave para inicialmente desrotacionar as imagens de suporte em relação a uma imagem de referência previamente selecionada. Então, nós usamos um algoritmo de fluxo ótico com suporte a grandes deslocamentos para obter um conjunto denso de correspondências entre a imagem de referência e as outras, e usamos esse conjunto para estimar um mapa de confiança que guia o processo de estimativa de profundidade. A metodologia proposta baseada em múltiplas vistas gera um mapa de profundidade denso completamente registrado à imagem colorida de referência, o que permite a aplicação de métodos para melhoramento da estrutura 3D estimada usando técnicas de filtragem guiadas por imagem. Essa representação também permite explorar técnicas de renderização baseada em imagens de cor e profundidade para gerar novas vistas da cena, tendo aplicações em navegação 3-DoF+ para realidade aumentada/mista/virtual. Nós ainda investigamos como melhorar a recuperação da geometria 3D de cenas através da agregação de informações em regiões perceptualmente coerentes da imagem de refe-

rência. Para esse propósito, nós adaptamos um algoritmo de extração de *superpixels* para o domínio esférico e usamos seu resultado para guiar uma versão com restrição espacial de nosso método de reconstrução calibrada. Nós também propomos usar as regiões de supersegmentação para selecionar boas e bem distribuídas correspondências do conjunto denso de casamentos de pontos para estimar a pose 6-DoF das câmeras de suporte, indo na mesma direção da análise com o 8-PA. Como uma contribuição adicional, nós introduzimos um arcabouço para inferir profundidade a partir de uma única imagem esférica, que pode ser acoplado a qualquer algoritmo, existente ou futuro, para inferência de profundidade de imagens *pinhole*. Nós validamos nossas abordagens usando ambos dados sintéticos e imagens geradas por computador para os quais é possível ter acesso aos parâmetros reais para pose e profundidade, mostrando resultados competitivos se comparados a métodos no estado da arte.

Palavras-chave: Imagens esféricas. Reconstrução de cenas 3D. Estrutura através do movimento.

LIST OF ABBREVIATIONS AND ACRONYMS

3DTV	3D television
d-D	d-dimensional
8-PA	Eight-point algorithm
A-KAZE	Accelerated KAZE
ASIFT	Affine SIFT
AR	Augmented reality
BA	Bundle adjustment
BnB	Branch-and-bound
BRIEF	Binary robust independent elementary features
BRISK	Binary robust invariant scalable keypoints
BRISKS	Spherical BRISK
CNN	Convolutional neural network
CRF	Conditional random fields
DFoV	Diagonal FoV
DIBR	Depth-image-based rendering
DoF	Degrees of freedom
DoG	Difference of Gaussians
DLT	Direct linear transformation
DT	Domain transform
EGS	Efficient graph-based segmentation
FAST	Features from accelerated segment test
FoV	Field of view
FPS	Frames per second
FVV	Free-viewpoint

GPS	Global positioning system
HFoV	Horizontal FoV
HHF	Hierarchical hole filling
HMD	Head mounted display
ICP	Iterative closest point
IRLS	Iterative reweighted least-squares
KLT	Kanade-Lucas-Tomasi
LIDAR	Light detection and ranging
M-LDB	Modified-local difference binary
MPEG	Motion Picture Experts Group
MR	Mixed reality
MVS	Multi-view stereo
NLR	Non-linear refinement
ORB	Oriented FAST and rotated BRIEF
PanoEGS	Panoramic EGS
PDE	Partial differential equation
PMVS	Patch-based Multiple View Stereo
PnP	Perspective-n-Point
RANSAC	Random sample consensus
RoI	Region of interest
SIFT	Scale-invariant feature transform
SfM	Structure from motion
SLF	Spherical light field
SLIC	Simple linear iterative clustering
SM	Stereo matching
SNIC	Simple non-iterative clustering

SnP	Spherical-n-Point
SPHORB	Spherical ORB
SS	Sphere sweeping
SSIFT	Spherical SIFT
SSLIC	Spherical SLIC
SSNIC	Spherical SNIC
SVD	Singular value decomposition
ToF	Time-of-flight
V-SLAM	Visual simultaneous localization and mapping
VFoV	Vertical FoV
vMF	von-Mises Fisher
VR	Virtual reality

LIST OF SYMBOLS

$\mathbf{X}^i$	i-th world point
$\mathbf{C}_j$	j-th camera center
$\mathbf{x}_j^i$	i-th projection in the j-th camera
S^2	2-sphere space
$\mathbf{R}_j$	j-th rotation matrix
$\mathbf{t}_j$	j-th translation vector
$SO(3)$	Group of all rotations in $\mathbb{R}^3$
$\|\cdot\|_2$	L-2 norm
$\mathbf{I}$	Identity matrix
$(r=1,\phi,\theta)$	Spherical coordinates
(x,y)	Image coordinates
ψ	Mapping of spherical to image coordinates
ψ^{-1}	Mapping of image to spherical coordinates
$\mathbf{E}$	Essential matrix
$\mathbf{F}$	Fundamental matrix
$[\,\cdot\,]_\times$	Skew-symmetric matrix
$\mathbf{A}$	Matrix that encodes the matchings to be solved via the 8-PA
n	Number of matched points / 3D world points
$\mathbf{e}$	Essential matrix in vector form
$\mathbf{U}_{(\cdot)}\mathbf{\Sigma}_{(\cdot)}\mathbf{V}_{(\cdot)}^\top$	SVD of the argument matrix
$\tilde{(\cdot)}$	Approximation of the argument matrix/vector/scalar
$\mathrm{diag}(\sigma_1,\ldots)$	Singular vectors
r	Rank of a matrix
$\mathbf{M}$	Approximately rank-r matrix

$\mathbf{P}$	Perturbation matrix
$\Theta(\cdot,\cdot)$	Canonical angles between argument matrices
δ	Gap
$\mathbb{P}_{(\cdot)}$	Projection operator onto the column space of the argument matrix
$\|\cdot\|_F$	Frobenius norm
η	Angle between approximate and actual Essential matrices in vector form
$\tilde{\lambda}_i$	i-th eigenvalue of $\tilde{\mathbf{A}}^\top\tilde{\mathbf{A}}$
β_{ij}	Angle between the i-th and j-th projections in the first camera
γ_{ij}	Angle between the i-th and j-th projections in the second camera
$\mathbf{W}$	Diagonal matrix of weights
ω	Angle between approximate and actual translation vectors
κ	Concentration parameter in the von Mises-Fisher distribution
$\varepsilon_{\angle\mathbf{t}}$	Angular translation error
$\varepsilon_{\angle\mathbf{R}}$	Angular rotation error
$G_e(\cdot,\cdot)$	Symmetric projected distance between argument features
λ_S	Threshold for the considering a valid match
λ_P	Minimal number of inliers for RANSAC
$P_e(\cdot,\cdot)$	Cross-checking error between argument features
$\mathfrak{I}_k$	Input image k
$\mathbf{u}^f$	Forward optical flow
$\mathbf{u}^b$	Backward optical flow
$J_c(\cdot,\cdot)$	Confidence score between argument features
γ_1	Parameter for controlling the influence of $P_e(\cdot,\cdot)$ in $J_c(\cdot,\cdot)$
γ_2	Parameter for controlling the influence of $G_e(\cdot,\cdot)$ in $J_c(\cdot,\cdot)$
ϑ_i	Depth associated to the i-th projection
w_j	Weight value for the j-th camera

E_{3D}	Weighted Euclidean error
$\|d\mathbf{x}_j\|$	Disturbance in the matchings relative to the j-th camera
T_d	Threshold value for depth
Q_q	q-th quantile of a distribution
σ_s	DT filter space isotropic kernel
σ_s^v	DT filter space anisotropic vertical kernel
σ_s^h	DT filter space anisotropic horizontal kernel
σ_r	DT filter range (color) kernel
J	Number of cameras
$\varepsilon_{(\cdot)}$	Relative error for the argument matrices/vectors
k	Number of SPHORB keypoints
λ_M	Threshold for the ration matching strategy
f	Fraction of dense matchings
$\mathcal{N}(0, \sigma_t^2)$	Normal distribution with zero mean and standard deviation σ_t
R	Number of superpixels
Q	Number of pixels within a superpixel
$dS(\cdot)$	Cosine dissimilarity distance
$dC(\cdot)$	CIE L*a*b* color distance
$dT(\cdot)$	Combined spatial and color distance
$\mathbf{c}$	Color vector in CIE L*a*b* space
s	Normalizing factor for cosine dissimilarity
m	Compactness parameter for SSNIC
N_s	Number of angular sections
(θ_C, ϕ_C)	Sphere projection center
$\mathbf{S}_k$	k-th angular section
$\mathbf{D}_k$	Depth estimate associated to the k-th angular section

$\mathcal{P}_k$	Set containing the pixels in the overlapping region of section k
$\mathcal{R}_k$	Set containing the rows of the pixels in the overlapping region of section k
$\mathbf{w}_L^k$	Row-wise weights for the depth estimate in the left overlapping region of section k
$\mathbf{w}_R^k$	Row-wise weights for the depth estimate in the right overlapping region of section k
C_D^k	Data term for minimizing the error in section k
C_R^k	Regularization term for minimizing the error in section k
$\chi(i,j)$	Weight dependent on the similarity of colors
τ	Threshold for color
C_T^k	Total cost function for minimizing the error in section k
j_L^i	Lowest column value in row i
j_R^i	Highest column value in row i
$\mathbf{\Gamma}$	Final single image-based depth map

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

1.1 Motivation

Image-based three-dimensional (3D) scene reconstruction approaches have been widely studied by the scientific community because the involved capturing devices are cheaper and simpler to use than other technologies such as light detection and ranging (LIDAR), time-of-flight (ToF) and structured light (LUKIERSKI; LEUTENEGGER; DAVISON, 2015). 3D models obtained from pinhole-based/perspective images are extensively employed in applications like archaeological (PAGANI et al., 2011) and architectural modeling (KIM; HILTON, 2015), infrastructure inspection (PATHAK et al., 2016b), robot navigation (MOREAU; AMBELLOUIS; RUICHE, 2012) and autonomous driving systems (TONG; NING, 2013), just to name a few. Moreover, due to the recent release of easy-to-use consumer-grade devices for acquisition and visualization of *omnidirectional* media, novel augmented, mixed and virtual reality (AR/MR/VR)-based applications are emerging. 3D layout modeling of indoor scenes (ZOU et al., 2018) and 3D semantic labeling (SONG et al., 2017) are examples of two applications that can benefit from this media type. Furthermore, in this promising context, extracting 3D information is fundamental for implementing (or approximating) the six-degrees of freedom (6-DoF) (ANDERSON et al., 2016; THATTE et al., 2017; HUANG et al., 2017) needed to ensure the AR/MR/VR users fully immersive experiences.

In order to recover 3D information from traditional planar images, one can highlight three intermediate tasks: (i) the calibration of the intrinsic camera(s) parameters; (ii) the estimation of the (calibrated) camera poses (positioning and orientation in world coordinates) with respect to some reference; and (iii) the selection of a set of features to be matched across the captured images (HARTLEY; ZISSERMAN, 2003). Intrinsic camera calibration is classically treated as a pre-processing step (MOLNAR et al., 2015; XU; MULLIGAN, 2008), and hereafter, unless otherwise mentioned, we will use “calibration” for referring to *extrinsic* calibration. Therefore, the number of input images, the (possible) prior knowledge about the extrinsic camera settings, and the density of the computed correspondences are the most relevant aspects to distinguish methods and applications.

Classic stereo matching (SM) techniques (SCHARSTEIN; SZELISKI; ZABIH, 2001) try to (globally or locally) minimize some cost function to match all the features from a reference image to their correspondences in another, often producing a *dense* dis-

parity or depth map of the scene from two camera views. Although the pioneer studies about SM date from the 1970s, there is a still growing collection of works trying to solve a variety of challenging problems involving textureless regions (YANG, 2015), translucent materials (GODARD; Mac Aodha; BROSTOW, 2017) and thin structures (ZHOU et al., 2017). However, most methods assume that the input images are rectified (KO et al., 2017), meaning that the relative camera pose is in a single-axis translational state with a known baseline. This premise leads to the simplification of the problem, drastically reduces the computational cost of the algorithms, but also restricts their applicability in real-world scenarios.

Because of the narrow field of view (FoV) of standard perspective cameras, only a small portion of the 3D scene can be recovered regardless of the SM technique employed. Multi-view stereo (MVS) algorithms (SEITZ et al., 2006) attempt to solve this problem by extending the SM concepts and considering (possibly much) more than two camera captures at different positions. Once again, the MVS approach is not a novelty but, as in the stereo case, there is still many recent methods trying to solve several intractable issues until now, like extreme scale and density diversity (MOSTEGEL et al., 2017) and non-rigid registration (PALMA et al., 2018). As expected, the algorithms become computationally expensive as the number of cameras increase, and thus most of them also rely on known camera positioning (THATTE et al., 2017), bypassing the pose estimation step.

In contrast to SM and MVS methods, structure from motion (SfM) (OZYESIL et al., 2017) and visual simultaneous localization and mapping (V-SLAM) (FUENTES-PACHECO; RUIZ-ASCENCIO; RENDÓN-MANCHA, 2012) techniques perform both the camera pose estimation and 3D scene reconstruction tasks together, making use of information from several pairwise-overlapping images. In general terms, those methods incrementally construct (and refine) the 3D scene as new camera views are added, relying on temporal coherence between adjacent captures or image clustering pre-processing to ensure the existence of feature matchings (SCHONBERGER; FRAHM, 2016). Unlike SM and MVS, SfM/V-SLAM techniques traditionally consider *sparse* features extracted by consecrated general-purpose keypoint detection/matching algorithms like the scale-invariant feature transform (SIFT) (LOWE, 2004), binary robust invariant scalable keypoints (BRISK) (LEUTENEGGER; CHLI; SIEGWART, 2011), oriented FAST and rotated BRIEF (ORB) (RUBLEE et al., 2011), and KAZE (ALCANTARILLA; BARTOLI; DAVISON, 2012).

Traditionally, SM and MVS techniques can generate dense depth maps but with

calibrated cameras. On the other hand, SfM/V-SLAM methods typically deal with uncalibrated image sequences but in a sparse level. Things can change significantly by considering truly omnidirectional (or 360° or spherical) cameras for capturing the whole scene instead of the narrow FoV pinhole-based ones. By nature, spherical images taken from the same scene should present a huge number of correspondences (exceptions are occlusions), since they store the whole $360^\circ \times 180^\circ$ environment information, *i.e.,* they present full FoV. Therefore, in theory, a dense depth map from the entire scene could be generated by considering only two view captures, as occurs in the SM scenario.

However, as a huge drawback, omnidirectional images contain intrinsic distortions resulting from the camera model (GUAN; SMITH, 2017b; SU; GRAUMAN, 2017; PATHAK et al., 2018), which prohibit them to be treated as pinhole-based, although their standard format representation (equirectangular projection of the sphere) is planar (PATHAK et al., 2016b). It means that most of the algorithms developed so far by the scientific community may not be capable of performing the tasks they were designed for when applied to spherical images. As an example, Figure 1.1 presents two views of a synthetically generated scene (*Classroom* scenario[1]) in a canonical pose and after rotating and translating a virtual camera. Perspective and omnidirectional captures (in the equirectangular format) are shown side by side, aiming to illustrate the deformations mentioned above (confer structures like the mural and the pipes on the ceiling), which depend on the objects positioning on the image (CRUZ-MOTA et al., 2012).

Despite the challenges, there is an increasing number of content and applications being generated, inciting both the industry and the scientific community (KOPF, 2016; HUANG; TSENG, 2016). Recent studies are focusing on estimating the camera pose in image sequences, and applying it on omnidirectional video stabilization (GUAN; SMITH, 2017b; PATHAK et al., 2017a), holder-free omnidirectional video (XU et al., 2016; XU et al., 2017) and view synthesis (GUAN, 2017; HUANG et al., 2017). However, only few studies have addressed the *dense* 3D scene reconstruction problem using omnidirectional cameras in the last years (PAGANI et al., 2011; PATHAK et al., 2016b; PATHAK et al., 2017a; KIM; HILTON, 2013; KIM; HILTON, 2015; IM et al., 2016; GAVA; STRICKER; YOKOTA, 2018; PATHAK et al., 2018; WEGNER et al., 2018; LAI et al., 2019; WON; RYU; LIM, 2019).

[1]The *Classroom* scene, rendered with Blender, is available under CC0 license in <https://www.blender.org/download/demo-files/>.

Figure 1.1: Perspective and omnidirectional images. Captures after (c),(d) rotating and translating the virtual camera *w.r.t.* (a),(b) a canonical view. Source: the author.

(a) Reference perspective capture

(b) Reference omnidirectional capture

(c) Perspective capture after moving the camera

(d) Omnidirectional capture after moving the camera

1.2 Goals and Contributions

Our main goal is to build a method that is capable of estimating a dense 3D representation of indoor scenes starting from two or more non-calibrated (and and possibly temporally unordered) spherical images. Note that one can estimate a fully dense depth map only from indoor scenarios, and thus we focus on these cases. For achieving this goal we need to investigate or propose solutions for (i) the (sparse or dense) correspondences matching; (ii) two/multi-view pose estimation, and (iii) two/multi-view depth estimation. As an additional contribution, we provide a framework for inferring depth information from a single spherical image that can be coupled to any existing or future technique for inferring depth from a single perspective image.

Our main contributions are explained throughout Chapters 3, 4, 5 and 6. Section 7.3 summarizes the published outcomes of this Dissertation.

1.3 Hypotheses

The main hypotheses of this Dissertation are the following:

(i) Using features matched over the sphere can yield better pose estimation results than using narrow-FoV localized features;

(ii) It is possible to obtain progressively more accurate 3D scene reconstructions by adding more spherical images;

(iii) With a fixed number of views, a proper positioning of the cameras can improve 3D scene reconstruction;

(iv) Standard optical flow algorithms can be used to obtain dense features, provided the camera views are close enough, for estimating both the 3D camera pose and dense geometry;

(v) It is possible to weight each one of the matched features so that a better 3D scene reconstruction is obtained than if all of them contribute the same;

(vi) A domain-adapted superpixel algorithm can be used to improve the average 6-DoF camera poses and depth estimates in a multi-view 3D reconstruction pipeline.

1.4 Chapters Organization

The remainder of this Dissertation is organized as follows. Chapter 2 revises related works. Basic concepts about spherical imaging, which may be insightful for the reader, are given in Section 2.1. The epipolar geometry for the spherical camera model, which is the base for the proposed multi-view method (and also related papers), is described Section 2.2 and existing methods for measuring the sensitivity to perturbations on the gold standard algorithm for this task are exposed in Section 2.2.3. A discussion about choosing sparse or dense feature matching algorithms when considering spherical images is provided in Section 2.3. Last but not least, a literature review about the camera pose estimation and the 3D scene reconstruction tasks is presented in Section 2.4.

Chapter 3 presents a novel perturbation analysis for the estimate of Epipolar (Fundamental or Essential) matrices using the eight-point algorithm (8-PA) (LONGUET-HIGGINS, 1987). It explores existing bounds for singular subspaces and relates them to the 8-PA,

without assuming any error distribution for the matched features. It also shows that having a wide spatial distribution of matched features in both views tends to generate lower error bounds for the Epipolar matrix error, which can be achieved by using spherical images.

Chapter 4 introduces the proposed method for estimating the 3D geometry of indoor scenes based on multiple spherical images. It produces a dense depth map registered to a reference view so that depth-image-based rendering (DIBR) techniques (OLIVEIRA; WALTER; JUNG, 2018) can be explored for providing 3-DoF+ immersive experiences to AR/MR/VR users. The core of the proposed method, which uses some of the findings from the previous chapter, is explained in detail throughout the sections within Chapter 4.

Chapter 5 explores image oversegmentation for improving depth estimates in the context of multi-view 360° image processing. In particular, we motivate and adapt an efficient superpixel algorithm to the spherical domain. Then, this tailored algorithm is applied for (i) selecting reliable and scattered correspondence pairs candidates, and (ii) imposing spatial coherence within regions containing pixels sharing the same appearance.

Chapter 6 presents a framework for inferring depth from a single spherical image that can be coupled to any generic planar image monocular depth estimation algorithm. It consists of first inferring depth from overlapping planar patches extracted from the spherical image and then using a regularized minimization scheme to stitch the patches back to the sphere. We test three state-of-the-art methodologies as baseline methods and show that for all of them our approach leads to better results than applying the methods directly to the equirectangular projection and disjoint sections of the sphere.

Chapter 7 presents the conclusions of this work. Section 7.1 provides some final remarks on the achieved results, and Section 7.2 exposes promising directions for future work. Finally, a list of published papers related to this Dissertation is given in Section 7.3.

2 RELATED WORK

Before addressing the closest works to this Dissertation, we firstly review in Section 2.1 the foundations of spherical imaging. Then, in Section 2.2, we revise the epipolar geometry and present the particularities of the spherical camera model, since several studies related to our work also use such concepts. Section 2.3 discusses the choice among sparse or dense feature matching algorithms, and how they perform on spherical images. Finally, Section 2.4 presents the related literature for the problems of estimating the relative 3D camera poses and retrieving the 3D geometry from a set of image captures using *truly* omnidirectional cameras.

2.1 Foundations of Spherical Images

Throughout this text, we will use boldface variables for referring to points, vectors, and matrices. The latter is written in capital letter, while the others are shown in lowercase.

The core of spherical imaging is to project a world point $\mathbf{X} \in \mathbb{R}^3$ onto the unit sphere centered at $\mathbf{C} \in \mathbb{R}^3$ (both with respect to a world coordinate system), yielding an intersection point $\mathbf{x} \in S^2$ (AKIHIKO; ATSUSHI; OHNISHI, 2005). Consider a spherical camera with extrinsic parameters $[\mathbf{R}|\mathbf{t}]$, where $\mathbf{R} \in SO(3)$ is the rotation matrix, and $\mathbf{t} = -\mathbf{RC} \in \mathbb{R}^3$ the translation vector. Then, the projected (imaged) point $\mathbf{x}$ is given by (GUAN; SMITH, 2017b)

$$\mathbf{x} = \frac{\mathbf{RX} + \mathbf{t}}{\|\mathbf{RX} + \mathbf{t}\|_2}, \tag{2.1}$$

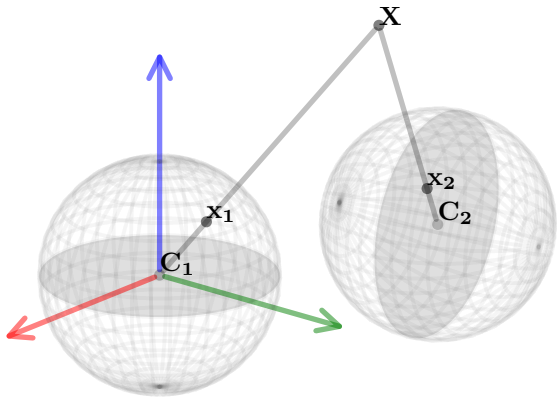
where $\|\cdot\|_2$ is the L2-norm. Figure 2.1 illustrates a world point $\mathbf{X}$ being projected onto two different spherical cameras, one of which is centered on the origin of and aligned to the world coordinate system, *i.e.,* $[\mathbf{I}|\mathbf{C_1} = \mathbf{0}]$, and the other not, having extrinsics $[\mathbf{R} \neq \mathbf{I}|\mathbf{t} = -\mathbf{RC_2} \neq \mathbf{0}]$. Here, $[\cdot]$ denotes the matrix/vector concatenation operator.

Note that a given imaged point $\mathbf{x}$ (in camera coordinates) has unit distance from the camera center, and thus it can be rewritten in terms of spherical coordinates $(r = 1, \theta, \phi)$ as (AKIHIKO; ATSUSHI; OHNISHI, 2005)

$$\mathbf{x} = [\cos\theta \sin\phi \;\; \sin\theta \sin\phi \;\; \cos\phi]^\top, \tag{2.2}$$

where $\theta \in [0, 2\pi)$ and $\phi \in [0, \pi)$. Since there is information associated to every position (θ, ϕ) on the sphere – *i.e.,* the light intensities captured by the omnidirectional camera –

Figure 2.1: The epipolar geometry of two spherical cameras. First camera is assumed to be centered on the origin and aligned to the world coordinate system, *i.e.,* $[\mathbf{I}|\mathbf{C_1} = \mathbf{0}]$, whilst the second camera has free positioning and rotation, meaning $[\mathbf{R}|\mathbf{t} = -\mathbf{C_2}]$. Projecting the world point $\mathbf{X}$ in the first and second cameras yields $\mathbf{x_1}$ and $\mathbf{x_2}$, respectively. Source: the author.



the whole image can be organized in a $[0, 2\pi) \times [0, \pi)$ planar representation. This representation is the so-called equirectangular projection of the sphere (TORII; HAVLENA; PAJDLA, 2009), which is widely employed among camera vendors and researchers (SU; GRAUMAN, 2017). Therefore, a given point $\mathbf{x}$ can be (approximately) mapped to a position (x, y) of a discrete image sized in $w \times h$ pixels, where

$$x = \left\lfloor \frac{\theta w}{2\pi} \right\rfloor \quad \text{and} \quad y = \left\lfloor \frac{\phi h}{\pi} \right\rfloor, \tag{2.3}$$

where $\lfloor \cdot \rceil$ is the floor rounding operator. Let us call $\psi : S^2 \to [0, w) \times [0, h)$ the function that does this mapping, and consider $\psi^{-1}(x, y)$ for the inverse mapping. Figure 2.2 exemplifies the mapping of a spherical image from the SUN360 dataset (XIAO et al., 2012) to its equirectangular projection.

In this study, as in many others (KOPF, 2016; ANDERSON et al., 2016; GUAN; SMITH, 2017b; PATHAK et al., 2017a; SU; GRAUMAN, 2017; HUANG et al., 2017), the input images are assumed to be represented in the equirectangular format. However, unless it is explicitly stated, all the algorithms work on the 3D unit vector representation of the image pixels, which are obtained by the inverse mapping of Equations (2.2) and (2.3).

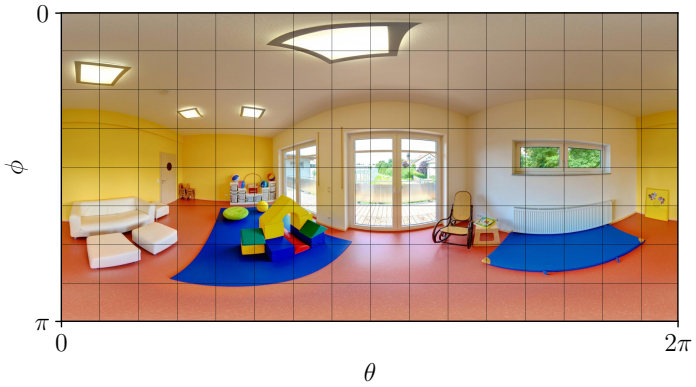
2.2 Spherical Epipolar Geometry

Epipolar geometry describes the geometrical relation between a pair of central projection cameras capturing a static scene (HARTLEY; ZISSERMAN, 2003) by exploring

Figure 2.2: Example of an equirectangular image. The intensity value from the projected point $\mathbf{x} = [\cos\theta \sin\phi \;\; \sin\theta \sin\phi \;\; \cos\phi]^\top$ is mapped to an integer pixel position (x, y) of a $w \times h$ equirectangular image where $x = \lfloor \frac{\theta w}{2\pi} \rfloor$ and $y = \lfloor \frac{\phi h}{\pi} \rfloor$. Source: the author.

(a) Spherical image

(b) Equirectangular projection of (a)



the coplanarity property between matched image points and the cameras centers (AKIHIKO; ATSUSHI; OHNISHI, 2005). Since an omnidirectional camera is, in fact, a central projection camera, so as pinhole-based ones are, this definition applies to the spherical camera model.

Consider $\mathbf{x_1} = [x_1 \; y_1 \; z_1]^\top$ and $\mathbf{x_2} = [x_2 \; y_2 \; z_2]^\top$ as projections of $\mathbf{X}$ onto two spherical cameras, as depicted in Figure 2.1. Without loss of generality, it is assumed that the first camera is canonical, *i.e.,* it is located at the origin and aligned to the world coordinate system. Unlike the perspective case, the spherical camera model has no intrinsic parameters (GUAN; SMITH, 2017b), meaning that the Essential $\mathbf{E}$ and the Fundamental matrices $\mathbf{F}$ are the same.

It is worth mentioning that although there are different camera settings for capturing the $360° \times 180°$ environment – like catadioptric devices (CRUZ-MOTA et al., 2012), rigs of perspective cameras (ANDERSON et al., 2016) or paired hemispherical fisheye lenses (DENG et al., 2008) – they all can produce equirectangular-like images via stitching processes. Once in this representation, which agrees with the spherical camera model, the images are free of intrinsic calibration.

Having said that, it is safe to state that the projections $\mathbf{x_1}$ and $\mathbf{x_2}$ are related according to the epipolar constraint (PAGANI et al., 2011; AKIHIKO; ATSUSHI; OHNISHI, 2005; PATHAK et al., 2016b; TROIANI et al., 2013)

$$\mathbf{x_2}^\top \mathbf{E} \mathbf{x_1} = 0, \quad \mathbf{E} = [\mathbf{t}]_\times \mathbf{R}, \tag{2.4}$$

where $[\mathbf{t}]_\times$ is the skew-symmetric matrix corresponding to the cross-product with $\mathbf{t}$ (OZYE-

SIL et al., 2017). Note that the Essential matrix $\mathbf{E}$ from Equation (2.4) encodes the *relative* translation and rotation between the two cameras, *i.e.,* the pose of the first camera is assumed to be $[\mathbf{I}|\mathbf{0}]$. Moreover, because $\mathbf{E}$ is defined up to an unknown scale (YANG; LI; JIA, 2014), only five from the six DoFs involving both the cameras can be resolved by exploring the epipolar constraint.

Geometrically, Equation (2.4) means that the projections $\mathbf{x_1}$ and $\mathbf{x_2}$ and the camera centers $\mathbf{C_1}$ and $\mathbf{C_2}$ (and also the world point $\mathbf{X}$) lie on the same plane, which is called the epipolar plane (HARTLEY; ZISSERMAN, 2003). In the perspective case, projecting the epipolar plane onto the image leads to a line, which is useful information for efficiently implementing many SM methods (SCHARSTEIN; SZELISKI; ZABIH, 2001). Differently, the projection of the epipolar plane onto a spherical camera creates a great circle (BRUNTON; LANG; DUBOIS, 2012), as illustrated in Figure 2.3. In Figure 2.3a, we show five (zoomed for visualization) points in a region of interest (RoI) of a first spherical image and, after translating the camera, we plot the corresponding great circles[1] in the second capture. They are shown in the equirectangular format in Figure 2.3b. Also, Figure 2.3c shows those great circles projected onto (rotated by $0°$, $90°$, $180°$, and $270°$ around the vertical axis versions of) the sphere, where we do believe it may be easier to convince the reader that those are, in fact, circles.

There are two observations we need to highlight here. The first one is that for a fixed projected point $\mathbf{x_1}$ (or $\mathbf{x_2}$), Equation (2.4) is always satisfied by every other imaged point $\mathbf{x_2}$ (or $\mathbf{x_1}$) that lies on the underlying great circle, no matter this is the actual matching or not. For instance, if we fix the green point in Figure 2.3a, then all the points located over the green great circle in Figure 2.3b satisfy Equation (2.4). The other note is that the two points of intersection of all the great circles, as shown in Figures 2.3b and 2.3c, form the epipoles that coincide with the direction of the relative translation. Keeping this concept in mind for the future will be important.

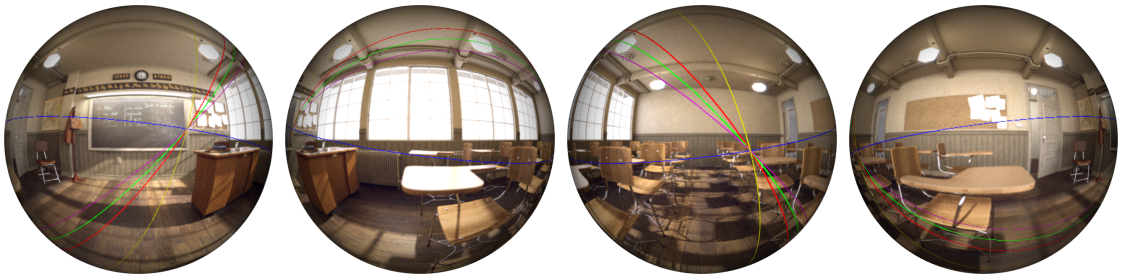
In practice, we have only two spherical images, and unfortunately, we know neither the projections matchings nor the (extrinsic) camera parameters. We will further discuss what are the ways to obtain both sparse and dense set of matched points from a pair of spherical images in the upcoming sections. By now, it is important to notice that if we do know a set of corresponding projected points, then we can compute the Essential matrix $\mathbf{E}$ and recover the relative camera motion. Furthermore, once we know a set of matchings and the relative camera, then we can estimate the 3D position of the world

[1] For this example, we rely on the synthetic Classroom scene rendered with Blender. The great circles are drawn with hand-picked "keypoints" and ground-truth parameters for the relative camera pose.

Figure 2.3: Projection of epipolar planes onto spherical images. Imaged points are selected and fixed in (a), and the underlying great circles are drawn in (b). The image in (b) is mapped to the sphere, and plotted in four different angles in (c). Both intersection points of all the great circles coincide with the epipoles. Source: the author.

(a) RoI (b) Great circles seen on the equirectangular projection of the sphere

(c) Great circles seen on rotated by 0°, 90°, 180°, and 270° (around the vertical axis) versions of the sphere



points associated with those imaged points. The latter is our central goal.

There is a number of strategies for estimating the (unscaled) Essential matrix $\mathbf{E}$ from pairs of correspondences (HARTLEY, 1997; NISTÉR, 2004; XU; MULLIGAN, 2008; YANG; LI; JIA, 2014). Several approaches (PAGANI et al., 2011; HUANG; TSENG, 2016; PATHAK et al., 2016b; XU et al., 2017) explore the eight-point algorithm (8-PA) (LONGUET-HIGGINS, 1987; HARTLEY, 1997) to extract the Essential matrix, possibly refining it using a non-linear method (GUAN; SMITH, 2017b; HUANG et al., 2017; PAGANI et al., 2011). We next briefly revise the 8-PA and how to extract the extrinsic parameters from the Essential matrix.

2.2.1 The Eight-Point Algorithm

The standard 8-PA for pinhole-based cameras (LONGUET-HIGGINS, 1987) assumes that the positions of the matched points are represented with homogeneous coordi-

nates. Later, Hartley (1997) found that a pre-processing step including data normalization and centering is needed to make the 8-PA robust. Contrarily, for the spherical images case, there is no need to normalize the feature location vectors. The projected points are already unitary, and therefore they can be directly used as input to the 8-PA (PAGANI et al., 2011; PATHAK et al., 2016b).

The solution for $\mathbf{E}$ using the 8-PA starts by creating a $n \times 9$ matrix $\mathbf{A}$ (with $n \geq 8$), for which the i-th row is given by

$$\mathbf{A}_i^\top = \left[x_1^i x_2^i \; x_1^i y_2^i \; x_1^i z_2^i \; y_1^i x_2^i \; y_1^i y_2^i \; y_1^i z_2^i \; z_1^i x_2^i \; z_1^i y_2^i \; z_1^i z_2^i\right], \tag{2.5}$$

where the superscript $(^i)$ stands for the i-th correspondence pair $(\mathbf{x_1}^i, \mathbf{x_2}^i)$, $i = 1, 2, \ldots, n$. Note that, because of the assumption of using homogeneous coordinates in the case of perspective images, Equation (2.5) slightly differs from Equation (3) in Hartley (1997).

Then, the problem of estimating $\mathbf{E}$ can be cast as a constrained homogeneous linear system

$$\mathbf{A}\mathbf{e} = \mathbf{0} \quad \text{s.t.} \quad \|\mathbf{e}\| = 1, \tag{2.6}$$

where

$$\mathbf{e} = [e_{11} \; e_{21} \; e_{31} \; e_{12} \; e_{22} \; e_{32} \; e_{13} \; e_{23} \; e_{33}]^\top \tag{2.7}$$

contains the elements of the Essential matrix $\mathbf{E} = [e_{ij}]_{3\times3}$ in vector form. The least-squares solution of Equation (2.6) can be obtained by using the singular value decomposition (SVD) (HARTLEY, 1997), where $\mathbf{e}$ is the eigenvector associated to the smallest eigenvalue of $\mathbf{A}^\top\mathbf{A}$, corresponding to the last column of $\mathbf{V_A}$, *i.e.,* $\mathbf{e} = \mathbf{V_{A9}}$, where $\mathbf{U_A}\mathbf{\Sigma_A}{\mathbf{V_A}}^\top = \mathbf{A}$ is the SVD of matrix $\mathbf{A}$. The adequate rearrangement of $\mathbf{e}$ leads to an initial estimate of the Essential matrix, which we will refer here as $\tilde{\mathbf{E}}$.

2.2.2 Extracting 5-DoF Pose from the Essential Matrix

By definition, the Essential matrix $\mathbf{E}$ is rank 2 (HARTLEY, 1997), with two equal singular values. However, $\tilde{\mathbf{E}}$ may have been estimated from noisy correspondence pairs. To enforce this constraint, we define $\mathbf{U}_{\tilde{\mathbf{E}}}\mathbf{\Sigma}_{\tilde{\mathbf{E}}}{\mathbf{V}_{\tilde{\mathbf{E}}}}^\top = \tilde{\mathbf{E}}$ as the SVD of $\tilde{\mathbf{E}}$, and make the two largest singular values of $\tilde{\mathbf{E}}$ equal (by averaging for instance) and set the remaining

one to zero (HARTLEY; ZISSERMAN, 2003, p.294). Explicitly,

$$\mathbf{\Sigma_E} = \operatorname{diag}\left(\frac{\tilde{\sigma}_{11}+\tilde{\sigma}_{22}}{2}, \frac{\tilde{\sigma}_{11}+\tilde{\sigma}_{22}}{2}, 0\right), \quad \text{where} \quad \mathbf{\Sigma_{\tilde{E}}} = \operatorname{diag}(\tilde{\sigma}_{11}, \tilde{\sigma}_{22}, \tilde{\sigma}_{33}). \tag{2.8}$$

Moreover, one needs to check whether the determinants of $\mathbf{U_{\tilde{E}}}$ and $\mathbf{V_{\tilde{E}}}$ are negative. If so, then the sign of the last column of these matrices is inverted. This process avoids to estimate improper rotation matrices (rotations with reflections) (XU; MULLIGAN, 2008; LIM; BARNES; LI, 2010). Formally,

$$\mathbf{U}_{\mathbf{E}3} = \begin{cases} \mathbf{U}_{\tilde{\mathbf{E}}3}, & \text{if} \det(\mathbf{U_{\tilde{E}}}) > 0 \\ -\mathbf{U}_{\tilde{\mathbf{E}}3}, & \text{otherwise} \end{cases} \tag{2.9}$$

and

$$\mathbf{V}_{\mathbf{E}3} = \begin{cases} \mathbf{V}_{\tilde{\mathbf{E}}3}, & \text{if} \det(\mathbf{V_{\tilde{E}}}) > 0 \\ -\mathbf{V}_{\tilde{\mathbf{E}}3}, & \text{otherwise} \end{cases}. \tag{2.10}$$

Finally, the Essential matrix is posed as $\mathbf{E} = \mathbf{U_E \Sigma_E V_E}^\top$. This is the closest singular matrix from the (possibly noisy) matrix $\tilde{\mathbf{E}}$ under the Frobenius norm (HARTLEY, 1997).

Because the described SVD-based method minimizes the objective function, Equation (2.6), in a least-squares sense, it is sensitive to outliers. We discuss some attempts for measuring the perturbation in $\tilde{\mathbf{E}}$ caused by noisy matches and outliers in Section 2.2.3, and present a more conclusive analysis about it, focusing on the spherical camera model case, in Chapter 3.

Once $\mathbf{E}$ is estimated, it is possible to extract both the rotation matrix $\mathbf{R}$ and the (unitary) translation vector $\mathbf{t}$ that relate the two spherical cameras. As in the pinhole camera model case, there are two candidates for $\mathbf{R}$ and two candidates for $\mathbf{t}$ (HUANG; TSENG, 2016). Those candidates will be denoted by $\tilde{\mathbf{R}}_\mathbf{1}, \tilde{\mathbf{R}}_\mathbf{2}$ and $\tilde{\mathbf{t}}_\mathbf{1}, \tilde{\mathbf{t}}_\mathbf{2}$, respectively. The estimate for the rotation matrix $\mathbf{R}$ is equal to one of the following two possibilities:

$$\tilde{\mathbf{R}}_\mathbf{1} = \mathbf{U_E}[\mathbf{e}_2 \ -\mathbf{e}_1 \ \mathbf{e}_3]\mathbf{V_E}^\top \quad \text{or} \quad \tilde{\mathbf{R}}_\mathbf{2} = \mathbf{U_E}[-\mathbf{e}_2 \ \mathbf{e}_1 \ \mathbf{e}_3]\mathbf{V_E}^\top, \tag{2.11}$$

where $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ form the natural basis for the Euclidean space. The estimate for the

translation vector $\mathbf{t}$ is either

$$\tilde{\mathbf{t}}_1 = -\mathbf{U}_{\mathbf{E}3} \quad \text{or} \quad \tilde{\mathbf{t}}_2 = \mathbf{U}_{\mathbf{E}3}. \tag{2.12}$$

For traditional perspective cameras, the pair $(\tilde{\mathbf{R}}, \tilde{\mathbf{t}}), \tilde{\mathbf{R}} \in \{\tilde{\mathbf{R}}_1, \tilde{\mathbf{R}}_2\}, \tilde{\mathbf{t}} \in \{\tilde{\mathbf{t}}_1, \tilde{\mathbf{t}}_2\}$ that better describes the scene is the one that makes all the visible points in front of the cameras (projection planes) (HARTLEY; ZISSERMAN, 2003). However, in the spherical camera model, there is no concept of front and back. Therefore, as a solution, one may check if both inequalities

$$\mathbf{x}_1{}^\top \tilde{\mathbf{X}} > 0 \quad \text{and} \quad \mathbf{x}_2{}^\top (\tilde{\mathbf{R}}\tilde{\mathbf{X}} + \tilde{\mathbf{t}}) > 0 \tag{2.13}$$

are simultaneously satisfied (PAGANI; STRICKER, 2011), where $\tilde{\mathbf{X}}$ is a candidate world point computed from the projection pair $(\mathbf{x}_1, \mathbf{x}_2)$ and a given pose candidate $(\tilde{\mathbf{R}}, \tilde{\mathbf{t}})$ through the direct linear transformation (DLT) (HARTLEY; ZISSERMAN, 2003). The intuition behind (2.13) is that both the projections and the world point estimated with $(\tilde{\mathbf{R}}, \tilde{\mathbf{t}})$ are pointing towards the same direction. If those inequalities are satisfied, then that pose candidate explains the projections. Although simple, a drawback of this strategy is that, for each correspondence pair, four "world points" are computed only to decide which pose candidate is the best choice.

Using a different approach, Guan and Smith (2017b) propose to look for the sign of the scalars $a, b \in \mathbb{R}$ that relate both the projections $\mathbf{x}_1$ and $\mathbf{x}_2$ to the actual world point $\mathbf{X}$ given a candidate pair $(\tilde{\mathbf{R}}, \tilde{\mathbf{t}})$, *i.e.,*

$$a\mathbf{x}_1 = \mathbf{X} \quad \text{and} \quad b\mathbf{x}_2 = \tilde{\mathbf{R}}\mathbf{X} + \tilde{\mathbf{t}}. \tag{2.14}$$

Since a and b in Equation (2.14) are meant as the scales for the projections, both are expected to be positive, *i.e.,* there is no sense in finding negative depths. Therefore, instead of estimating $\tilde{\mathbf{X}}$, one can solve a 3×2 linear system formed by the terms of $b\mathbf{x}_2 - a\tilde{\mathbf{R}}\mathbf{x}_1 - \tilde{\mathbf{t}} = \mathbf{0}$, and test the sign of the solutions. If both $a > 0$ and $b > 0$, then a vote for $(\tilde{\mathbf{R}}, \tilde{\mathbf{t}})$ is cast. The candidate pose which explains most of the correspondences is chosen. Note that if all the matchings are noiseless, then the pair $(\tilde{\mathbf{R}}, \tilde{\mathbf{t}})$ is voted unanimously.

It is important to highlight that there are other approaches for extracting the motion parameters from the estimated Essential matrix, as in Wang and Tsui (2000), Nistér (2004), Yang, Li and Jia (2014). We focus on the technique described in this section

since it is the simplest linear solution, being widely employed in recent works focused on spherical images, such as Guan and Smith (2017b). Furthermore, the output of this algorithm is also used in non-linear refinement approaches of Pagani et al. (2011), Pathak et al. (2016a), Pathak et al. (2018).

2.2.3 Perturbation Analysis of Epipolar Matrices

To the best of our knowledge, there is no study focusing on perturbation analysis of the 8-PA when considering the spherical camera model. Hence, the literature review of this part is focused on classic works that consider the pinhole camera model, although, as mentioned before, the 8-PA linear formulation is generic for central projection cameras. Note that the 8-PA can be applied to estimate either the Fundamental or the Essential matrix from calibrated and uncalibrated cameras, respectively, in intrinsic parameters sense.

Estimating the uncertainty when computing Epipolar (Fundamental or Essential) matrices has been studied by several authors (CSURKA et al., 1997; HARTLEY, 1997; MAIR; SUPPA; BURSCHKA, 2013; MÜHLICH; MESTER, 1998; WENG; HUANG; AHUJA, 1989; SUR; NOURY; BERGER, 2008), focusing on the relationship between correspondence errors and the Epipolar matrices themselves or directly on the errors of estimated 3D structure and/or 5-DoF pose.

Weng, Huang and Ahuja (1989) presented an error analysis of the Essential matrix estimation based on first order perturbations, as well as the errors in the estimated pose (translation and rotation) extracted from this matrix. For the Essential matrix perturbation analysis, they relate the covariance matrix of the matching errors with the covariance of the Essential matrix. This assumption implies that the variance of the feature detector error must be determined, which is very dependent on the detector itself and the scene, and very sensitive to outliers. Furthermore, they assumed the un-normalized 8-PA, which was later shown by Hartley (1997) that leads to numerical instabilities in the pinhole case.

Mühlich and Mester (1998) related the error produced by the 8-PA with the perturbation of eigenvalues and singular values, since the Epipolar matrix is extracted through the SVD. They assumed that the covariance matrix of the correspondence matching errors is known to obtain a bound for the Essential matrix error, and use this bound to propose a new feature normalization scheme. Mair, Suppa and Burschka (2013) extended the analysis in Weng, Huang and Ahuja (1989) by including the normalization schemes presented in Hartley (1997) and Mühlich and Mester (1998), focusing on V-SLAM ap-

plications. Notice that the methods from both Mühlich and Mester (1998) and Mair, Suppa and Burschka (2013) present the drawback of assuming known feature matching variances.

Csurka et al. (1997) presented an error analysis of the Fundamental matrix $\mathbf{F}$ obtained by minimizing the (non-linear) sum of epipolar distances. They model $\mathbf{F}$ as a random vector, such that the mean of the distribution is the actual matrix and the covariance encodes the uncertainty errors. They assume that outliers were rejected in a previous step so that the analysis is focused only on noisy correspondences. Sur, Noury and Berger (2008) follow a similar path but focus on the errors of Epipolar matrix estimation using the 8-PA. However, as in Csurka et al. (1997), they present the uncertainty as a covariance matrix and discard the presence of outliers, which limits the application of their method. Also, they consider the un-normalized version of the 8-PA, as Weng, Huang and Ahuja (1989).

In his seminal work, Hartley (1997) evaluated the condition number of the measurement matrix used in the 8-PA, suggesting an alternative normalized version that is more stable numerically. The core of his analysis was that using "raw" homogeneous coordinates (just appending the value 1 to the pixel coordinates) leads to a magnitude imbalance and hence ill-conditioned matrices. When using normalized homogeneous coordinates (unit vectors) as suggested in Hadfield, Lebeda and Bowden (2 018) and used in most approaches that explore spherical cameras (GUAN; SMITH, 2017b; PAGANI; STRICKER, 2011; PATHAK et al., 2016; PATHAK et al., 2018; PATHAK et al., 2017), the reasoning used by Hartley no longer applies.

Despite the existence of works that perform perturbation analysis of Epipolar matrices, most of them assume a distribution model for the correspondence matching errors and disregard the impact of outliers. They also empirically evaluate the effect of the camera FoV on the errors, without any mathematical formalism. In Chapter 3, we present our bounds for Epipolar matrices based on correspondence errors and provide a tighter relationship between the distribution of matched features and error propagation. In particular, we show that when the features are spatially well distributed, then the estimate of the Epipolar matrix tends to be more accurate. In particular, we conclude that using spherical cameras can potentially lead to good estimates for the Essential matrix when descriptors tailored to the spherical domain (CRUZ-MOTA et al., 2012; ZHAO et al., 2014) are used.

2.3 Finding Correspondences in Spherical Images

In order to obtain the required correspondences for the 8-PA, *sparse* feature-based matching algorithms are the most popular choice. Traditionally, those keypoint matching algorithms look for salient (in some aspect) image points, encoding their local information discriminatingly while being robust to affine transformations, noise and contrast changes (CRUZ-MOTA et al., 2012). On the other hand, dense feature matching, which is commonly accomplished by using *dense* optical flow methods, minimize both data and smoothness terms that model the brightness constancy of the imaged scene over time (RADKE, 2012). Because of this assumption, optical flow algorithms generally do not work well on image pairs for which the camera poses differ considerably.

2.3.1 Sparse Feature Matching

In this literature review, we could find two standard keypoint algorithms (ALCANTARILLA; NUEVO; BARTOLI, 2013; MOREL; YU, 2009) being applied as part of SfM pipelines based on spherical images (PAGANI et al., 2011; PATHAK et al., 2016). Furthermore, three methods (CRUZ-MOTA et al., 2012; ZHAO et al., 2014; GUAN; SMITH, 2017a) specifically developed for detecting and matching keypoints on the spherical domain were found. We briefly discuss those five methodologies in the following.

Accelerated KAZE (A-KAZE) (ALCANTARILLA; NUEVO; BARTOLI, 2013), a modified version of KAZE (ALCANTARILLA; BARTOLI; DAVISON, 2012), is a fast multi-scale keypoint detection and extraction algorithm which exploits non-linear scale spaces in perspective images. In the original paper (ALCANTARILLA; NUEVO; BARTOLI, 2013), the authors claim that A-KAZE outperforms methods like KAZE, SIFT (LOWE, 2004) and ORB (RUBLEE et al., 2011) regarding repeatability, when applied artificial rotation, blurring, compression, noise, etc., at the same time it spends lower processing time than the first two methods. The pipeline of the A-KAZE algorithm consists basically of: (i) building the non-linear scale-space using fast explicit diffusion schemes in a pyramidal way; (ii) searching for maxima responses in scale and spatial locations of the scale-normalized determinant of the Hessian of each filtered image in the scale-space; and (iii) describing the keypoint by the modified-local difference binary (M-LDB) algorithm using gradient and intensity information from the scale space. Finding the dominant local orientation around the keypoint and keeping the relationship between

the grid size of the M-LDB and the scale of the filtered images makes the descriptor invariant to rotation and scale. According to Pathak et al. (2016), Pathak et al. (2018), A-KAZE works well under distortions and, for that reason it was used in their spherical SfM pipeline.

Affine SIFT (ASIFT) (MOREL; YU, 2009) is a planar algorithm based on the strengths and weaknesses of the consecrated SIFT keypoint extractor and descriptor. In the original proposal, ASIFT is supposed to replace the SIFT keypoint extractor, keeping the descriptor unaltered. In practical terms, while SIFT achieves invariance to translation, rotation and scale, ASIFT adds invariance to "viewpoint changes", proving to be fully affine transformation invariant (MOREL; YU, 2009). In a nutshell, ASIFT becomes viewpoint invariant by simulating a comprehensive set of perspective distortions that the images can suffer and comparing them by SIFT algorithm. In order to minimize the running time of their algorithm, Morel and Yu (2009) propose to use a two-resolution mechanism. The viewpoint simulations are firstly performed in the lower resolution images and, if some keypoint matching is obtained, they are reapplied in the full-size images. The authors show their improvements *w.r.t.* SIFT, among others, when dealing with significant perspective view changes. Pagani et al. (2011) successfully applied a modified version of ASIFT in an SfM pipeline based on spherical images.

Although promising results have been shown when applying standard keypoint matching algorithms to spherical images (PAGANI et al., 2011; PATHAK et al., 2016), the scientific community argue that it is just a naive approach (GUAN; SMITH, 2017a), since it is not even correct in a geometric sense (ZHAO et al., 2014). The distortions induced by the mapping of spherical images to (any) planar representation are not affine (SU; GRAUMAN, 2017) and, more than that, they depend on the objects position in the scene (CRUZ-MOTA et al., 2012) and camera orientation (GUAN; SMITH, 2017a).

In this sense, Cruz-Mota et al. (2012) completely adapt the planar SIFT algorithm to the spherical domain. Each step of the original SIFT keypoint detector is performed on its spherical counterpart: (i) creation of the scale-space representation, (ii) computation of the Difference of Gaussians (DoGs); and (iii) local extrema extraction and filtering. The proposed local spherical descriptor, as its planar version, is invariant to rotation and scale changes. Cruz-Mota et al. (2012) assess the performance of their algorithm, spherical SIFT (SSIFT), against its planar version through the repeatability metric on spherical images synthetically rotated and corrupted with noise. The results presented in their paper show that the SSIFT algorithm is considerably more robust to the omnidirectional

sensor distortions than planar SIFT. Guan and Smith (2017b) use SSIFT algorithm as a fundamental building block for their spherical SfM-based method and application.

Zhao et al. (2014) propose a scale-invariant version of the FAST detector (ROSTEN; DRUMMOND, 2006) coupled to a rotation invariant ORB-like descriptor, both operating on a hexagonal geodesic grid representation of the sphere. More precisely, they presented a fast and robust algorithm that constructs binary features to describe image keypoints on the spherical domain, which they named spherical ORB (SPHORB). The way omnidirectional images are represented is probably one of the main insights of their work. The authors show that the geodesic grid they use has important properties when dealing with binary features that, differently from cubic and equirectangular representations of the spherical images, helps to speed up the SPHORB algorithm. Basically, the spherical FAST detector looks for points that are sufficiently brighter or darker than their neighborhood, and attributes a weight for how distinguishable from its vicinity they are. This procedure is performed using a pyramidal structure, enabling SPHORB with robustness to scale changes. The bit-string that describes each of the selected keypoints is nothing but a series of intensity comparisons that are further reoriented to keep the algorithm invariant to rotations. Zhao et al. (2014) compare their algorithm, among others, with SSIFT and planar versions of ORB and SIFT. Their results point out the effectiveness of SPHORB regarding repeatability, precision and recall under synthetic rotation and noise corruption. The authors also present some statistics regarding the proportion of correct matchings in real pairs of images on small camera change setups.

Guan and Smith (2017a) propose to adapt another standard binary descriptor to the spherical domain, namely the BRISK (LEUTENEGGER; CHLI; SIEGWART, 2011) algorithm. The authors named their method as BRISK on the sphere (BRISKS). Similarly to Zhao et al. (2014), the keypoint detector is also an adaptation of the FAST algorithm to operate on a multi-scale geodesic grid representation of the sphere. Most of the novelty lies on the descriptor. It comprises four steps: (i) direction assignment of the feature; (ii) rotation of the local neighborhood; (iii) sampling of the neighborhood; and (iv) construction of the descriptor by comparing intensities. These steps are somehow related to those of the SPHORB algorithm. BRISKS was originally assessed regarding repeatability under deformations caused by rotation and translation, and illumination changes using a rendered scene. Furthermore, Guan and Smith (2017a) test the robustness of their method against different noise levels. Although similar to the SPHORB algorithm, the authors compare BRISKS only to SSIFT and the traditional SIFT methods, proving to

outperform both of them, according to the metrics above.

A preliminary study about the performance of sparse feature extraction, description, and matching algorithms when applied to spherical images was presented in Silveira and Jung (2017). In such study, we assessed how well ASIFT, A-KAZE, SSIFT, and SPHORB performed on the two-view relative pose estimation task when using the 8-PA. We concluded that the estimated 5-DoF poses using filtered features estimated via planar and spherical algorithms did not differ so much. Especially, planar algorithms do not work well for highly distorted images (*e.g.* due to the camera model, objects too close to the sensor or rotation), presenting concentrated features on the equator region of the spherical image pair, differing from the algorithms tailored to the spherical domain.

2.3.2 Dense Feature Matching

Since we are interested in producing dense 3D maps, a natural choice would be to use a dense optical flow algorithm to obtain the matching pairs. However, the equirectangular projection (SU; GRAUMAN, 2017) highly distorts the information depending on its location on the imaged scene (CRUZ-MOTA et al., 2012), being particularly high near the poles of the sphere (FERREIRA; SACHT; VELHO, 2017). Note that there are other ways to represent the sphere on the plane, but all of them introduce some distortion as well (SU; GRAUMAN, 2017). Hence, applying *traditional* optical flow algorithms directly to a pair of images captured in very different poses tends to produce poor results.

Alibouch et al. (2012) adapt a phase-based optical flow algorithm originally suited for perspective images to the omnidirectional context, and explores signal phase information from band-pass filters for modelling the motion field. They propose to perform the spatial filtering using spherical Morlet wavelets (ANTOINE et al., 2002), and reformulate the phase gradient constraint equation from Gautama and Hulle (2002) to the sphere. The authors tested their method for *semi-dense* optical flow using catadioptric image sequences, recalling that catadioptric sensors capture all the 360° horizontal FoV but have a limited vertical FoV (NAYAR, 1997; AGGARWAL; VOHRA; NAMBOODIRI, 2016). Their analysis was further extended in Alibouch et al. (2016), and there is no public implementation of their method.

Conversely to phase-based optical flow methods, Radgui et al. (2011) proposed a multi-channel decomposition approach that presents a compromise between robustness and computational time. Their method is also based on convolutions of the images with

spherical Morlet wavelet filter banks and has been outperformed by Alibouch et al. (2016). The algorithm from Radgui et al. (2011) does not produce a dense optical flow map and has no publicly available implementation or source code.

Kirisits, Lang and Scherzer (2014) proposed several variational regularization methods for the estimation and decomposition of motion fields on spherical data. Instead of working with a pixel-level representation of spherical images, the authors proposed to compute the optical flow on a icosahedral approximation of the sphere (icosphere). The output of their method is also a *semi-dense* flow map which is given for each of the icosphere faces. The authors tested their method on time-lapses from microscopic multi-channel fluorescence imagery mapped to hemispherical images, but their method can tackle only very small baseline induced motion.

To the best of our knowledge, there is no *dense large-displacement optical flow* algorithm specifically designed to work on the *spherical domain*. Thus, the techniques for pose estimation and 3D geometry recovery that we found so far, which will be discussed in Section 2.4, rely either on sparse features or dense wide baseline optical flow algorithms suited for planar images. The latter, besides producing dense correspondences, need to cope simultaneously with both small and large displacements during the matching assignments. Traditionally, the disparities from a pair of perspective images depend basically on the camera baseline and how far the scene objects are from the camera. When dealing with equirectangular images, besides that, the optical flow algorithm needs to be able to match information largely displaced on the south and north poles of the sphere, where there is, in fact, an oversampling (FERREIRA; SACHT; VELHO, 2017).

Although not being ideal from the geometric point of view, the optical flow can be computed on $w \times h$ equirectangular images and then mapped to the sphere for posterior processing. The flow components $\mathbf{u} = [u_x, u_y]^\top$ at a given pixel position $\mathbf{p} = (x, y)$ of the first image are transformed to 3D Euclidean velocities via (GLUCKMAN; NAYAR, 1998):

$$\begin{bmatrix} U_x \\ U_y \\ U_z \end{bmatrix} = \begin{bmatrix} -\sin\theta\sin\phi & \cos\theta\cos\phi \\ \cos\theta\sin\phi & \sin\theta\cos\phi \\ 0 & -\sin\phi \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix}, \tag{2.15}$$

where the Jacobian matrix relates spherical coordinates $\theta = \frac{2\pi x}{w}$ and $\phi = \frac{\pi y}{h}$ (recall Equation (2.3)) to the 3D Euclidean space. We will further refer to this transformation as $\Psi^{-1}(\mathbf{p}, \mathbf{u})$.

Both purely translational and purely rotational motion draw known flow patterns on the sphere, which are documented in Nelson and Aloimonos (1988). On the one hand, the flow field of pure translation movement is characterized by a focus of expansion and a focus of contraction diametrically separated. Those foci are the epipoles, and any flow is parallel to the geodesics connecting them (NELSON; ALOIMONOS, 1988). On the other hand, any flow in purely rotational motion forms a loop around the rotation axis. Figure 2.4 exemplifies these two motion patterns. The knowledge about these patterns proved useful for the pose estimation problem (PATHAK et al., 2017a).

In this review, we found two dense optical flow algorithms (FARNEBÄCK, 2003; WEINZAEPFEL et al., 2013) being applied to spherical image pairs either for estimating the camera pose or depth. The Farneback's method (FARNEBÄCK, 2003) is a fast optical flow algorithm which is efficiently solved by a hierarchical scheme of separable convolutions. A multi-scale approach allows to Farneback's algorithm improves the chance for a good displacement estimate. However, as pointed out in the original work (FARNEBÄCK, 2003), the main weakness of this algorithm is the assumption of a slowly varying displacement field, causing discontinuities to be smoothed out. Despite that, Farneback's optical flow is used for helping to estimate the pose from a pair of spherical captures in Pathak et al. (2015), Pathak et al. (2016a), Pathak et al. (2017a).

A more robust optical flow algorithm, namely DeepFlow (WEINZAEPFEL et al., 2013), is considered for estimating depth from two equirectangular images in Pathak et al. (2016), Pathak et al. (2017). The main focus of DeepFlow is to handle large displacements efficiently. To accomplish that, the authors firstly estimate a constrained quasi-dense set of correspondences, what they call Deep Matching. The Deep Matching algorithm is built in a multi-stage scheme that retrieve adequate correspondences even in weakly-textured regions and from deformed objects. Then, those correspondences are blended to a non-convex and non-linear minimization framework, which properly weights the classic data and smoothness terms, and a novel matching term. The whole process is what the authors call DeepFlow.

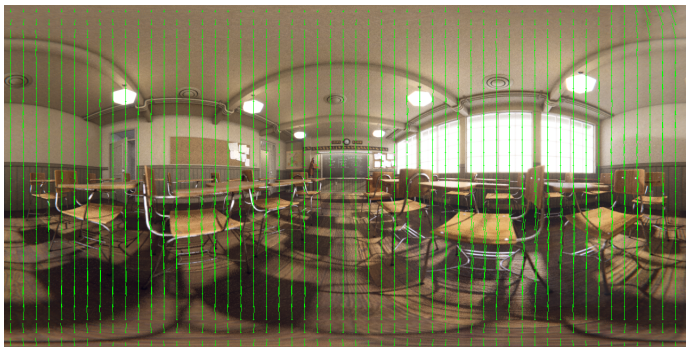
2.4 Pose Estimation and 3D Scene Reconstruction

As in traditional pinhole camera model, it is possible to extract relative poses and 3D information when two or more spherical views of the same scene are available. In fact, camera pose estimation is typically used in SfM pipelines, feeding a calibrated 3D recon-

Figure 2.4: Optical flow estimates of captures after (b),(c) pure translational and (d),(e) pure rotational movements *w.r.t.* (a) a canonical image. Optical flow is computed on the equirectangular images and then projected to the sphere. Source: the author.

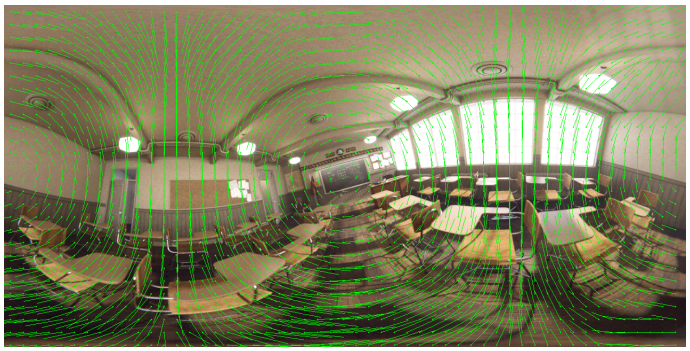
(a) Reference capture

(b) Translational optical flow computed in the equirectangular image representation.



(c) Translational optical flow projected onto the sphere

(d) Rotational optical flow computed in the equirectangular image representation.



(e) Rotational optical flow projected onto the sphere

struction stage. Aside from the 3D reconstruction context, pose estimation can also be used in other applications, such as video stabilization (GUAN; SMITH, 2017b), outdoor GPS-less mapping and localization (TORII; HAVLENA; PAJDLA, 2009) or spherical video inpainting (XU et al., 2016; XU et al., 2017).

Although the vast majority of 3D estimation techniques for perspective cameras explore two or more views (SCHARSTEIN; SZELISKI; ZABIH, 2001; SEITZ et al., 2006; OZYESIL et al., 2017), some authors (LIU et al., 2016; EIGEN; FERGUS, 2015;

GODARD; Mac Aodha; BROSTOW, 2017) have tried to answer the question "is it possible to infer depth from a single image?", presenting good results based on learning approaches. As in the multi-view setup, a similar question can be posed in spherical context. In the following, we revise the literature for 5-DoF/6-DoF pose and depth estimation based on two or more spherical images in Section 2.4.1, and then the ill-posed depth estimation problem from a single view in Section 2.4.2.

2.4.1 Pose and Depth Estimation from Two or More Spherical Images

Kim and Hilton (2013) propose a hierarchical partial differential equation (PDE)-based algorithm to estimate the disparities of stereo-aligned panoramic image pairs. By using several *stereo pairs* from the target environment, the authors can generate a reliable 3D structure with attenuated occlusion-caused issues and well defined textured regions. To do so, a mesh representation is extracted by triangulation from each disparity map, which is registered with the others through an adaption of the iterative closest point (ICP) algorithm (RUSINKIEWICZ; LEVOY, 2001) with surface reliability checking. The images used in Kim and Hilton (2013) are captured by commercial off-the-shelf rotating line-scan cameras equipped with 180° full-frame fish-eye lenses. Since their method relies on solving PDEs in multiple scales, its computational burden is known to be costly, taking about 3.75h to process only two pairs of 6284×2794 images (KIM; HILTON, 2013) on an Intel Xeon 3.0GHz with 32Gb RAM.

In a subsequent study, Kim and Hilton (2015) propose a light-weight alternative to Kim and Hilton (2013), assuming the scenes they are interested in can be modeled as a collection of blocks. The camera setup used in their study is the same as in Kim and Hilton (2013), although the images are now processed in cube map format. From the input images, after segmented into regions, planes are fitted and registered together. A final step is applied to refine the resulting planes and to mount the cuboid representations. A drawback of their initial hypothesis is that every thin structure must be either represented as an additional cuboid or suppressed.

Still in the context of calibrated camera rigs, Won, Ryu and Lim (2019) tackle both the pose and 3D scene reconstruction problems by considering a set of four fisheye lenses spaced by a wide baseline (30-60cm). Intrinsic camera calibration is needed for fisheye imagery and it is done on a offline process in Won, Ryu and Lim (2019). The authors introduce a convolutional neural network (CNN) model that takes warped fisheye images

and outputs a cost volume that is further filtered using a sweeping method, and aggregated by semi-global matching. Their results are compared with another CNN-based method suited for inferring depth from pinhole-based images (ZBONTAR; LECUN, 2016). The authors disregard the north and south poles in their evaluation.

Pathak et al. (2017a) argue that dense optical flow approaches are more efficient and robust against outliers when compared to the ones that rely purely on sparse features for the tasks they are interested, namely spherical-based video stabilization and depth reconstruction. In this sense, in a sequence of works (PATHAK et al., 2015; PATHAK et al., 2016a; PATHAK et al., 2016b; PATHAK et al., 2016; PATHAK et al., 2017a), Pathak and colleagues explore the concept of image derotation allied to standard dense optical flow algorithms (FARNEBÄCK, 2003; WEINZAEPFEL et al., 2013) to provide a basis for solving the two tasks above. Derotation can be understood as the process of (theoretically losslessly) estimating and removing the effect of spherical camera rotation (PATHAK et al., 2017a). Thus, a derotated spherical image differs from a reference image (*e.g.* the first frame in a video sequence) only by translational camera movements, and thus presents a flow pattern similar to the one shown in Figures 2.3b and 2.3c.

Pathak et al. (2016a) propose an optical flow-based derotation method and apply it to stabilize full spherical image sequences. The main motivation exposed in their paper is to avoid users' disorientation when watching videos captured by drones, vehicles or robots equipped with an omnidirectional camera. In practice, as optical flow methods rely on the assumption of small camera movement in between the frames, Pathak et al. (2016a) firstly derotate each frame based solely on the 8-PA random sample consensus (RANSAC) (FISCHLER; BOLLES, 1981) algorithm computed on 200 A-KAZE features per image. The method they introduce, which non-linearly minimizes the moment of the magnitude-normalized flow, is applied as a refinement, running at 5 FPS on an Intel Core i7 with 250×500 images. The results they obtain are compared in terms of absolute error of the Euler angle estimates against the ones given by the 8-PA RANSAC method alone, proving to be better. Preliminary results of Pathak et al. (2016a) can be found in Pathak et al. (2015).

Latter, Pathak et al. (2017a) extend their own previous works (PATHAK et al., 2016a; PATHAK et al., 2015) where instead of estimating only the rotation parameters (3-DoF), they also take into account the translation direction, making the optimization process fully aware of the epipolar constraints (5-DoF). In the paper, they show an increased performance of the new method regarding its simplified version, along with an

undesired augmented runtime. In all the works Pathak et al. (2015), Pathak et al. (2016a), Pathak et al. (2017a), the popular Farneback approach is used to compute the optical flow.

The very same concepts of Pathak et al. (2017a) are applied for estimating 3D geometry in Pathak et al. (2016). Precisely, as before, the 5-DoF are estimated by a non-linear least squares minimization, solved by Levenberg-Marquardt approach (LEVENBERG, 1944), which is initialized with the output of the 8-PA RANSAC computed on A-KAZE matches. This time, the authors applied their method on optical flow estimates from DeepFlow (WEINZAEPFEL et al., 2013). In the context of 3D geometry estimation, there is an additional advantage of using optical flow-based techniques with respect to the ones based on sparse features: the flow field can naturally be converted to dense depth maps. The same methodology is also published in Pathak et al. (2017).

In the previous works (PATHAK et al., 2015; PATHAK et al., 2016a; PATHAK et al., 2016; PATHAK et al., 2017a), the dense optical flow is computed on the equirectangular projection of the spherical images and then projected to the sphere. Contrarily, in Pathak et al. (2016b), the complete optimization process is performed on the equirectangular format because, according to the authors, it avoids issues caused by numerical error. The main pipeline remains the same as in their previous works: firstly the camera pose is estimated via A-KAZE features and RANSAC 8-PA, and the images are rectified. Then, after estimating the flow matches by the DeepFlow algorithm, the target spherical image is iteratively derotated in a non-linear least square framework. The same methodology is applied to generate motion parallax for VR/AR/MR users equipped with head-mounted displays (HMDs) in Pathak et al. (2017b).

In a subsequent study, Pathak et al. (2018) used their previous proposal in Pathak et al. (2016b) to estimate the two-view motion, and added a second part to the pipeline for computing the pose for additional images. The novel part of their method minimizes a weighted photometric error in an iterative process for estimating the 6-DoF pose concerning some reference image. Their analysis is restricted to the pose estimation problem, and they do not consider the 3D reconstruction task.

The studies of Wegner et al. (2018) and Lai et al. (2019) also work with stereoscopic spherical imagery, but tackle the problem in different ways. Wegner et al. (2018) use the cylindrical projection of 360° images instead of fitting them into the spherical model. The authors adapt a patch-based depth estimation reference software maintained by the Motion Picture Experts Group (MPEG) to the circular binocular formulation that they present. Wegner et al. (2018) rely on rotation-free videos, and present their results for

a single proprietary stereoscopic video frame with no quantitative evaluation. On the other hand, Lai et al. (2019) tackled the the omnidirectional depth estimation problem from a learning-based perspective. They present a encoder-decoder CNN model that receives as input a stereo-rectified pair of 256×128 equirectangular images with small baseline (6.5cm) and outputs an estimate for the depth of one of them. The authors use traditional CNN architecture components and present a loss function that encourages associating the left and right boundary information from equirectangular images. They provide results indicating that the presented loss function benefits their regression method. The approaches from both Wegner et al. (2018) and Lai et al. (2019) estimate frame-wise dense depth maps from stereoscopic 360° videos with no temporal coherency.

Some studies focus on the detection, matching, and tracking of sparse features, and, as a final step, they produce a densified version of the depth map. In this sense, Im et al. (2016) adopt the Kanade-Lucas-Tomasi (KLT) algorithm (TOMASI; KANADE, 1991) to track features found by the Harris corner detector (HARRIS; STEPHENS, 1988) on unstitched fish-eye hemispherical captures of narrow-baseline video clips. These features feed their bundle adjustment (BA) method, which considers the intrinsic parameters of the fish-eye views and minimizes the reprojection error on the sphere. Finally, as their major contribution, Im et al. (2016) present a sphere sweeping (SS) method for estimating a dense depth map from the previously matched set of correspondences. After rectifying the captures, their SS algorithm groups the spherical images into "onion-like layers", and searches for the virtual sphere that maximize the photo-consistency for each pixel. The sub-pixel intensities are calculated by bicubic interpolation, and the final depth map is filtered by a non-local cost aggregation method.

Pagani et al. (2011) divide their pipeline into two main phases: (i) a SfM part for recovering the pose and sparse 3D representation of the scene; and (ii) a MVS step for estimating denser representations of the scene. The authors start by extracting and matching ASIFT-like keypoints and then recovering the pose via the 8-PA RANSAC framework. Afterward, they refine the obtained pose by a novel non-linear optimization of the pre-computed Essential matrix. The "Perspective-n-Point" (PnP) problem is solved via DLT, and once more the pose is non-linearly refined. Iteratively, the authors apply a BA step which minimizes the Euclidean distance between the reprojected 3D and the imaged features. Once the sparse point cloud is at hand, Pagani et al. (2011) generate a set of virtual perspective images for given virtual camera locations, which they call anchor-points. They consider 10 to 30 anchor-points. Those (calibrated) images feed the traditional

Patch-based Multiple View Stereo (PMVS) algorithm (FURUKAWA; PONCE, 2007), which outputs semi-dense sets of patches. From those patches, a denser point cloud is generated for each anchor-point. Finally, all the point clouds are merged. An initial study about the non-linear refinements proposed in Pagani et al. (2011) can be found in Pagani and Stricker (2011).

Guan and Smith (2017b) introduce an SfM pipeline, focusing mostly on the pose estimation part of the problem. The authors apply their method to the video stabilization problem, and because of that, they work with sparse features, namely the SSIFT, which are computed only for adjacent frames. Assuming the von Mises-Fisher distribution (WOOD, 1994) to model the noise in spherical matching, the authors contribute in a series of steps of the SfM pipeline. As in Pagani and Stricker (2011), Pagani et al. (2011), the PnP problem is tackled on the spherical domain by Guan and Smith (2017a). The authors rename the problem as "Spherical-n-Point" (SnP) and propose constrained calibrated reconstruction variants of the linear DLT. Their BA framework instead of minimizing the Euclidean or the angular error, maximizes the dot product of the imaged and reprojected 3D points.

Similarly, Huang et al. (2017) present a warping algorithm capable of synthesizing both small-baseline translation and rotation movements in HMDs. Their method estimates the camera motion and the 3D geometry from the scene by sparsely detecting and tracking features via KLT in cube-map represented spherical images. The authors rely on a BA step for refining both the camera pose and sparse 3D point estimates in an incremental setup, where the camera pose of a new view is initialized with the parameters of its predecessor. In a post-processing step, Huang et al. (2017) interpolate a triangulated version of the initial set of 3D points of each frame to generate a dense depth map. Finally, they merge all the depth maps and create a suitable representation for rendering the 3D scene.

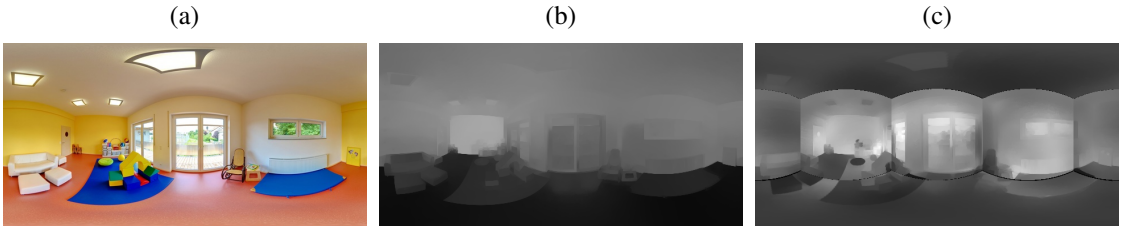
Gava, Stricker and Yokota (2018) introduced a method for estimating a dense depth map from a large set of omnidirectional images with known in-between baseline configuring a spherical light field (SLF) setup. The 3D geometry recovery problem is modeled in a variational framework and is solved using modern energy minimization techniques. The authors explore the main idea behind 3D light field approaches which is to construct a structure containing the cost of matching for each ray starting from the camera center referring to a previously selected reference image. More precisely, they construct and filter a volume cost aiming to estimate the profile of the underlying geometry captured by the light field. Gava, Stricker and Yokota (2018) show compelling results

using both indoor and outdoor scenes.

2.4.2 Spherical Monocular Depth Estimation

In the limit case, it would be desirable to infer depth information from a single spherical image. There are a few approaches for depth estimation from monocular perspective images (LIU et al., 2016; EIGEN; FERGUS, 2015; GODARD; Mac Aodha; BROSTOW, 2017), but their extension for panoramas is not trivial. Su and Grauman (2017) claim that methods developed for planar images (in the context of object detection) can be applied to the spherical domain: (i) directly on the (equirectangular projection of the) sphere, and (ii) to tangent planar projections with lower FoV images, where the camera model distortions are attenuated. However, as can be seen in Figure 2.5, neither the first nor the second strategy works well for the monocular depth estimation problem using a CNN-based method trained for planar images (LIU et al., 2016) (other CNN-based approaches (EIGEN; FERGUS, 2015; GODARD; Mac Aodha; BROSTOW, 2017) tend to produce poor results as well). In the first scenario, shown in Figure 2.5(b), the ceiling and the floor are estimated with completely different distances. If we consider disjoint perspective images extracted from the sphere, as shown in Figure 2.5(c), the projection of the estimated depth maps back to the sphere generate lots of artifacts, especially along the boundaries of the planar projections.

Figure 2.5: (a) Color image. Depth estimates from (b) the entire equirectangular image and (c) from six disjoint sections. Source: the author.



We attempt to solve this problem by considering overlapping regions which are submitted to planar monocular depth estimation algorithms (LIU et al., 2016; GODARD; Mac Aodha; BROSTOW, 2017), and the corresponding depth maps are combined back in the spherical image representation. Although this is not the main focus of this Dissertation, the proposed approach is briefly explained in Chapter 6.

Similarly, Yang, Liu and Kang (2018) proposed to infer depth from a single equirect-

angular image by extracting 18 overlapping 90° FOV sub-views to the respective tangent planes. For each sub-view, geometric (lines, vanishing points, orientation map, and surface normals), salience and object detection information are estimated using algorithms suitable for perspective images. Then, all the information estimated per sub-view is back-projected to the sphere and combined, and is used to segment the scene content into layout (background) and object (foreground). More precisely, layout depth information is initially extracted from superpixel segments under the assumption of Manhattan worlds, which is constrained to the combined geometric information obtained from the sub-views. Then, the objects' depths are initialized by those in boundary regions between the ground plane and the respective object. These initial estimates are propagated to the entire object, assuming that neighbor super-pixels have similar depth values. The depth estimation process proposed in Yang, Liu and Kang (2018) is encapsulated in an optimization framework which takes about 8 minutes to process a 2048×1024 image on an Intel Core i7 with 8GB RAM computer.

In addition to the other approaches, Zioulis et al. (2018) introduced a learning framework to infer depth from a single omnidirectional image in equirectangular format. They test two fully convolutional encoder-decoder network architectures that are structured differently. One of them, which presents the best results, accounts for the distortions of the input images by using dilated convolutions for increasing the effective receptive fields depending on the latitudes. The models they present were trained in a completely supervised manner with ground truth depth. To accomplish this, the authors reused 3D datasets with both synthetic and real-world scanned indoor scenes by synthesizing single texture plus depth images via rendering.

For the sake of completeness, we also mention the existence of methods that, instead of depth, infer a simplified representation of the scene "layout" from a single spherical image, such as Zou et al. (2018) and Yang et al. (2019). These strategies infer the locations on the input equirectangular image taken in indoor environments that may contain 3D corners or joints between three or more planes. In both studies, CNN-based solutions are proposed to recover the layout of complex shaped indoor scenes that fit the Manhattan world hypothesis, as Kim and Hilton (2015), Yang, Liu and Kang (2018).

Table 2.1: Comparison of methods for depth reconstruction from spherical images

Method	Camera setup	Image representation	Matching algorithm	Output
Kim and Hilton (2013)	Stereoscopic MVS	Equirectangular	PDE-based stereo and ICP	Dense mesh
Kim and Hilton (2015)	Stereoscopic MVS	Cube-map	Segmentation, plane fitting and registration	Cuboids
Wegner et al. (2018)	Stereoscopic MVS	Cylindrical projection	-	Dense depth map
Lai et al. (2019)	Stereoscopic MVS	Equirectangular	Convolutional Neural Network	Dense depth map
Won, Ryu and Lim (2019)	2× Stereoscopic MVS	Hemispherical imagery	Convolutional Neural Network	Dense depth map
Pathak et al. (2016) Pathak et al. (2017)	Stereo image pair	Equirectangular	A-KAZE and DeepFlow	Dense point cloud
Pathak et al. (2016b) Pathak et al. (2017b)	Stereo image pair	Equirectangular	A-KAZE and DeepFlow	Dense point cloud
Im et al. (2016)	Monoscopic SfM	Hemispherical pairs	Harris detector, KLT and SS	Dense point cloud
Pagani and Stricker (2011) Pagani et al. (2011)	Monoscopic SfM	Equirectangular	ASIFT and PMVS	Densified point cloud
Huang et al. (2017)	Monoscopic SfM	Cube-map	KLT and NCC-based image-quided filter	Densified point cloud
Guan and Smith (2017a)	Monoscopic SfM	Equirectangular	SSIFT	Sparse point cloud
Our approach from Chapter 4	Monoscopic SfM	Equirectangular	SPHORB and DeepFlow	Dense depth map
Our approach from Chapter 5	Monoscopic SfM	Equirectangular	SPHORB, SSNIC and DeepFlow	Dense depth map
Gava, Stricker and Yokota (2018)	Monoscopic SLF	Equirectangular	Variational approach	Dense point cloud
Yang, Liu and Kang (2018)	Single image	Tangent planes	Geometric, saliency and object detection information	Dense point cloud
Zioulis et al. (2018)	Single image	Equirectangular	Convolutional Neural Network	Dense depth map
Our approach from Chapter 6	Single image	Tangent planes	Convolutional Neural Network	Dense depth map

2.5 Conclusions of the Chapter

We have presented some of the basic concepts about spherical imaging, highlighting the epipolar geometry that encodes the pose constraints relating two image points in correspondence. Also, we described the linear 8-PA and listed some works that invested efforts on how to measure the impact of noisy matchings on the estimated Epipolar matrix. Then, we presented alternative algorithms for extracting, describing and matching relevant points in spherical image pairs, using both sparse or dense approaches. Finally, we focused on briefly describing some related methods for estimating the relative (5-DoF or 6-DoF) pose from two or more images, as well as depth information from multiple images and, in the most extreme case, from a single image.

A summary of the methods addressing the 3D reconstruction problem presented in this section is shown in Table 2.1. Ideally, the user should be free for using a single camera in a non-calibrated setup without depending on proprietary image representation. Moreover, dense depth maps or point clouds tend to be naturally interesting for most of applications. One may note the several differences among the explained methods, starting from the input, until the image representation, base algorithms, and the output. We complement the table presenting our three novel *dense* 3D reconstruction methods, which will be described in Chapters 4, 5, and 6. Before introducing those methods, we firstly present, in Chapter 3, a deeper analysis about the influence of noisy matches on the Essential matrix estimation based on the 8-PA, going in the same direction of the works revised in Section 2.2.3.

3 PERTURBATION ANALYSIS FOR THE EIGHT-POINT ALGORITHM

This chapter presents a perturbation analysis for the estimate of Epipolar (Fundamental/Essential) matrices using the eight-point algorithm (8-PA). Our approach explores existing bounds for singular subspaces and relates them to the 8-PA, without assuming any error distribution for the matched features. In particular, if we use unit vectors as homogeneous image coordinates as in Equation (2.5), we show that having a wide spatial distribution of matched features in both views tends to generate lower error bounds for the Epipolar matrix error. Our experimental validation indicates that the bounds and the effective errors tend to decrease as the camera FoV increases and that using the 8-PA for spherical images (that present full $360^\circ \times 180^\circ$ FoV) leads to accurate Essential matrices. Additionally, we present bounds for the direction of the translation vector extracted from the Essential matrix based on singular subspace analysis.

3.1 Bounds for Singular Subspaces

Perturbation bounds aim to quantify how the spectrum changes after adding a small perturbation to a matrix, and they play an important role in SVD and spectral methods analysis (CAI; ZHANG et al., 2018). Given an approximately rank-r matrix $\mathbf{M}$ and a perturbation matrix $\mathbf{P}$, both of dimension $n \times m$, an important problem is to understand how much the (left and/or right) singular spaces of $\mathbf{M}$ and $\tilde{\mathbf{M}} = \mathbf{M} + \mathbf{P}$ differ from each other (CAI; ZHANG et al., 2018). Consider that the SVD of matrix $\mathbf{M}$ is given by

$$\mathbf{M} = \begin{bmatrix} \mathbf{U} & \mathbf{U}_\perp \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma}_1 & 0 \\ 0 & \mathbf{\Sigma}_2 \end{bmatrix} \begin{bmatrix} \mathbf{V} & \mathbf{V}_\perp \end{bmatrix}^\top, \tag{3.1}$$

where subscript $\perp$ denotes the orthogonal complement of a subspace. Not that $\begin{bmatrix} \mathbf{U} & \mathbf{U}_\perp \end{bmatrix}$ and $\begin{bmatrix} \mathbf{V} & \mathbf{V}_\perp \end{bmatrix}$ are orthogonal matrices of orders n and m, respectively, $\mathbf{\Sigma}_1 = \mathrm{diag}(\sigma_1, \ldots, \sigma_r)$ and $\mathbf{\Sigma}_2 = \mathrm{diag}(\sigma_{r+1}, \ldots)$ are $r \times r$ and $(n-r) \times (m-r)$ matrices with null off-diagonal values, respectively, and $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$ are the singular values of $\mathbf{M}$ in descending order. Decomposing the perturbed matrix $\tilde{\mathbf{M}}$ as

$$\tilde{\mathbf{M}} = \mathbf{M} + \mathbf{P} = \begin{bmatrix} \tilde{\mathbf{U}} & \tilde{\mathbf{U}}_\perp \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{\Sigma}}_1 & 0 \\ 0 & \tilde{\mathbf{\Sigma}}_2 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{V}} & \tilde{\mathbf{V}}_\perp \end{bmatrix}^\top \tag{3.2}$$

produces matrices having the same structures as $\mathbf{U}, \mathbf{U}_\perp, \mathbf{V}, \mathbf{V}_\perp, \mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$.

A well known bound for estimating the perturbation influence within the singular subspaces comes from Wedin's $\sin\Theta$ theorem (WEDIN, 1972), which provides a uniform bound for both the left and right singular spaces in terms of the singular value gap and perturbation level. Precisely, it states that if the gap $\delta = \min(\tilde{\mathbf{\Sigma}}_1) - \max(\mathbf{\Sigma}_2) > 0$, then:

$$\max\left\{\left\|\sin\Theta(\mathbf{V},\tilde{\mathbf{V}})\right\|_2, \left\|\sin\Theta(\mathbf{U},\tilde{\mathbf{U}})\right\|_2\right\} \leq \frac{\max\left\{\left\|\mathbf{P}\tilde{\mathbf{V}}\right\|_2, \left\|\tilde{\mathbf{U}}^\top\mathbf{P}\right\|_2\right\}}{\delta} \leq \frac{\|\mathbf{P}\|_2}{\delta}, \tag{3.3}$$

where $\|\cdot\|_2$ is the spectral norm of a matrix (CAI; ZHANG et al., 2018), and $\Theta(\mathbf{M}_1, \mathbf{M}_2)$ are the canonical angles between matrices $\mathbf{M}_1$ and $\mathbf{M}_2$ (STEWART, 1990). Despite of the wide application range, Wedin's theorem may not be sufficiently precise for some analysis where left and right singular spaces change in different orders of magnitude after the perturbation.

Many works present tighter perturbation bounds, but they are applicable only to problems with known noise properties (O'ROURKE; VU; WANG, 2013; WANG, 2015). On the other hand, Cai, Zhang et al. (2018) established rate-optimal perturbation bounds for the left and right singular spaces separately without any noise assumption. In short, these bounds are given by:

$$\left\|\sin\Theta(\mathbf{U},\tilde{\mathbf{U}})\right\| \leq \min\left(\frac{\xi z_{21} + \zeta z_{12}}{\xi^2 - \zeta^2 - \varsigma}, 1\right) \tag{3.4}$$

and

$$\left\|\sin\Theta(\mathbf{V},\tilde{\mathbf{V}})\right\| \leq \min\left(\frac{\xi z_{12} + \zeta z_{21}}{\xi^2 - \zeta^2 - \varsigma}, 1\right), \tag{3.5}$$

provided that $\xi^2 > \zeta^2 + \varsigma$, where $\xi = \sigma_{\min}(\mathbf{U}^\top\tilde{\mathbf{M}}\mathbf{V})$, $\zeta = \|\mathbf{U}_\perp^\top\tilde{\mathbf{M}}\mathbf{V}_\perp\|$, $\varsigma = \min(z_{21}^2, z_{12}^2)$, $z_{12} = \|\mathbb{P}_{\mathbf{U}}\mathbf{P}\mathbb{P}_{\mathbf{V}_\perp}\|$ and $z_{21} = \|\mathbb{P}_{\mathbf{U}_\perp}\mathbf{P}\mathbb{P}_{\mathbf{V}}\|$. Here, $\mathbb{P}_{\mathbf{D}}$ is projection operator onto the column space of a matrix $\mathbf{D}$ (CAI; ZHANG et al., 2018), and $\|\cdot\|$ is either the spectral ($\|\cdot\|_2$) or the Frobenius ($\|\cdot\|_F$) matrix norm.

The bounds provided in Equations (3.4) and (3.5) tackle the left and right subspaces separately and are tighter then Wedin's bound (CAI; ZHANG et al., 2018). However, they involve projections of the perturbation matrix onto the noiseless right and left singular subspaces, which are not known in practical applications.

3.2 Perturbation Bounds and the 8-PA

In this section, we relate generic bounds for singular subspaces and the 8-PA. Ideally, the measurement matrix $\mathbf{A}_{n\times 9}$ used in the 8-PA (recall Equation (2.5)) presents rank $r = 8$. Using the notation of Equation (3.1), its SVD generates an 8×8 diagonal matrix $\mathbf{\Sigma}_1$ containing all the non-zero singular values of $\mathbf{A}$, with the corresponding left and right singular vectors provided in $\mathbf{U}$ and $\mathbf{V}$, respectively. Also, $\mathbf{\Sigma}_2$ should be an $(n-8) \times 1$ null matrix, and in particular $\mathbf{e} = \mathbf{V}_\perp$ is the least right singular vector that contains the elements of the Epipolar matrix.

In practice, feature matching is not exact. Without loss of generality, as done in Mühlich and Mester (1998), let us assume that $\mathbf{x}_1^i$ corresponds to the exact feature points in the first image and $\tilde{\mathbf{x}}_2^i$ to the noisy correspondences in the second image, leading to an approximate matrix $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{P}$, where $\mathbf{P} = [p_{ij}]_{n\times 9}$ is the perturbation. Due to matching errors, there is no guarantee that $\tilde{\mathbf{A}}$ presents rank 8, so that $\tilde{\mathbf{\Sigma}}_2$ may not be null.

Our goal here is to estimate the error between the actual Essential matrix $\mathbf{e} = \mathbf{V}_\perp$ and the estimated one $\tilde{\mathbf{e}} = \tilde{\mathbf{V}}_\perp$, both expressed in vector form. A natural distance measure is the angular distance between them, computed as

$$\eta = \angle(\mathbf{e}, \tilde{\mathbf{e}}) = \cos^{-1} \left| \mathbf{e}^\top \tilde{\mathbf{e}} \right|, \tag{3.6}$$

which is a particular case of the canonical angles (STEWART, 1990).

Furthermore, as analyzed in Drmac (2000), the canonical angles relate to projection errors. More precisely, if $\mathbb{P}_{\mathbf{V}} = \mathbf{V}\mathbf{V}^\top$ and $\mathbb{P}_{\tilde{\mathbf{V}}} = \tilde{\mathbf{V}}\tilde{\mathbf{V}}^\top$ are the orthogonal projection matrices onto the subspaces spanned by the columns of $\mathbf{V}$ and $\tilde{\mathbf{V}}$, respectively, then

$$\|\mathbb{P}_{\mathbf{V}} - \mathbb{P}_{\tilde{\mathbf{V}}}\|_2 = \left\| \sin\Theta(\mathbf{V}, \tilde{\mathbf{V}}) \right\|_2. \tag{3.7}$$

Also, since $\mathbf{e}$ and $\tilde{\mathbf{e}}$ are the orthogonal complements of $\mathbf{V}$ and $\tilde{\mathbf{V}}$, respectively, then

$$\begin{aligned} |\sin\eta| &= \|\mathbb{P}_{\mathbf{e}} - \mathbb{P}_{\tilde{\mathbf{e}}}\|_2 = \|(\mathbf{I} - \mathbb{P}_{\mathbf{V}}) - (\mathbf{I} - \mathbb{P}_{\tilde{\mathbf{V}}})\|_2 \\ &= \|\mathbb{P}_{\mathbf{V}} - \mathbb{P}_{\tilde{\mathbf{V}}}\|_2 = \left\| \sin\Theta(\mathbf{V}, \tilde{\mathbf{V}}) \right\|_2, \end{aligned} \tag{3.8}$$

recalling that η is the angle between $\mathbf{e}$ and $\tilde{\mathbf{e}}$. Combining Equation (3.8) with Wedin's

bound provided by Equation (3.3), we can conclude that

$$|\sin\eta| \leq \frac{\|\mathbf{P}\|_2}{\delta} \leq \frac{\|\mathbf{P}\|_F}{\delta}, \tag{3.9}$$

meaning that the error bound in the estimate of the Essential matrix is proportional to the norm of the perturbation $\mathbf{P}$ and inversely scaled by the second least singular value of $\tilde{\mathbf{A}}$. Although the spectral norm provides a tighter bound, the Frobenius norm will be used, since it can be expressed only in terms of the matching errors. In fact, we can express the perturbation matrix $\mathbf{P} = \tilde{\mathbf{A}} - \mathbf{A}$ as a function of the matched points $\mathbf{x}_1^i$, $\mathbf{x}_2^i$ and $\tilde{\mathbf{x}}_2^i$. Based on Equation (2.5), the i-th line of $\mathbf{P}$ is given by

$$\mathbf{P}_i = \begin{bmatrix} x_1^i (\Delta\mathbf{x}_2^i)^\top & y_1^i (\Delta\mathbf{x}_2^i)^\top & z_1^i (\Delta\mathbf{x}_2^i)^\top \end{bmatrix} = \begin{bmatrix} x_1^i(\tilde{x}_2^i - x_2^i) \\ x_1^i(\tilde{y}_2^i - y_2^i) \\ x_1^i(\tilde{z}_2^i - z_2^i) \\ y_1^i(\tilde{x}_2^i - x_2^i) \\ y_1^i(\tilde{y}_2^i - y_2^i) \\ y_1^i(\tilde{z}_2^i - z_2^i) \\ z_1^i(\tilde{x}_2^i - x_2^i) \\ z_1^i(\tilde{y}_2^i - x_2^i) \\ z_1^i(\tilde{z}_2^i - z_2^i) \end{bmatrix}^\top, \tag{3.10}$$

where $\Delta\mathbf{x}_2^i = \tilde{\mathbf{x}}_2^i - \mathbf{x}_2^i = [(\tilde{x}_2^i - x_2^i)\ (\tilde{y}_2^i - y_2^i)\ (\tilde{z}_2^i - z_2^i)]^\top$. Hence,

$$\|\mathbf{P}\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{9} |p_{ij}|^2} = \sqrt{\sum_{i=1}^{n} \|\Delta\mathbf{x}_2^i\|^2 \|\mathbf{x}_1^i\|^2} = \sqrt{2\sum_{i=1}^{n}(1 - \cos\alpha_j)}, \tag{3.11}$$

where $\alpha_j = \angle(\tilde{\mathbf{x}}_2^i, \mathbf{x}_2^i)$ is the angular matching error. Consequently, the total perturbation $\|\mathbf{P}\|_F$ does not depend on the choice of the feature points $\mathbf{x}_1^i$ on the first image, but solely on the matching errors on the second image.

Furthermore, a typical error measure when estimating Fundamental/Essential matrices is based on the relative error using the Frobenius norm. If $\mathbf{E}$ and $\tilde{\mathbf{E}}$ denote the matrix forms of the essential matrices related to $\mathbf{e}$ and $\tilde{\mathbf{e}}$, respectively, then $d(\mathbf{E}, \tilde{\mathbf{E}}) = \min\{\|\mathbf{E} - \tilde{\mathbf{E}}\|_F, \|\mathbf{E} + \tilde{\mathbf{E}}\|_F\}$ can be used to measure the error of the estimated matrix. Hence,

$$d(\mathbf{E}, \tilde{\mathbf{E}}) = \min\{\|\mathbf{e} - \tilde{\mathbf{e}}\|_2, \|\mathbf{e} + \tilde{\mathbf{e}}\|_2\} = \sqrt{2\,(1 - \cos\eta')}, \tag{3.12}$$

where $\eta' = \min\{\eta, \pi - \eta\} = \sin^{-1} |\sin \eta| \in [0, \pi/2]$.

Since Equation (3.9) provides bounds for $|\sin \eta|$, we have

$$d(\mathbf{E}, \tilde{\mathbf{E}}) \leq \sqrt{2 \left[1 - \cos\left(\sin^{-1} \min\left\{1, \frac{\|\mathbf{P}\|}{\delta}\right\}\right)\right]}. \tag{3.13}$$

3.3 Relationship Between δ and the Spread of the Features

One interesting aspect of the bound presented in Inequality (3.3) is that the denominator δ is fully computable based on the observed matrix $\tilde{\mathbf{A}}$, without any knowledge on the noiseless matrix $\mathbf{A}$. More precisely, $\delta = \tilde{\sigma}_8$ is the second least singular value of $\tilde{\mathbf{A}}$, which depends on several aspects: the 3D structure of the scene (noting that points along a single plane lead to degeneracy), the locations of selected keypoints, the relative camera poses and the FoV of the cameras. In particular, some authors (HADFIELD; LEBEDA; BOWDEN, 2 018; YANG; LI; JIA, 2014; ZHANG et al., 2016) have empirically studied the effect of the camera FoV. Here, we provide a more formal relationship of the gap δ with the spatial distribution of the features, which is highly related to the camera FoV.

Let us consider the singular values of $\tilde{\mathbf{A}}$ given by $\tilde{\sigma}_1 \geq \tilde{\sigma}_2 \geq \cdots \geq \tilde{\sigma}_9$, where $\tilde{\sigma}_i = \sqrt{\tilde{\lambda}_i}$ and $\tilde{\lambda}_i$ is one of the first nine eigenvalues of $\tilde{\mathbf{A}}^\top \tilde{\mathbf{A}}$ (or, equivalently, of $\tilde{\mathbf{A}} \tilde{\mathbf{A}}^\top$). Hartley (1997) showed that when using un-normalized homogeneous coordinates, the entries along the diagonal of $\tilde{\mathbf{A}}^\top \tilde{\mathbf{A}}$ vary considerably in magnitude and used interlacing properties to find estimates on the eigenvalues and condition number of the matrix. When using unit vectors, however, such analysis is not possible. Instead, we evaluate the impact of the spatial distribution of the matched features, which is strongly affected by the FoV of the cameras.

Merikoski, Sarria and Tarazaga (1994) presented several bounds for singular values and eigenvalues based on traces. In particular, they showed that for a square matrix $\mathbf{B}_{p \times p}$ with real non-negative eigenvalues (in decreasing order), the second least eigenvalue satisfies the following condition:

$$\lambda_{p-1}(\mathbf{B}) \leq \frac{\text{trace}(\mathbf{B})}{p-1} - \sqrt{\frac{1}{(p-1)(p-2)} \left(\text{trace}(\mathbf{B}^2) - \frac{\text{trace}(\mathbf{B})^2}{p-1}\right)}. \tag{3.14}$$

If we consider $\mathbf{B} = \tilde{\mathbf{A}}^\top \tilde{\mathbf{A}}$ (so that $p = 9$), we have that

$$\text{trace}(\mathbf{B}) = \|\tilde{\mathbf{A}}\|_F^2 = \sum_{i=1}^{n} \|\mathbf{x}_1^i\|^2 \|\tilde{\mathbf{x}}_2^i\|^2 = n, \tag{3.15}$$

recalling that n is the number of matched points. Since $\mathbf{B}$ is symmetric, we also have that $\text{trace}(\mathbf{B}^2) = \text{trace}(\mathbf{B}^\top \mathbf{B}) = \|\mathbf{B}\|_F^2$. Let us also consider $\mathbf{C} = \tilde{\mathbf{A}}\tilde{\mathbf{A}}^\top$, so that $\text{trace}(\mathbf{B}^\top \mathbf{B}) = \text{trace}(\mathbf{C}^\top \mathbf{C}) = \|\mathbf{C}\|_F^2$. Matrix $\mathbf{C} = [c_{ij}]_{n \times n}$ presents an interesting structure, since each element is given as a dot product of rows from $\tilde{\mathbf{A}}$:

$$\begin{aligned} c_{ij} = \tilde{\mathbf{A}}_i \tilde{\mathbf{A}}_j^\top = x_1^i x_1^j (\tilde{\mathbf{x}}_2^i)^\top \tilde{\mathbf{x}}_2^j + y_1^i y_1^j (\tilde{\mathbf{x}}_2^i)^\top \tilde{\mathbf{x}}_2^j + z_1^i z_1^j (\tilde{\mathbf{x}}_2^i)^\top \tilde{\mathbf{x}}_2^j = \\ = (\mathbf{x}_1^i)^\top \mathbf{x}_1^j (\tilde{\mathbf{x}}_2^i)^\top \tilde{\mathbf{x}}_2^j = (\cos \beta_{ij})(\cos \gamma_{ij}), \end{aligned} \tag{3.16}$$

where $\beta_{ij} = \angle(\mathbf{x}_1^i, \mathbf{x}_1^j)$ and $\gamma_{ij} = \angle(\tilde{\mathbf{x}}_2^i, \tilde{\mathbf{x}}_2^j)$ are the angles between features i and j in the first and second images, respectively. Clearly, both β_{ij} and γ_{ij} are limited by the FoV of the camera: if it is small, the entries c_{ij} tend to be closer to one. Also, we have that

$$\|\mathbf{C}\|_F^2 = \sum_i \sum_j c_{ij}^2 = \sum_i \sum_j (\cos^2 \beta_{ij})(\cos^2 \gamma_{ij}). \tag{3.17}$$

Recalling that $\tilde{\sigma}_8 = \sqrt{\tilde{\lambda}_8}$, we simplify Equation (3.14) to obtain

$$\tilde{\sigma}_8 \leq \sqrt{\frac{n}{8} - \frac{1}{8}\sqrt{\frac{8\|\mathbf{C}\|_F^2 - n^2}{7}}}. \tag{3.18}$$

If all the angles β_{ij} and γ_{ij} are all small, $\|\mathbf{C}\|_F$ tends to be larger, yielding a smaller value for $\delta = \tilde{\sigma}_8$ and hence more potential sensibility to perturbations. In the limit, we have $\|\mathbf{C}\|_F \approx n$, which leads to $\delta \approx 0$. In this case, even small perturbations $\mathbf{P}$ can lead to highly degraded estimates for the epipolar matrix. On the other hand, the bound in Inequality (3.18) is at most $\sqrt{n}/8$, which is an "optimistic" upper bound for δ (best case scenario), leading to $\delta = \mathcal{O}(\sqrt{n})$. Also, note that a single outlier can significantly increase the perturbation $\|\mathbf{P}\|_F$ according to Equation (3.11) so that n must be very large to compensate for the presence of "bad" outliers.

Our analysis can be easily extended to weighted versions of the 8-PA, which is used in iterative reweighted least-squares (IRLS) schemes (TORR; MURRAY, 1997) or in the loss function of recent deep learning approaches (YI et al., 2018). This involves defining an $n \times n$ diagonal matrix $\mathbf{W} = [w_{ij}]$ with the weights for each correspon-

dence pair and minimizing $\|\mathbf{WAe}\|^2$. In that case, the perturbation error is $\|\mathbf{WP}\|_F = \sqrt{\sum_{i=1}^{n} w_{ii}^2 \|\tilde{\mathbf{x}}_2^i - \mathbf{x}_2^i\|^2}$, so that the influence of outliers can be alleviated by choosing small weights for bad matches.

3.4 Perturbation Analysis of Pose Estimation from the Essential Matrix

For calibrated or spherical cameras, the 5-DoF pose parameters – rotation and (scaleless) translation vector – can be extracted from the Essential matrix through the SVD (HARTLEY; ZISSERMAN, 2003), as discussed in Section 2.2.2. In particular, the direction of the translation vector is given by the least left singular vector $\mathbf{t}$ of $\mathbf{E}$, and it is more prone to errors in the Essential matrix than the rotation matrix, as noted in (NISTÉR, 2004; TIAN; TOMASI; HEEGER, 1996).

Let us consider a true Essential matrix $\mathbf{E}$ with $\|\mathbf{E}\|_F = 1$, and let $\mathbf{t}$ and $\tilde{\mathbf{t}}$ denote the least left singular values of $\mathbf{E}$ and its estimate $\tilde{\mathbf{E}}$, respectively, and assume that the direction ambiguity was solved (*e.g.* by using chirality constraints (SZELISKI, 2010)). Using the notation of Equations (3.1) and (3.2), the SVD of $\mathbf{E}$ generates a 2×2 diagonal matrix $\mathbf{\Sigma}_1 = \frac{1}{\sqrt{2}}\mathbf{I}$ containing the two equal singular values of $\mathbf{E}$, and an 1×1 null matrix $\mathbf{\Sigma}_2$ (consider an analogous notation for the SVD of $\tilde{\mathbf{E}}$). The least left singular vectors of $\mathbf{E}$ and $\tilde{\mathbf{E}}$ are given by $\mathbf{t} = \mathbf{U}_\perp$ and $\tilde{\mathbf{t}} = \tilde{\mathbf{U}}_\perp$, respectively. Note that the gap between $\tilde{\mathbf{\Sigma}}_1$ and $\mathbf{\Sigma}_2$ is given by $\delta_{\mathbf{E}} = \min\{\tilde{\mathbf{\Sigma}}_1\} = \sigma_2(\tilde{\mathbf{E}})$, so that Wedin's theorem gives

$$|\sin\omega| \leq \frac{1}{\sigma_2(\tilde{\mathbf{E}})}\|\mathbf{E} - \tilde{\mathbf{E}}\|_F = \frac{1}{\sigma_2(\tilde{\mathbf{E}})}\|\mathbf{e} - \tilde{\mathbf{e}}\|, \tag{3.19}$$

where $\omega = \angle(\tilde{\mathbf{t}}, \mathbf{t})$ is the angle between the actual and the estimated translation values, and $\sigma_2(\tilde{\mathbf{E}})$ is the second least singular value of $\tilde{\mathbf{E}}$. For small perturbations, we expect $\sigma_2(\tilde{\mathbf{E}}) \approx \sigma_2(\mathbf{E}) = 1/\sqrt{2}$. More precisely, Weyl's bound (WEYL, 1912) relates the q-th pair of singular values $\sigma_q(\tilde{\mathbf{E}})$ and $\sigma_q(\mathbf{E})$ through

$$\left|\sigma_q(\tilde{\mathbf{E}}) - \sigma_q(\mathbf{E})\right| \leq \|\mathbf{E} - \tilde{\mathbf{E}}\|_2 \leq \|\mathbf{E} - \tilde{\mathbf{E}}\|_F, \tag{3.20}$$

so that a looser version of the bound in Equation (3.19) can be expressed solely based on the difference between $\mathbf{e}$ and $\tilde{\mathbf{e}}$:

$$|\sin\omega| \leq \frac{\sqrt{2}\|\mathbf{e} - \tilde{\mathbf{e}}\|}{1 - \sqrt{2}\|\mathbf{e} - \tilde{\mathbf{e}}\|}. \tag{3.21}$$

3.5 Experimental Results on the Perturbation Analysis of the 8-PA

In this section, we present the results regarding the perturbation analysis of the 8-PA depending on the noise level and spreading of the features throughout the imaged scene. The results in Section 3.5.1 are related to fully synthetic experiments, whereas in Section 3.5.2 we include the uncertainty of real feature matching in our analysis.

3.5.1 Synthetic Feature Matching

In our experimental setup, we first present results using a set of synthetic 3D points projected to calibrated cameras with known parameters, and add artificial noise to the feature locations on the second view. Since the feature are 3D unit vectors (*i.e.,* on the unit sphere), we add von Mises-Fisher (vMF) noise, as done in Guan and Smith (2017b). For analyzing the uncertainty of the 8-PA concerning the spreading of the features, we consider the FoVs of typical perspective cameras with $54.4^\circ \times 37.8^\circ$ and $65.5^\circ \times 46.4^\circ$ (OSBORNE; GEORGIEV; GOMA, 2014), a 195° fisheye wide-angle camera (LO; SHIH; CHEN, 2018) and a full-spherical camera (AKIHIKO; ATSUSHI; OHNISHI, 2005).

Noise is controlled by parameter κ in the vMF distribution. Here, we select $\kappa \in \{500; 1,000; 2,000; 10,000\}$ corresponding to average angular matching errors equivalent to 3.21°, 2.27°, 1.60° and 0.72°, respectively. For the sake of illustration, Figure 3.1 shows different noise levels and how it reflects on points lying on the unit sphere. For each combination of FoV and noise level, we generate 1,000 experiments, each one containing 100 3D points randomly selected within a 5-10m radius (constrained to the camera FoV), simulating a large indoor environment. The first camera is placed on and aligned to the origin of the world coordinate system, whereas the second camera was randomly placed within a $[-1, 1]^3$ cube with arbitrary rotation. For the sake of illustration, the spherical projection of one set of features using the four selected FoVs is depicted in Figure 3.2.

Table 3.1 presents the average sine error between $\mathbf{e}$ and $\tilde{\mathbf{e}}$ for each combination, as well as the Wedin's bound (Equation (3.9)) and Cai and Zhangs' bound (Equation (3.5))[1]. Both bounds decrease as the noise level decreases and the FoV increases, as expected. However, on average they showed to be quite loose bounds when compared to the actual errors. Cai and Zhangs' bound tends to be tighter than the Wedin's, but it is important to

[1]Since these bounds produce trivial values (≥ 1) for narrow FoVs, we only show the results for wider FoVs.

Figure 3.1: Unit vectors represent the selected mean directions, whilst the spread colored points are the samples from the vMF distribution with different concentration parameters κ. Red, green, blue, yellow, cyan and magenta data are associated, respectively, to $\kappa = 10^1, 10^2 \ldots 10^6$, presenting average angular errors around 22.982°, 7.183°, 2.272°, 0.717°, 0.226°, 0.072°. Source: the author.

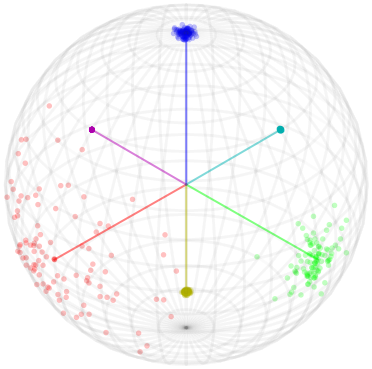
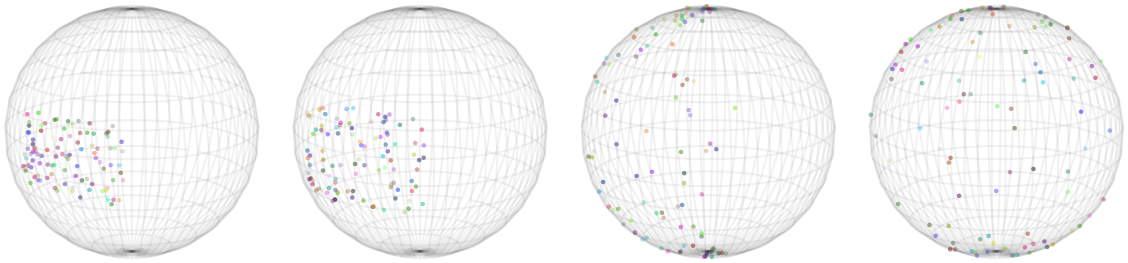


Figure 3.2: Unitary feature projections in different FoVs. Source: the author.

(a) $54.4° \times 37.8°$ (b) $65.5° \times 46.4°$ (c) $195° \times 195°$ (d) $360° \times 180°$



recall that it is not computable on practical applications (requires the projections of the matching errors onto singular subspaces), and thus we will focus only on Wedin's bound hereafter in our analysis.

Table 3.1: Impact of the variation in FoV and noise level when computing the perturbation levels.

κ	54.4°×37.8°	65.5°×46.4°	195°×195°	360°×180°	195°×195°	360°×180°	195°×195°	360°×180°
	Sine error				Wedin's bound		Cai and Zhangs' bound	
500	0.782 ± 0.230	0.778 ± 0.226	0.340 ± 0.252	0.085 ± 0.045	0.868±0.088	0.607±0.076	0.891±0.073	0.671±0.148
1,000	0.781 ± 0.221	0.756 ± 0.241	0.190 ± 0.180	0.054 ± 0.027	0.769±0.136	0.446±0.054	0.705±0.174	0.335±0.082
2,000	0.780 ± 0.223	0.756 ± 0.234	0.103 ± 0.090	0.036 ± 0.017	0.666±0.165	0.326±0.039	0.543±0.205	0.177±0.036
10,000	0.679 ± 0.248	0.563 ± 0.253	0.033 ± 0.025	0.015 ± 0.007	0.356±0.117	0.149±0.017	0.223±0.145	0.053±0.011

Although Wedin's bound showed to be loose, it still provides useful insights into the Essential matrix accuracy as a function of the camera FoVs. Figure 3.3 illustrates the averaged results for the singular gap δ, the sine error and Wedin's perturbation bound by using an extensive combination of the horizontal and vertical FoVs (abbreviated as HFoV and VFoV, respectively). A total of 100 simulations per HFoV×VFoV was performed using the same setup explained before, and Wedin's bound was truncated in value 1. Also, we vary the noise levels, setting $\kappa = 500$, $\kappa = 10,000$ and $\kappa = 1,000,000$ which cor-

respond to an average angular error of $3.21°$, $0.72°$ and $0.07°$, respectively. The selected range for κ encompasses the tolerances of $0.5625°$ and $2°$ for considering corresponding points as "exact" matches as argued in Zhao et al. (2014) and Guan and Smith (2017a), respectively.

Figure 3.3: Average results for the delta value, the sine error and the Wedin's bound (in the rows) for different noise levels (in the columns) and FoVs. From the left to the right, $\kappa = 500$, $\kappa = 10,000$ and $\kappa = 1,000,000$. Source: the author.

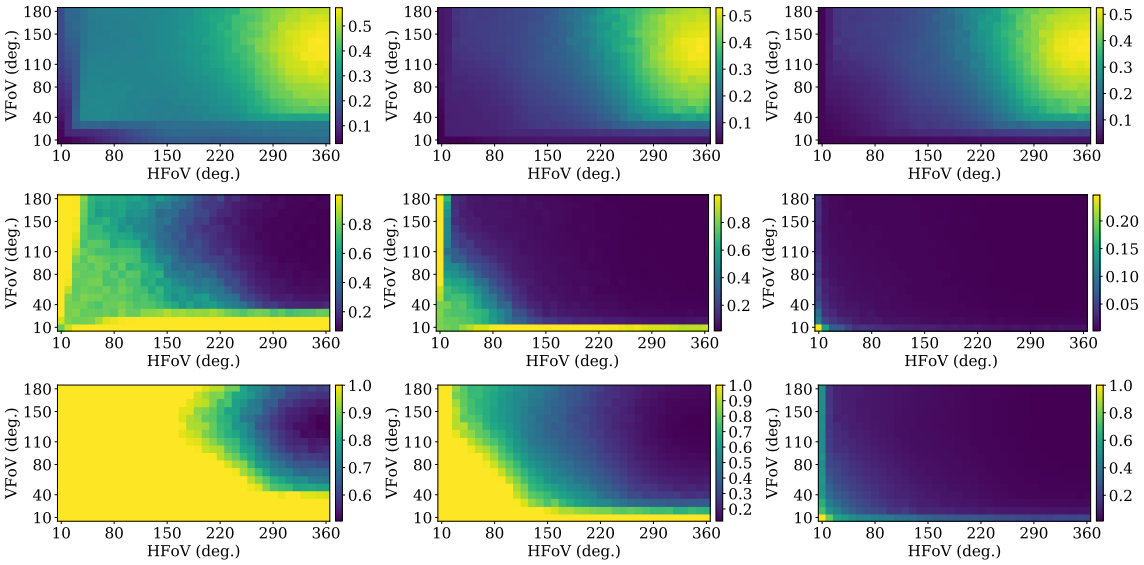
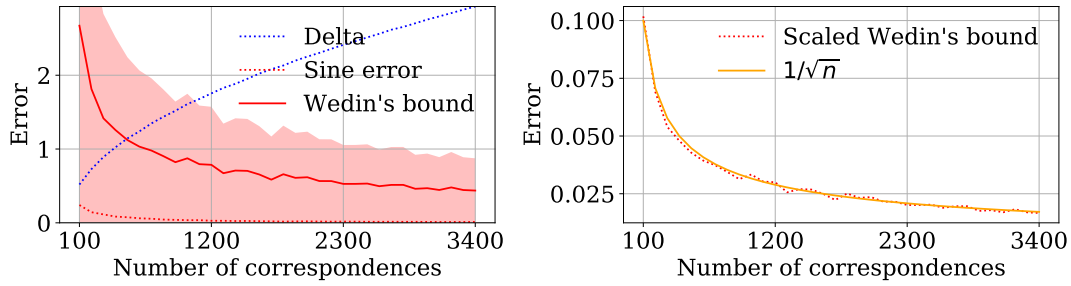


Figure 3.3 shows that the maximum δ occurs around the full 360 degrees FoV (first row), presenting a practically stable value regardless of the tested noise level. In fact, the variance on the full FoV for all noise levels is around 2.8×10^{-5}. Interestingly, the "optimal" VFoV is slightly below $180°$. We believe that this happens because using the full VFoV leads to a circular domain, which might increase the number of neighboring features. Moreover, as expected, when the noise level increases the sine error also increases (second row), and Wedin's bound behaves similarly (third row). It is also possible to see that for higher noise levels combined with narrower FoVs, Wedin's bound turns to be useless because its value is even greater than the trivial bound 1. Last but not least, we found in our experiments that the Spearman's correlation (SPEARMAN, 1904) between the the gap δ and the *diagonal* FoV (DFoV) is around 0.775, 0.863 and 0.877 for $\kappa = \{500; 10,000; 1,000,000\}$, respectively (p-value $\ll 0.01$), indicating a strong relationship.

To illustrate the impact of outliers, we corrupted a set of $n \in [100; 3,400]$ noisy matchings ($\kappa = 16,250$) with a single outlier, and evaluated its effect on the estimation of the Essential matrix. Figure 3.4(a) shows the actual sine error, Wedin's bound (the solid

Figure 3.4: The impact of a single outlier on the error estimates when using full FoV.



red curve shows the mean, and the shaded red area shows the points within one standard deviation) and the gap δ. Although Wedin's bound showed to be loose, both the bound and the actual error seem to decay proportionally to $\mathcal{O}(1/\sqrt{n})$, which is corroborated by Pearson's correlation values $\rho = 0.9932$ and $\rho = 0.9973$, respectively (p-value $\ll 0.01$). For the sake of illustration, we show in Figure 3.4(b) that the (scaled) Wedin's bound roughly matches function $1/\sqrt{n}$. These results corroborate our findings in Section 3.3.

The second part of the analysis consists of estimating the accuracy of the 5-DoF pose extracted from the Essential matrix (here, we assume the calibrated/spherical case). Our evaluation metric for the 2-DoF translation vector is the angular error (YANG; LI; JIA, 2014) given by

$$\varepsilon_{\angle \mathbf{t}} = \cos^{-1}(\mathbf{t}^\top \tilde{\mathbf{t}}). \tag{3.22}$$

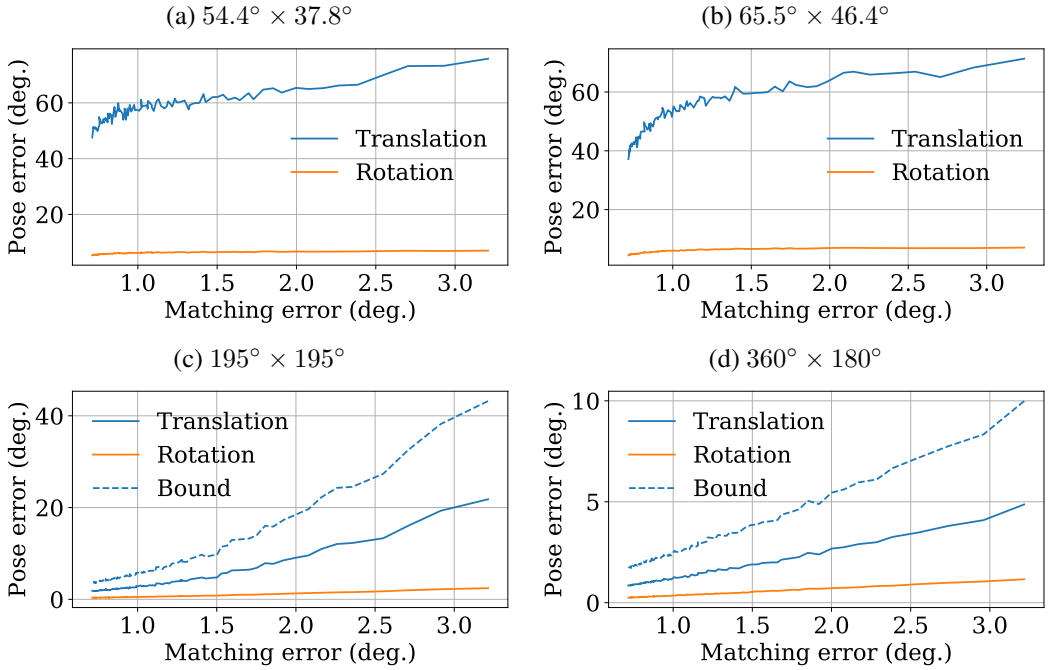
For the sake of illustration, we also show the rotation matrix error, given as the angles between the actual matrix $\mathbf{R}$ and the estimate $\tilde{\mathbf{R}}$ (TIAN; TOMASI; HEEGER, 1996), which is defined as

$$\varepsilon_{\angle \mathbf{R}} = \cos^{-1}\left(\frac{\text{trace}(\mathbf{R}^\top \tilde{\mathbf{R}}) - 1}{2}\right). \tag{3.23}$$

Figure 3.5 presents the average translation and rotation errors as a function of the angular matching error. The distribution of the 3D points, minimum and maximum values for κ and the FoVs are the same as in the experiment related to Table 3.1. Note that the rotation error is much smaller than the translation error not only for narrow FoVs, as noted in (NISTÉR, 2004; TIAN; TOMASI; HEEGER, 1996), but also for wider FoV cameras. It is also evident that the translation error decreases as the FoV increases since the Essential matrix is estimated more accurately. For the wider FoVs ($195° \times 195°$ and $360° \times 180°$) we also present Wedin's bound for the angle between the actual and the estimated scaleless translation vector, as given in Equation (3.19). For narrower FoVs the

bound is larger than the trivial value, and hence not shown.

Figure 3.5: 5-DoF pose error for different noise levels and FoVs. Source: the author.



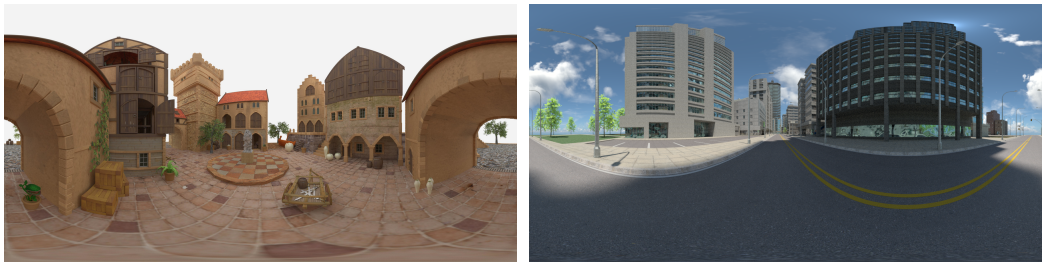
3.5.2 Real Feature Matching

Although the vMF noise model is suitable for the chosen features, matching noise in real images is highly dependent on the feature extractor and the local appearance of the images. Since we are not aware of existing datasets with wide FoV cameras (*e.g.* spherical) and ground truth data *w.r.t.* the Essential matrix, we use realistic computer generated scenes as done in Gava, Stricker and Yokota (2018), Guan and Smith (2017a), Zhang et al. (2016). We rendered non-aligned and non-rectified pairs of spherical images using the Blender models *Urban Canyon* and *Indoor*, made available by (ZHANG et al., 2016), and the *Classroom*, used in (JEONG et al., 2018). We also considered scene captures of the *Medieval Port* model along with the 6-DoF pose ground-truth from (GAVA; STRICKER; YOKOTA, 2018). The former and the latter datasets are outdoors, and the other two are indoors. All images are rendered at a 1280×640 resolution in equirectangular format, and Figure 3.6 illustrates a single spherical view from each dataset.

To obtain the required correspondences for the 8-PA, we used the SPHORB descriptor (ZHAO et al., 2014), which is suited for spherical images, faster than SSIFT (CRUZ-MOTA et al., 2012) and, differently from BRISKS (GUAN; SMITH, 2017a), has a pub-

Figure 3.6: Datasets used for validation. Sources given in the main text.

(a) *Medieval Port* (b) *Urban Canyon*



(c) *Indoor* (d) *Classroom*

licly available implementation[2]. Given the correspondence pairs, we robustly estimate the Essential matrix by using RANSAC. In this experiment, a feature pair $(\mathbf{x}_1, \mathbf{x}_2)$ is considered an inlier if its symmetric projected distance (PAGANI; STRICKER, 2011), given by

$$G_e(\mathbf{x}_1, \mathbf{x}_2) = \frac{|\mathbf{x}_2^\top \mathbf{E}\mathbf{x}_1|}{\|\mathbf{E}\mathbf{x}_1\|} + \frac{|\mathbf{x}_1^\top \mathbf{E}^\top \mathbf{x}_2|}{\|\mathbf{E}^\top \mathbf{x}_2\|}, \tag{3.24}$$

is smaller than 10^{-2}. We accept a model if it has at least 70% of inliers. For the sake of illustration, Figure 3.7 shows a pair of spherical images along with the matched SPHORB features (without restricting the number of features).

Tables 3.2 and 3.3 present the average translation and rotation errors (Equations (3.22) and (3.23)), as well as the epipolar error (Equation (3.12)), the δ value for the four datasets, and the four different FoVs (set as in the synthetic matching experiments). For each experiment, a total of 1,000 pairs of images was randomly selected, and the FoV was restricted so that the narrow FoV cameras are pointing out to some location aligned to the scene's horizon. Besides the results for 8-PA, we also show the pose errors after applying the non-linear 5-DoF pose refinement (NLR) based on the projected distance that is, among other options, pointed out as the best-performing in Pagani and Stricker (2011).

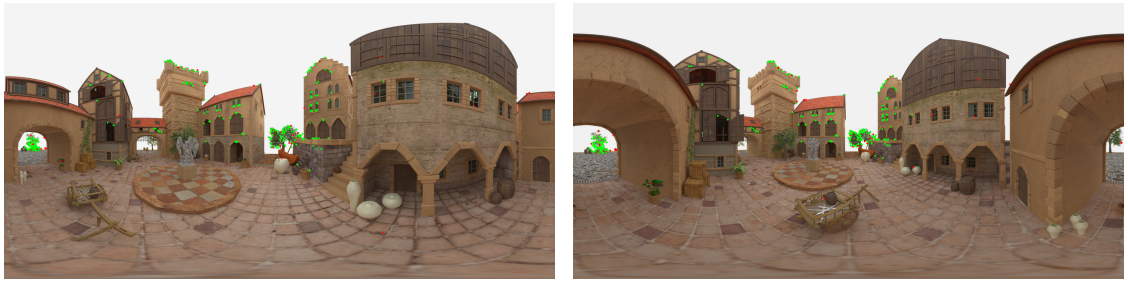
In the experiment shown in Table 3.2, the number of matchings is restricted to be the same as the one of the smaller FoV, so that the main observable variable on the results

[2]Source code available in <https://github.com/tdsuper/SPHORB>.

Table 3.2: Results for synthetic imagery for different FoVs when the number of keypoints is limited.

Metric	$54.4° \times 37.8°$	$65.5° \times 46.4°$	$195° \times 195°$	$360° \times 180°$
$d(\mathbf{E}, \tilde{\mathbf{E}})$	0.849±0.687	0.815±0.698	0.134±0.287	0.100±0.260
$\varepsilon_{\angle \mathbf{t}}$ (8-PA)	39.727±36.301	37.445±35.618	11.941±37.707	8.936±27.426
$\varepsilon_{\angle \mathbf{t}}$ (NLR)	38.497±37.456	35.770±37.816	11.521±37.899	8.214±27.770
$\varepsilon_{\angle \mathbf{R}}$ (8-PA)	8.383±20.913	7.629±18.385	1.923±7.802	1.684±8.952
$\varepsilon_{\angle \mathbf{R}}$ (NLR)	8.276±20.852	7.235±17.511	1.500±7.022	1.453±8.705
δ	0.033±0.017	0.038±0.022	0.150±0.087	0.257±0.157

Figure 3.7: Matched SPHORB features throughout the sphere. Source: the author.



is the spreading of the features. The average number of keypoints in this experiment was 132.8 ± 83.9. Note that the results in synthetic images with real feature matching corroborate the results from Section 3.5.1, *i.e.,* the wider the FoV, the smaller the epipolar and 5-DoF pose errors, and larger the value of the gap δ.

Table 3.3 shows the results using all available matches for each tested FoV (more matches are expected for wider FoVs). The average number of correspondences in this new test indeed increased with the FoV: 134.8 ± 86.0, 166.8 ± 115.1, 722.9 ± 681.3 and 1288.4 ± 1189.0, respectively. Our results indicate that increasing the number of ("good") features indeed helps to improve even more the 8-PA results, especially for wider FoVs. Also, as noted by (ZHANG et al., 2016), wider FoVs greatly improve pose and 3D estimation based on non-linear BA algorithms since features are more likely to be visible in more than two captures of temporally aligned image sets.

Table 3.3: Results for synthetic imagery for different FoVs with free number of keypoints.

Metric	$54.4° \times 37.8°$	$65.5° \times 46.4°$	$195° \times 195°$	$360° \times 180°$
$d(\mathbf{E}, \tilde{\mathbf{E}})$	0.818±0.667	0.777±0.675	0.053±0.611	0.038±0.052
$\varepsilon_{\angle \mathbf{t}}$ (8-PA)	37.280±33.974	35.323±33.994	8.480±34.474	3.427±15.588
$\varepsilon_{\angle \mathbf{t}}$ (NLR)	35.936±34.534	33.728±36.002	8.421±34.469	3.157±15.566
$\varepsilon_{\angle \mathbf{R}}$ (8-PA)	6.053±10.479	5.417±7.015	0.2865±0.5924	0.083±0.077
$\varepsilon_{\angle \mathbf{R}}$ (NLR)	6.203±12.603	5.126±7.209	0.208±0.519	0.065±0.066
δ	0.034±0.017	0.043±0.023	0.388±0.188	0.851±0.365

3.6 Conclusions of the Chapter

We presented a perturbation analysis for Epipolar matrix estimation using the well-known 8-PA by exploring singular subspace analysis. We showed that the sine error bound is inversely proportional to the second least singular value of the observation matrix, which is strongly affected by the spatial distribution of the matched features. In particular, the features extracted when using narrow FoV images are spatially concentrated, leading to larger bounds (and, according to our experiments, also larger errors in the estimate of the Epipolar matrix). On the other hand, cameras with wider FoV (in the limit case, spherical images) present a much better spatial distribution of features, leading to smaller bounds and smaller effective errors in the estimated matrix. The mathematical analysis and experimental results presented in this chapter were published in Silveira and Jung (2019b).

4 DENSE 3D RECONSTRUCTION FROM MULTIPLE SPHERICAL IMAGES

4.1 Overview

Our multi-view 3D reconstruction method estimates a depth value for each pixel (in equirectangular format) of a reference view, relying on one or more additional views. For navigation applications, DIBR techniques are widely applied since they allow the generation of multiple virtual viewpoints using a single image along with its corresponding depth map (OLIVEIRA et al., 2019). Registered color and depth spherical images allow implementing DIBR techniques, which can be used, for instance, to generate stereoscopic data for VR HMD 3-DoF+ exploration (JUNG et al., 2018; JEONG et al., 2018).

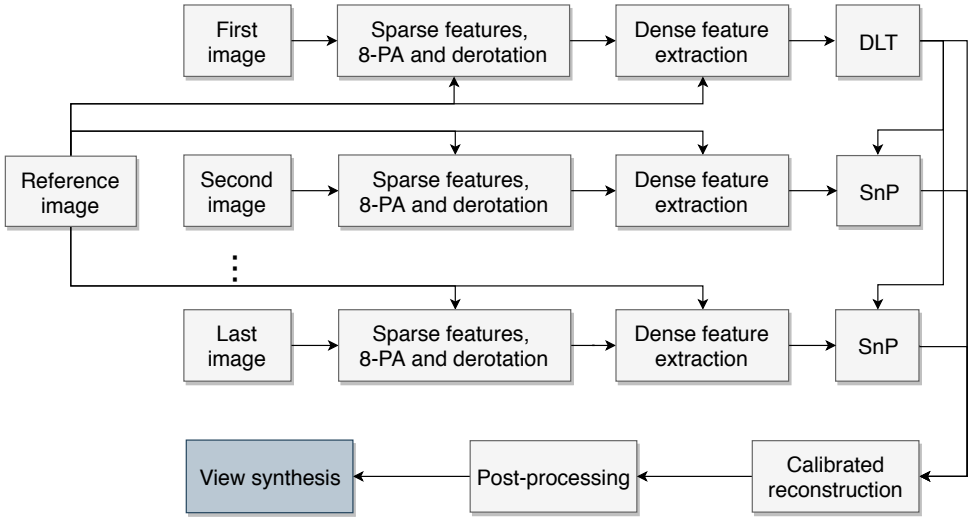
Our method first computes and matches sparse features between the reference view and the others, estimates the 5-DoF camera poses via the 8-PA with outlier removal, and then derotates all other images *w.r.t.* the reference view. As shown in Chapter 3, the estimate of the rotation matrix is accurate when using wide FoV cameras so that our procedure provides a good initial alignment between them. We then explore an optical flow algorithm to obtain dense matches and use both cross-checking and epipolar geometry consistency to detect and remove the contribution of inconsistent flow vectors. By considering a subset of the dense matches with the smallest error, we accurately compute the 6-DoF pose of each view using a linear approach, allowing to skip popular (but expensive) non-linear alternatives based on BA. Finally, we estimate the 3D scene geometry from the calibrated cameras and dense matches by minimizing a weighted error in the 3D space. As an additional contribution, we adapt a fast edge-aware filter to the spherical domain, and use it to smooth the depth map. A schematic representation of the proposed pipeline is shown in Figure 4.1, and the individual modules are detailed next.

4.2 Sparse Feature Matching and 5-DoF Pose Estimation

Recall from Section 2.1 how 3D world points are projected onto spherical cameras, and how two imaged points $\mathbf{x}_1$ and $\mathbf{x}_2$ corresponding to the same world point $\mathbf{X}$ relate to each other by the extrinsic parameters $[\mathbf{R}|\mathbf{t}]$, which are encoded by the Essential matrix $\mathbf{E}$. Moreover, recall the direct and inverse mapping from spherical to image coordinates, also exposed in Section 2.1.

To obtain the required correspondences for the 8-PA, detailed in Section 2.2.1,

Figure 4.1: Overview of the proposed pipeline. Light gray boxes relate to the proposed multi-view 3D reconstruction method. The darker box shows a possible application. Source: the author.



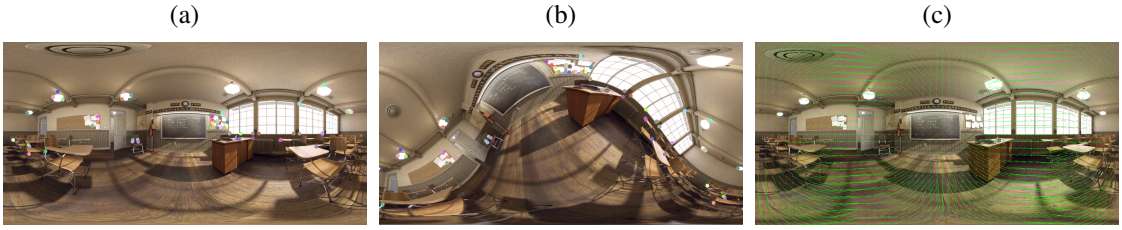
we used SPHORB descriptors (ZHAO et al., 2014), as in the experiments from Chapter 3. We explore RANSAC (FISCHLER; BOLLES, 1981) to filter outliers, using as acceptance criterion the projected distance (PAGANI; STRICKER, 2011) in its symmetric form, as defined in Equation (3.24). Here, a correspondence pair is validated if $G_e(\mathbf{x}_1, \mathbf{x}_2) \leq \lambda_S$ (set to 5×10^{-3} based on experiments), and the model is accepted if it presents at least a fraction λ_P (set experimentally to 70%) of the matched points as inliers.

The parameters $\mathbf{R}$ and $\mathbf{t}$ (omitting the symbol "$\tilde{}$" for simplicity) are obtained from $\mathbf{E}$ through the traditional pipeline that explores the SVD (PATHAK et al., 2016b; PATHAK et al., 2016) associated to the voting scheme from Guan and Smith (2017b) for best pose selection, as shown in Section 2.2.2. At the end of this process – RANSAC 8-PA and extraction of $\mathbf{R}$ and $\mathbf{t}$ –, we are capable of estimating five from the six degrees of freedom because translation magnitude is still unknown. As noted in Mair, Suppa and Burschka (2013), the robustness of the image based pose computation, in general, continuously improves by increasing the FoV of the camera when using the 8-PA. In Chapter 3, we theoretically and experimentally proved that it is indeed true.

4.3 Image Derotation and Dense Feature Matching

The equirectangular projection highly distorts the information depending on its location on the scene, being particularly high near the poles of the sphere (FERREIRA; SACHT; VELHO, 2017), as illustrated in Figures 4.2(a)-(b). As such, the direct appli-

Figure 4.2: (a)-(b) Two views of the same scene and SPHORB matchings (highlighted as colored dots); and (c) optical flow estimates from the derotated version of (b) to (a). Source: the author.



cation of optical flow to obtain dense correspondence tends to fail because of the severe rotation-induced distortions (XU et al., 2017). Similarly to Pathak et al. (2016b), we first derotate all images *w.r.t.* the reference view using the estimated rotation matrix $\mathbf{R}$, and find correspondences using optical flow. To the best of our knowledge, there is no dense optical flow method designed to work on the spherical domain that can deal with large baselines, as required by our approach. From the several options for traditional perspective images, we chose DeepFlow (WEINZAEPFEL et al., 2013), which can handle large displacements and showed good results for derotated spherical pairs. Due to the circular horizontal property of the equirectangular representation, a circular horizontal padding of $1/16$ of the image width is done before computing the optical flow. The derotated version of Figure 4.2(b) overlaid with the dense set of correspondences with Figure 4.2(a) is shown in Figure 4.2(c).

Although the derotation process indeed reduces rotation-based deformations, erroneous matches might still occur along the whole image, such as in textureless regions or, in particular, along the poles of the image. The poles are over-sampled in the equirectangular domain (the top and bottom rows correspond to a small region on the sphere), enhancing the distortions. The problem is further aggravated since, in the most common scenario, the poles contain the ceiling and floor of the scene and both, usually, lack textural information.

To detect bad correspondences, we propose two complementary criteria: photometric and geometric consistency. Photometric consistency is *implicitly* computed from the magnitude of the cross-checking error (RADKE, 2012), given by

$$P_e(\mathbf{p}) = \left\| \Psi^{-1}\left(\mathbf{p}, \mathbf{u}^f(\mathbf{p})\right) + \Psi^{-1}\left(\mathbf{p} + \mathbf{u}^f(\mathbf{p}), \mathbf{u}^b\left(\mathbf{p} + \mathbf{u}^f(\mathbf{p})\right)\right) \right\|, \tag{4.1}$$

for which the forward and backward flows at a given location $\mathbf{p} = (x, y)$ are denoted by $\mathbf{u}^f(\mathbf{p})$ and $\mathbf{u}^b(\mathbf{p})$, respectively. Recall that the function Ψ^{-1} maps the *flow* from image to

camera coordinates, according to Equation (2.15).

The cross-checking error can capture incoherent photometric matches, but it does not encode the geometry of the scene. In particular, in homogeneous regions or close to the poles, we might encounter forward and backward flows that are both wrong but consistent. On the other hand, the epipolar distance in Equation (3.24) indicates how well the matches are regarding the scene/camera geometry, independently from the photometric consistency. Hence, these two measures are complementary, and can be used to build a joint "confidence measure" J_c for each matched pair, represented either by $(\mathbf{x}_k, \mathbf{x}_l)$ in spherical coordinates or $(\psi(\mathbf{x}_k), \psi(\mathbf{x}_l))$ in image coordinates. This measure should be low if any of the two errors is large, which indicates a bad match either concerning photometric or geometric consistency. Inspired by the bilateral filter (TOMASI; MANDUCHI,), which also combines geometric (spatial) with photometric distances, we build the confidence measure by using the 2D anisotropic Gaussian function given by

$$J_c(\mathbf{x}_k, \mathbf{x}_l) = \exp\left(-\frac{P_e(\psi(\mathbf{x}_k))^2}{\gamma_1^2} - \frac{G_e(\mathbf{x}_k, \mathbf{x}_l)^2}{\gamma_2^2}\right), \tag{4.2}$$

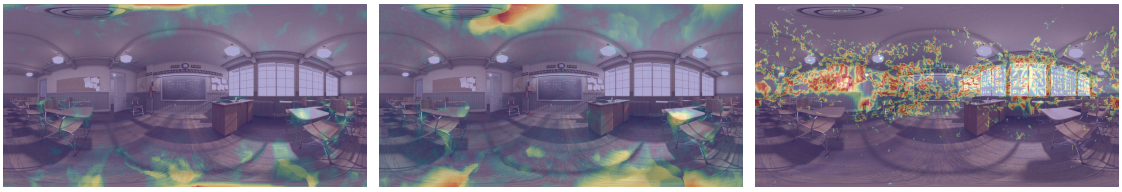
where γ_1 and γ_2 control the decay rate of each term. In our experiments we set $\gamma_1 = \gamma_2 = 10^{-2}$ because both errors vary in similar ranges and we found they are equally important, so that Equation (4.2) becomes an isotropic weighting function.

Figure 4.3(a) shows the cross-checking error for the rectified image pair from Figures 4.2(a) and (b). The symmetric projected distance and the joint confidence map are illustrated in Figures 4.3(b) and (c), respectively. *Colder and warmer colors represent smaller and larger values, respectively.* Please note that the smaller, the better for error metrics, and the opposite for the confidence metric. Notice that the measurements captured by the cross-checking error and projected distance maps may not coincide so that a reliable matching must present both values low, as mentioned before. On the one hand, occlusion-caused errors are identified near the chairs and the desk in Figure 4.3(a). On the other hand, note that the geometric distance highlights several unreliable matches along the poles and textureless regions of Figure 4.3(b). The final confidence metric shows better matches along the equator line. We expect that a bad matching result in one supporting image can be alleviated by a better one in another, as further discussed in Section 4.5.

While other approaches deal exclusively with video sequences (HUANG et al., 2017; GUAN; SMITH, 2017b; IM et al., 2016) or require a known baseline between the camera captures (KIM; HILTON, 2013; KIM; HILTON, 2015; WEGNER et al., 2018; LAI et al., 2019), our method has a smaller constraint that is linked to the quality of the

Figure 4.3: Determining the reliability of each correspondence pair: (a) optical flow cross-checking error; (b) symmetric projected distance; (c) joint confidence map. Colder and warmer colors represent, respectively, small and large values. Source: the author.

(a) $P_e(\cdot)$ (b) $G_e(\cdot)$ (c) $J_c(\cdot)$



obtained optical flow. Pathak et al. (2017) also use a dense optical flow algorithm for estimating the (5-DoF) camera pose and depth from a pair of equirectangular images, and they *empirically* report that all scene objects need to be within 5 to 20 times the translation distance of the camera to obtain reliable correspondences. We intend to further investigate the effect of the camera baseline in the future.

4.4 Stereo Reconstruction and 6-DoF Pose Estimation

As in standard SfM algorithms, our multi-view 3D reconstruction method starts by selecting one image for which the two-view reconstruction will be conducted jointly with the reference view (SCHONBERGER; FRAHM, 2016). Among all images, the one with the highest average reliability *w.r.t.* the reference view according to Equation (4.2) is selected to initialize the process, since it will be used to estimate the 3D points in a first moment. Then, relying on the derotated translation vector obtained via sparse RANSAC 8-PA and the dense correspondences computed by the optical flow, we estimate the 3D geometry via DLT triangulation (ABDEL-AZIZ; KARARA, 1971).

After obtaining a dense 3D representation via DLT, we estimate the 6-DoF of the other non-reference camera views *w.r.t.* the reference camera. Guan and Smith (2017b) proposed an SnP algorithm that linearly optimizes 12 parameters related to the 6-DoF camera pose (9 parameters for rotation and 3 parameters for translation), and then estimates the actual pair of rotation matrix and translation vector by solving an orthogonal Procrustes problem.

In our approach, the rotation matrix of any view *w.r.t.* the reference was already extracted from the Essential matrix when performing the derotation, and its error is far smaller than the 2-DoF translation error, as shown in Chapter 3 and also observed by other authors in experiments with traditional pinhole cameras (TIAN; TOMASI; HEEGER,

1996; NISTÉR, 2004; MAIR; SUPPA; BURSCHKA, 2013). Hence, we reduce the 6-DoF camera pose estimation to only the 3-DoF translation vector $\mathbf{t}$. For image pairs with no rotation (which is the case after derotation), a world point $\mathbf{X}_i = [X_i\, Y_i\, Z_i]^\top$ relates to a spherical image point $\mathbf{x}_i = [x_i\, y_i\, z_i]^\top$ through $\vartheta_i \mathbf{x}_i = \mathbf{X}_i + \mathbf{t}$. Multiplying both sides by the cross product matrix $[\mathbf{x}_i]$ leads to

$$[\mathbf{x}_i]_\times \mathbf{t} = \begin{bmatrix} Y_i z_i - Z_i y_i \\ Z_i x_i - X_i z_i \\ X_i y_i - Y_i x_i \end{bmatrix}, \tag{4.3}$$

and stacking the equations for each matched pair $i = 1, 2, 3, \ldots, n$ leads to an overdetermined linear system, solved by least squares.

An alternative solution would be simply to estimate the scale parameter Λ of the unscaled derotated translation vector $\tilde{\mathbf{t}} = [\tilde{t}_x\, \tilde{t}_y\, \tilde{t}_z\,]^\top$, such that $\mathbf{t} = \Lambda\tilde{\mathbf{t}}$. This leads to a simplification of Equation (4.3), generating a $3n \times 1$ linear system, obtained as the composition of n equations of the form

$$\begin{bmatrix} \tilde{t}_z y_i - \tilde{t}_y z_i \\ \tilde{t}_x z_i - \tilde{t}_z x_i \\ \tilde{t}_y x_i - \tilde{t}_x y_i \end{bmatrix} \Lambda = \begin{bmatrix} Y_i z_i - Z_i y_i \\ Z_i x_i - X_i z_i \\ X_i y_i - Y_i x_i \end{bmatrix}, \tag{4.4}$$

that presents a simple closed-form solution of the least squares error minimization.

It is important to note that for estimating the 6-DoF pose of a given camera view we could either use the same sparse features used in the 8-PA, or all dense features obtained by optical flow. In this section, we adopt an intermediate solution by selecting a subset of the best dense matches ranked based on the proposed confidence metric. We show in Section 4.7.1 that using a sparsified set of dense matches is better than using directly sparse or dense feature matching.

4.5 Multi-view Calibrated Reconstruction

For pinhole cameras, the reprojection error has shown to produce more accurate estimates than the 3D error, and it is considered the gold standard for reconstruction (SZELISKI, 2010). In fact, recently introduced SfM methods for spherical images (PAGANI et al., 2011; GUAN; SMITH, 2017b; HUANG et al., 2017; IM et al.,

2016) also explore the reprojection error in the spherical domain. Here, we investigate the scene recovery problem based on the error computed directly in the 3D space, and conclude that this approach can be effective in the spherical context if a proper weighting scheme is used.

Let us consider a single 3D point $\mathbf{X}$ seen by J rotation-free spherical cameras. If the centers of the derotated cameras are given by $\mathbf{C}_j = -\mathbf{t}_j$, and the estimated projections of $\mathbf{X}$ onto the cameras are $\mathbf{x}_j$, the *weighted* 3D Euclidean error is given by

$$E_{3D} = \sum_{j=1}^{J} w_j \left\| (\mathbf{X} - \mathbf{C}_j)^\top (\mathbf{X} - \mathbf{C}_j) - [\mathbf{x}_j^\top (\mathbf{X} - \mathbf{C}_j)]^2 \right\|, \tag{4.5}$$

where $w_j > 0$ controls the importance of each camera j.

Without loss of generality, we assume that the reference camera (the first one) is centered at the origin of the world coordinate system (*i.e.,* $\mathbf{C}_1 = \mathbf{0}$), and consider that the 3D point $\mathbf{X}$ is aligned with this reference view. This means that $\mathbf{X} = \vartheta \mathbf{x}_1$, where $\vartheta > 0$ represents the distance from the camera center (depth) of point $\mathbf{X}$. Solving $\frac{dE_{3D}}{d\vartheta} = 0$ by least squares yields the following closed form expression:

$$\vartheta = \frac{\sum_{j=2}^{J} w_j \left[\mathbf{x}_1^\top \mathbf{C}_j - (\mathbf{x}_1^\top \mathbf{x}_j)(\mathbf{x}_j^\top \mathbf{C}_j) \right]}{\sum_{j=2}^{J} w_j \left[1 - (\mathbf{x}_1^\top \mathbf{x}_j)^2 \right]}. \tag{4.6}$$

To reduce the influence of outliers, we additionally perform a pre-processing step that aims to remove matches that vote for very large depths (which is implausible since we are dealing with indoor environments). This typically happens when $\mathbf{x}_1$ is almost parallel to a corresponding point $\mathbf{x}_j$ due to matching (or pose estimation) errors. Our outlier removal strategy consists of computing the depth ϑ for each point and camera separately using Equation (4.6) with a single camera j at each time with $w_j = 1$. We then use the upper bound of Tukey's fence (TUKEY, 1977) to define a depth threshold T_d, given by

$$T_d = Q_1 + 1.5(Q_3 - Q_1), \tag{4.7}$$

where Q_1 and Q_3 are the first and third quantiles of the distribution, respectively. We then remove all matched points $\mathbf{x}_j$ that generate a depth value $\vartheta > T_d$, which is equivalent to setting $w_j = 0$.

4.5.1 Weighting Correspondences

The weights w_j in Equation (4.6) should be larger for good correspondences and smaller for bad ones. The quality of the correspondences depends on the estimated poses and the dense matching process performed by optical flow. The results shown in Section 4.7 indicate that the camera poses can be estimated accurately with the proposed pipeline so that errors in the optical flow typically dominate.

Analyzing the sensibility of the 3D projection error of each view j in Equation (4.5) when there is a disturbance in the matchings $\mathbf{x}_j$ is an intuitive way to determine the weights:

$$\frac{dE_{3Dj}}{d\mathbf{x}_j} = -2\mathbf{x}_j^\top(\mathbf{X} - \mathbf{C}_j)(\mathbf{X} - \mathbf{C}_j). \tag{4.8}$$

If $\vartheta\mathbf{x}_1$ is an initial approximation for $\mathbf{X}$ obtained by applying Equation (4.6) with equal weights (e.g. $w_j = 1$), the magnitude of dE_{3Dj} can be estimated as

$$\|dE_{3Dj}\| \approx |\mathbf{x}_j^\top(\vartheta\mathbf{x}_1 - \mathbf{C}_j)|\|\vartheta\mathbf{x}_1 - \mathbf{C}_j\|\|d\mathbf{x}_j\|. \tag{4.9}$$

Views with larger 3D errors $\|dE_{3Dj}\|$ should contribute less to the final 3D estimation so that an adequate weighting function should decay with $\|dE_{3Dj}\|$. In this work we chose an exponential decay function (SU et al., 2015) given by

$$w_j = \exp\left(-\frac{\|dE_{3Dj}\|}{\varrho}\right), \tag{4.10}$$

where $\varrho = \min(\|dE_{3Dj}\|)$ is the minimum error of that point for all the $J-1$ matches, so that the minimum error implies a maximum weight $w_j = 1$. We have also tested other two decay functions, namely $\exp\left(-\left(\frac{\|dE_{3Dj}\|}{\varrho}\right)^2\right)$ and $1/\left(1+\left(\frac{\|dE_{3Dj}\|}{\varrho}\right)^2\right)$, that were used in the context of anisotropic diffusion (PERONA; MALIK, 1990) but the 3D error results were slightly worse.

A key component in Equation (4.9) is the knowledge of $\|d\mathbf{x}_j\|$. We show in Section 4.7.3 that if this term can be approximated, then we can significantly boost the accuracy of the estimated 3D points if compared to the unweighted version (i.e., all $w_j = 1$). Here, we assume that good matches relate to high confidence values as defined in Equation (4.2), and use $\|d\mathbf{x}_j\| \approx 1 - J_c(\mathbf{x}_1, \mathbf{x}_j)$ as an approximation to the matching error. Thus, we relate higher matching errors to smaller weights, and lower matching errors to higher weights, reweighting our calibrated 3D reconstruction method via (4.10).

4.6 Post-processing Using Guided Filters

Smoothness terms from optical flow algorithms may impose spatial coherence in between the matches, preventing outliers (VALGAERTS et al., 2012). However, note that our calibrated reconstruction method considers each image and 3D point individually, neglecting any spatial coherence assumption.

As shown in Barron and Poole (2016), image-guided filters can be used to post-process depth maps produced by conventional stereo matching approaches when pinhole cameras are used. These filters explore the content of the original color image as a guidance to the depth image, such that depth regions with intensity and spatial coherence in the color image are smoothed, but blurring across edges is prevented. Most guided filtering approaches, such as based on the popular joint bilateral (PETSCHNIGG et al., 2004; BARRON; POOLE, 2016) or domain transform (DT) (GASTAL; OLIVEIRA, 2011), employ an isotropic and spatially invariant kernel support σ_s, which controls the influence region of each pixel. However, recall that equirectangular images are not uniformly sampled regarding pixel distance: the poles are much denser than the equator.

From Equations (2.1) and (2.2) and the mapping from spherical to image coordinates (x, y), one can observe that vertical pixel variations Δy lead to constant angular variations. However, horizontal pixel variations Δx lead to spatially varying angular differences along the lines (latitudes) of the image, so that around the poles ($\phi \approx 0$ or $\phi \approx \pi$) large horizontal pixel variations relate to points that are close on the sphere.

Based on these observations, we propose an adaptation of the fast DT approach to deal with the non-uniform sampling of equirectangular images. In the original DT formulation, a single isotropic kernel σ_s is applied to filter, in alternation, the rows and columns of the image. In this work, given a "baseline" value for σ_s, we generate a spatially varying anisotropic version (σ_s^v, σ_s^h) given by

$$\sigma_s^v = \sigma_s, \;\; \sigma_s^h = \sigma_s / \sin\phi = \sigma_s / \sin(\pi y / h), \tag{4.11}$$

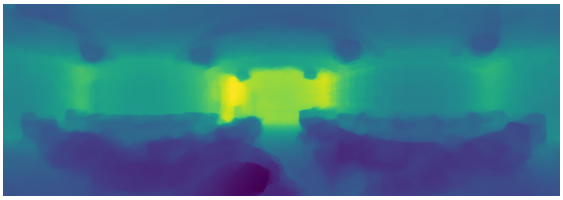
so that the vertical effective kernel σ_s^v is fixed while the horizontal kernel σ_s^h increases in the top and the bottom of the image (i.e., near the poles). These modifications do not impact the linear cost of each iteration of the baseline method, which is very fast. In all experiments, we empirically set the DT filter parametrization $\sigma_r = 0.35$ fixed (color range), and baseline isotropic spatial kernel $\sigma_s = 5.0$.

Figure 4.4: Example of depth estimation based on 9 views from the *Classroom* dataset. Top to bottom: reference view, ground-truth and estimated depth maps using the unweighted, weighted and post-processed approaches. Source: the author.

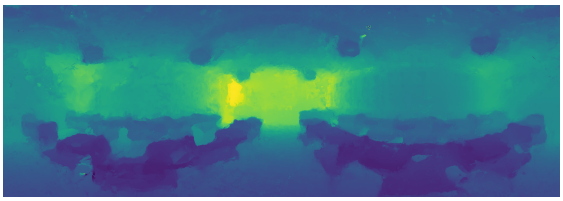
(a) Color image

(b) Ground truth

(c) Unweighted



(d) Weighted



(e) Post-processed

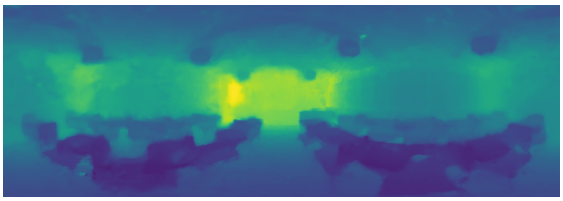


Figure 4.4 illustrates the results of the proposed approach for $J = 9$ views. Figure 4.4(a) and (b) exhibit the reference image and the ground-truth depth map. Figure 4.4 (c) shows the depth estimates with equal camera weights ($w_j = 1$ in Equation (4.6)), and Figure 4.4 (d) presents the results using the proposed weighting scheme. As can be observed, the 3D structure of the chairs is much better preserved in the weighted version, but a little noisier. This phenomenon probably occurs because of inconsistent contributions of the several cameras in neighboring pixels. Figure 4.4 (e) shows the final post-processed results, which smooth the previous depth map but keeps the object boundaries sharp. In fact, we show in Section 4.7.3 that our post-processing also improves the accuracy in the dataset with available ground truth data.

4.7 Experimental Results on the Multi-view 3D Reconstruction Method

This section evaluates our 3D reconstruction method, and presents as an application view synthesis based on DIBR for 3-DoF+ exploration. Our pipeline was sequen-

Table 4.1: Computation time for each step of the proposed pipeline for $J = 3$

Step	Runtime (s)
Sparse feature matching (SPHORB)	1.96
8-PA and derotation	3.24
Dense feature matching (DeepFlow)	32.82
DLT	0.10
SnP (3-DoF)	0.16
Calibrated reconstruction (weighted)	1.08
Post-processing (DT filter)	0.66

tially implemented in Python[1] and ran on a 14GB RAM desktop computer equipped with a 3.07GHz Intel Core i7 processor (no GPU was used). Obtaining a full dense map takes around 40s for $J = 3$, and optical flow computations using DeepFlow are responsible for 82% of the processing time. The main bottleneck of the current pipeline is the DeepFlow algorithm, noting that it can be changed by a faster approach provided that it can deal with both large and small displacements. The runtime for each step is presented in Table 4.1, and it is important to recall that the steps of our method scale linearly with the input size.

For quantitative validation of our multi-view 3D reconstruction approach, we consider both noisy artificial 3D data and captures of a realistic synthetic 3D indoor scenario, similarly to what was done in Chapter 3. We use the software Blender for rendering color and depth from a preset path within the *Classroom* scene, as in Jeong et al. (2018). As a reference, due to the lack of publicly available datasets for multi-view spherical images with pose and depth ground truths, synthetic views were also used for assessing keypoint matching in spherical images (GUAN; SMITH, 2017a) and depth estimation based on calibrated hemispherical image pairs (MOREAU; AMBELLOUIS; RUICHE, 2012). For qualitative validation, we used a set of real captures of three different indoor scenarios using a Samsung Gear 360 camera mounted onto a monopod. Although the used camera may produce artifacts when stitching the fisheye captures, we did not find abnormal depth estimates in our full pipeline, probably because our confidence metric is able to identify incoherent matches on stitching regions. In both datasets, the equirectangular images are resized to 1280×640 due to SPHORB implementation restrictions. We expected that increasing images resolution will lead to better correspondence pairs, and hence higher quality 3D reconstructions.

Aiming to assess the quality of the estimated camera poses (translation and rotation parameters) and the 3D points, we use the relative error (WENG; HUANG; AHUJA,

[1] DT filter, and SPHORB codes are in Matlab and C++, respectively.

Table 4.2: Comparison of SnP algorithms using SPHORB (Sparse) or sparsified set from DeepFlow (Dense) matches, optionally using the non-linear refinement proposed in Guan and Smith (2017b). We show average relative errors for translation and rotation, and runtime (seconds), from top to bottom. Values are scaled to $\times 10^3$ for a better analysis.

Figure of merit	Sparse	Dense	Refined sparse	Refined dense
	Optimization of 12 parameters			
$\varepsilon_{\mathbf{t}}$	44.2 ± 21.1	31.9 ± 27.4	23.7 ± 10.0	13.7 ± 9.4
$\varepsilon_{\mathbf{R}}$	3.7 ± 1.7	3.1 ± 2.5	1.5 ± 0.6	1.2 ± 0.7
Runtime	0.9 ± 0.1	1.7 ± 0.1	268.4 ± 13.9	316.3 ± 21.5
	Optimization of 3 parameters			
$\varepsilon_{\mathbf{t}}$	23.1 ± 11.6	16.8 ± 10.3	23.7 ± 10.0	13.7 ± 9.4
$\varepsilon_{\mathbf{R}}$	1.8 ± 0.7	1.8 ± 0.7	1.5 ± 0.6	1.2 ± 0.7
Runtime	0.4 ± 0.0	0.6 ± 0.1	234.3 ± 26.5	287.4 ± 30.8
	Optimization of 1 parameter			
$\varepsilon_{\mathbf{t}}$	24.6 ± 11.4	18.5 ± 9.1	23.7 ± 10.0	13.7 ± 9.4
$\varepsilon_{\mathbf{R}}$	1.8 ± 0.7	1.8 ± 0.7	1.5 ± 0.6	1.2 ± 0.7
Runtime	0.3 ± 0.0	0.5 ± 0.1	240.8 ± 28.8	294.3 ± 27.4

1989; LIU et al., 2016), defined as

$$\varepsilon_{\mathbf{Z}} = \frac{\|\widetilde{\mathbf{Z}} - \mathbf{Z}\|}{\|\mathbf{Z}\|}, \tag{4.12}$$

where $\mathbf{Z} \in \{\mathbf{R}_j, \mathbf{t}_j, \mathbf{X}_i\}$ and superscript $\widetilde{}$ stands for the estimated values. Note that the assessment metrics presented here differ from those in Chapter 3 because in that case, we were interested in evaluating the angular error directly in the subspaces.

4.7.1 Analysis of the 6-DoF Pose Estimation Algorithm

In this experiment, we assess the impact of using a sparsified set of dense features in the 6-DoF pose estimation problem using different SnP formulations, as described in Section 4.4. In particular, we compare the approaches for 1- or 3-DoF estimation of the translation vector and the 12-parameter estimation presented in Guan and Smith (2017b).

Aiming to isolate only the effect of different pose estimation algorithms, we use the ground-truth data for the world points that are aligned to the reference view, and the actual matchings computed both by SPHORB (sparse) and DeepFlow (dense) algorithms. The results are averaged for 100 random views from the *Classroom* dataset.

We set the target number for SPHORB keypoints to $\mathsf{k} = 2,000$, and associate them in image pairs using the ratio matching strategy (MIKOLAJCZYK; SCHMID, 2005), thresholded to $\lambda_M = 0.75$ (default value in SPHORB's paper (ZHAO et al., 2014)). In

fact, we have also tested other values for k ($\mathsf{k} = 1,000$ and $\mathsf{k} = 3,000$), but the pose estimates were worse. This probably occurred because too few correspondences do not suffice for obtaining a good pose estimate, whereas adding unreliable correspondences may introduce errors. The SPHORB matchings are filtered using RANSAC with same parameters used in Section 4.2.

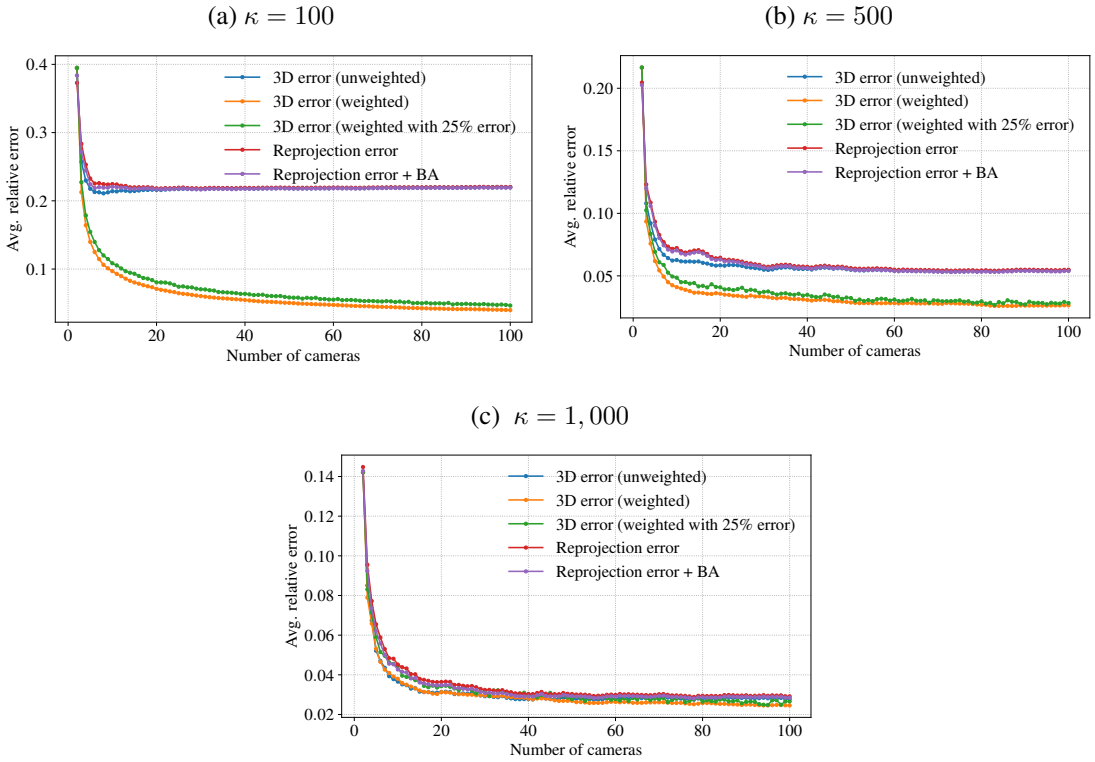
We claim that the confidence metric, given in Equation (4.2), can be used to select an even better set of "keypoints" from the dense set of optical flow matches. For that aim, we rank the correspondences according to $J_c(\cdot,\cdot)$, and select a fraction f of correspondence pairs with the highest confidence for being considered in the SnP algorithm. We found that $f \approx 0.1\%$ of the dense correspondences set, which leads to approximately 819 matches, presents a good compromise between accuracy and processing time. In fact, adding points until approximately $f = 0.5\%$ yields even better results, but running times are higher. However, we focus our analysis on $f = 0.1\%$ because the number of "valid keypoints" obtained by both techniques, SPHORB and sparsified DeepFlow, is comparable. Recall that the three SnP algorithms scale linearly with the input size.

Table 4.2 presents the average translation and rotation errors, as well as the average runtimes, considering all the tested pairs for $f = 0.1\%$. Rotation errors related to the optimization of 3 and 1 parameters were obtained directly from the Essential matrix (recall that these SnP versions use derotated images). As a comparison, we also present the errors and running times for the estimated 6-DoF camera poses after the application of the non-linear pose refinement proposed in Guan and Smith (2017b).

The results in Table 4.2 indicate that: (i) both approaches for estimating the translation vector presented in Section 4.4 lead to smaller translation errors than the one in Guan and Smith (2017b), using either sparse or dense set of matches; (ii) the proposed approach to select sparsified dense matches from the optical flow leads to better estimates than using sparse SPHORB features for all tested approaches; (iii) using the rotation matrix estimated directly from the Essential matrix is better than re-estimating all pose parameters (6-DoF), as done in Guan and Smith (2017b); (iv) adding a non-linear refinement may improve the pose estimates, but with an overhead of two to three orders of magnitude in runtime; and (v) translation errors are about ten times higher than the rotation ones, corroborating the findings of Tian, Tomasi and Heeger (1996), Nistér (2004) for pinhole cameras.

Based on the presented results, we adopt the 3-parameter version of the SnP algorithm introduced in Section 4.4 without any non-linear refinement as our default approach.

Figure 4.5: Relative error of the 3D estimates for $J = 2, \ldots, 100$ cameras and different vMF noise levels. Source: the author.

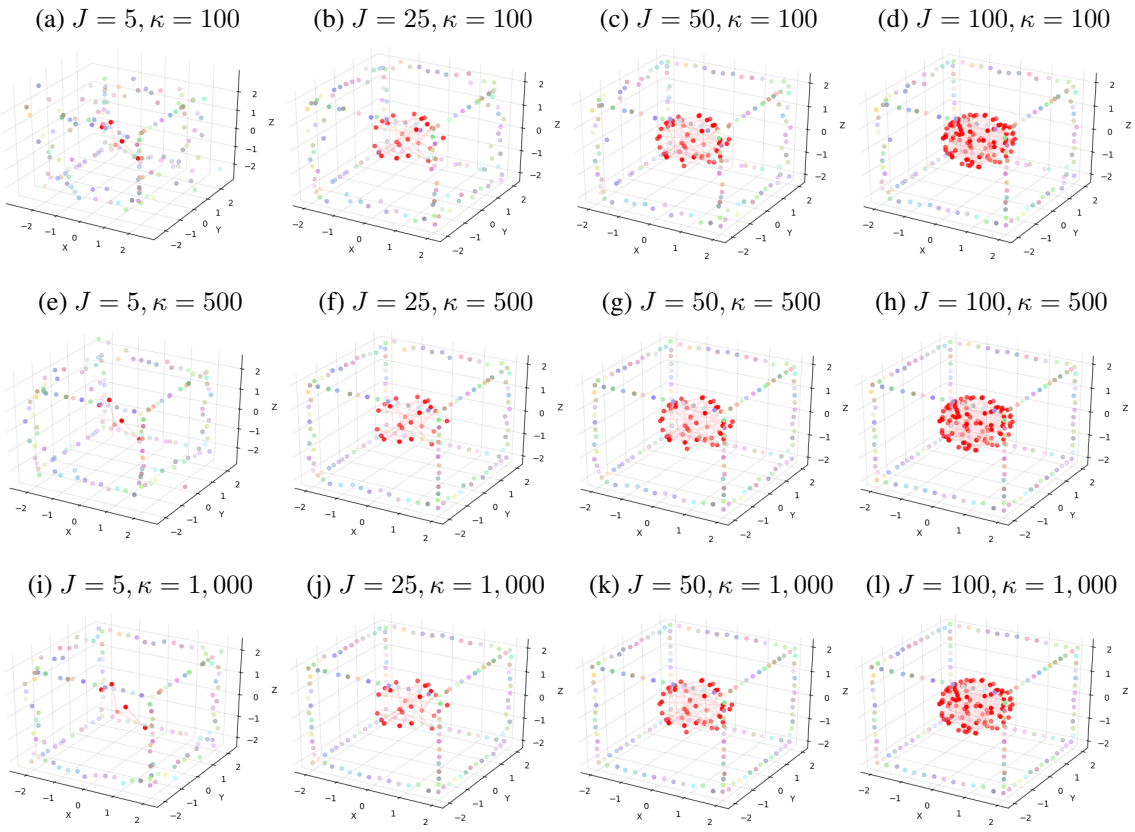


4.7.2 Analysis of the Calibrated 3D Reconstruction

Aiming to evaluate the results produced by the proposed calibrated reconstruction algorithm (confer Section 4.5) on different camera setups and matching noise levels, we rely on a synthetic experiment. We randomly select a set of $J - 1$ points representing the camera centers, which are uniformly distributed in the cube $[-1, 1]^3$ around one reference view at the center of the world coordinate system. We also select at random n 3D world points, in which each vector coordinate is in the interval $[2, 4]$ and project them onto the cameras via Equation (2.1). The adopted distribution for the cameras and 3D points roughly emulate a small indoor scenario, and it guarantees that a point will not be surrounded by opposite-located cameras, which would result on occlusions and lack of matchings.

As in the experiments of Chapter 3, we consider the vMF distribution to model the matching error. The vMF also encompasses the rotation errors in the camera estimation procedure, and an error component from $\mathcal{N}(0, \sigma_t^2)$ is added for each translation vector coordinate in all the $J - 1$ non-reference cameras. Also, recall that our calibrated reconstruction method assumes that we can measure the matching errors $\|d\mathbf{x}_j\|$. In our tests,

Figure 4.6: Calibrated reconstruction of a cube-like 3D structure with different number of cameras J and level noises κ. Source: the author.



we also add an uncertainty factor of (on average) 25% to the $\|d\mathbf{x}_j\|$ estimates.

Figure 4.5 presents the average relative error for the 3D points considering $1,000$ simulations using $J = 2, \ldots, 100$, $n = 100$, and $\sigma_t = 0.05$. The method referred to as "weighted" is based on Equations (4.6) and (4.10), whereas the "unweighted" approach is also based on Equation (4.6) but considers all equal weights w_j (*i.e.*, $w_j = 1$). We tested different vMF noise levels by selecting $\kappa \in \{100; 500; 1,000\}$, which represents an average angular matching error on the sphere around $7.25°$, $3.20°$ and $2.27°$, respectively. For comparison purposes, we also show the results obtained by the calibrated reconstruction method from Guan and Smith (2017b), which is based on the reprojection error. We also show their results after refining the estimates with the BA algorithm they propose. Note that their linear method optimizes the 3D structure by taking an equal contribution from all the cameras. Although the impact of their BA algorithm seems to be small, it may improve not only the 3D structure but also the camera poses, which is the main focus in Guan and Smith (2017b).

We have also tested the scalable BA algorithm from Agarwal et al. (2010), which minimizes the reprojection error in a Euclidean sense, as done in Pagani et al. (2011) and

Huang et al. (2017), but the results were similar. It is worth mentioning that although the BA algorithm from Agarwal et al. (2010) may scale *w.r.t.* the number of cameras, its computational burden is still prohibitive for a dense set of matches. For instance, running times for processing a subset of 25% of the total number of pixels using only a pair of cameras takes over 417s in our hardware, and the optimization process did not finish when using the full frame. Figure 4.6 provides an illustration of how the proposed calibrated 3D reconstruction algorithm (with 25% of uncertainty) performs under different noise conditions ($\kappa \in \{100; 500; 1,000\}$) and depending on the number of cameras ($J \in \{5; 25; 50; 100\}$). For simplicity, the cameras – points in red – are randomly selected within a region with radius 1. Note that, as both κ and J increase, the 3D geometry (colored points) is better recovered, approaching a set of cube-like edges.

Our experiments varying the number of cameras and matching noise levels indicate that: (i) error estimates using even the unweighted 3D error are smaller than the reprojection error (with and without BA refinement) used in Guan and Smith (2017b) for all tested vMF noise levels; and (ii) the re-weighting scheme highly improves the estimates of the reconstructed scene points, even when considering an error of around 25% when estimating $\|d\mathbf{x}_j\|$. This gain is more evident as the noise level and the number of cameras increase.

4.7.3 Evaluation of the Complete Pipeline

We quantitatively assess the quality of the 3D reconstruction obtained with the complete pipeline by using J randomly selected camera views from the *Classroom* dataset. Similarly to Chang et al. (2017), Won, Ryu and Lim (2019), we keep latitudinal information from -62° to 62°, because optical flow matchings tend to be noisy on the poles. Notice that the discarded part of the equirectangular image corresponds to a small region in the 3D scene. Table 4.3 presents the average relative error and standard deviation from 30 simulations, for $J \in \{3, 5, 7, 9\}$. Note that the weighted version of the calibrated reconstruction presents lower average error and standard deviation than the unweighted one, corroborating the results obtained with synthetic matching. These results are an evidence that the error in $\|d\mathbf{x}_j\|$ can be well approximated by $1 - J_c(\mathbf{x}_1, \mathbf{x}_j)$.

Table 4.3 also indicates that post-processing the weighted map using the DT filter adapted to the spherical domain provides additional error reduction besides improving the visual results by adding spatial coherence to the depth map as shown in Figure 4.4. In

Table 4.3: Average relative error of the 3D scene reconstruction.

Number of cameras	Method		
	Unweighted	Weighted	Post-processed
3	0.1941 ± 0.0690	0.1515 ± 0.0321	0.1469 ± 0.0279
5	0.1921 ± 0.0815	0.1277 ± 0.0346	0.1201 ± 0.0283
7	0.1702 ± 0.0343	0.1111 ± 0.0189	0.1074 ± 0.0177
9	0.1597 ± 0.0175	0.1051 ± 0.0186	0.1020 ± 0.0172

fact, the Welch's t-test (WELCH, 1947) indicated that the equal mean (null) hypothesis when comparing the unweighted with the post-processed version can be rejected at the 1% significance level for all values of J tested.

Unfortunately, as already mentioned, we are not aware of other techniques that tackle the 3D scene reconstruction problem in the same scenario as we do (multiple unordered and uncalibrated views). Most techniques deal only with stereo-rectified images, multi-view calibrated camera setups, small motion video sequences, or single-view (confer Section 2.4). Moreover, we could not find source code available for which we could test the other methods performance using a common dataset and metrics.

Finally, Figure 4.7 presents visual results for typical 3D reconstructions provided by the proposed pipeline. The columns show the reference view, the post-processed depth map, and a novel synthesized view (obtained by directly rendering the point cloud, without treating occlusions or disocclusions), respectively. The rows consider image sets containing 8, 3 and 5 real spherical images, respectively. The first one is a private indoor parking, exemplifying a wider environment; the second one is a small living room, lacking texture on most of the walls; and the third one is a medium size room, illustrating a cluttered environment with translucent doors and lateral window.

Figure 4.7: Examples of results produced by our approach. From left to right: the reference image, estimated depth map, and a view of the resulting point cloud, respectively. Source: the author.

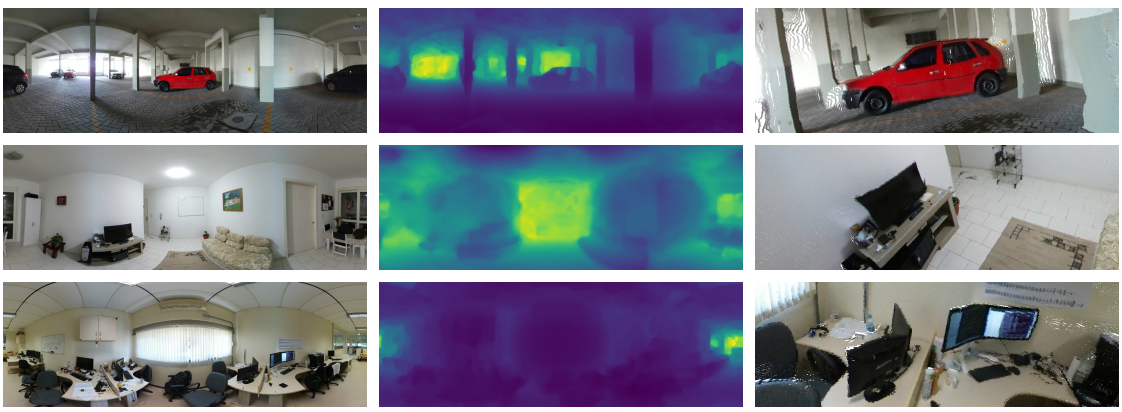
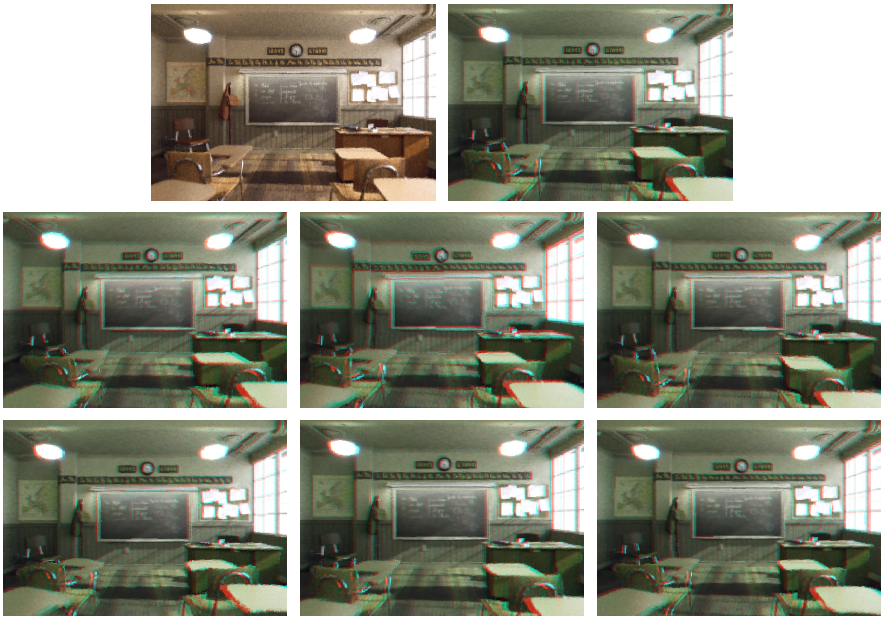


Figure 4.8: Example of 3-DoF+ exploration. First row: reference image and stereo visualization in the original position; Other rows, from left to right: synthetic stereo views after moving the virtual camera to the left, right, up, down, forward and backward. Source: the author.



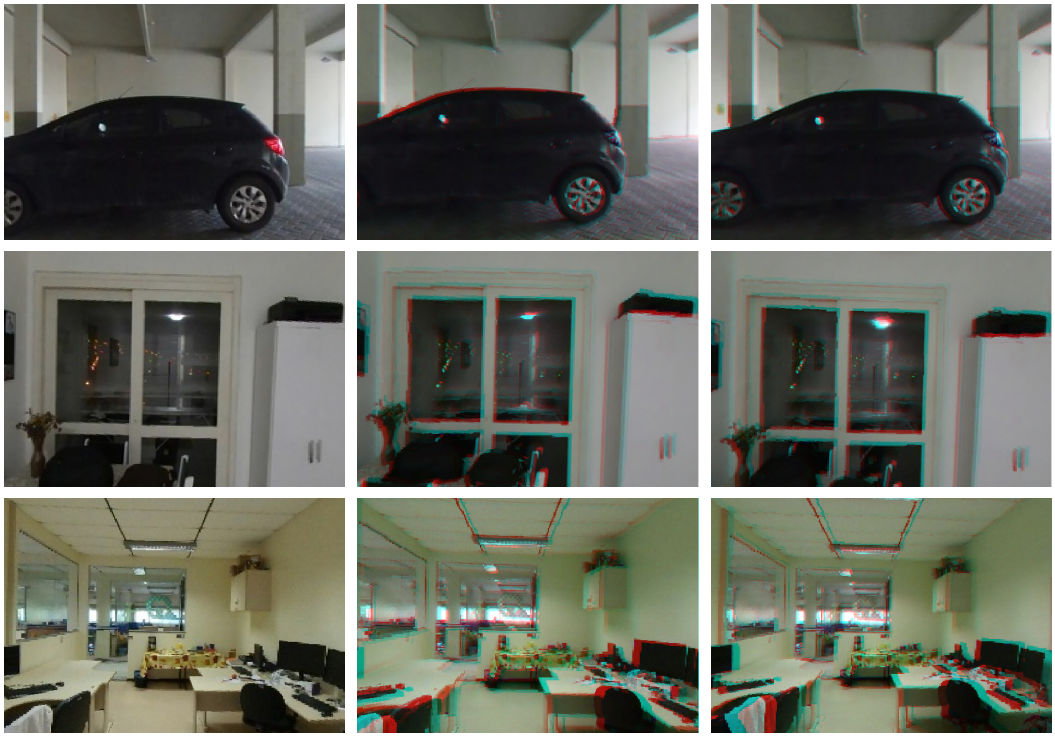
4.7.4 Application to 3-DoF+ Exploration

The proposed 3D reconstruction method provides a depth map registered to a color spherical image, which is the default input for DIBR approaches. As noted in Schwarz and Hannuksela (2017), traditional DIBR approaches can be used for generating small baseline synthetic views such as those required for small head movements or binocular stereopsis for 3D immersion in VR 3-DoF+ setups.

The literature on DIBR is vast (please refer to Atapour-Abarghouei and Breckon (2018) for a recent review), and for small baselines, only a few pixels are expected to be affected by occlusions/disocclusions. The hierarchical hole-filling (HHF) approach introduced in Solh and AlRegib (2012) presents compelling results for filling small holes at low computational cost and was used to generate the synthesized views in this work. We apply HHF to the narrower-FoV ($70^\circ \times 52^\circ$) images, constrained to the user's viewport, which are not deformed by the camera model. Although we considered HHF, any other DIBR approach could be used with the proposed 3D reconstruction scheme, as well as the warping scheme presented in Huang et al. (2017). Recall, however, that the color plus depth representation is suitable for 3-DoF+ only, while the mesh representation from Huang et al. (2017) can deal with larger displacements (6-DoF).

Figures 4.8 and 4.9 illustrate the synthesis of binocular views produced by our

Figure 4.9: Examples of 3-DoF+ exploration. From left to right: a view-port of the original image, and two synthesized binocular views after moving the camera. The motion is to left and right in the first row, up and down in the second, and forward and backward in the third row. Source: the author.

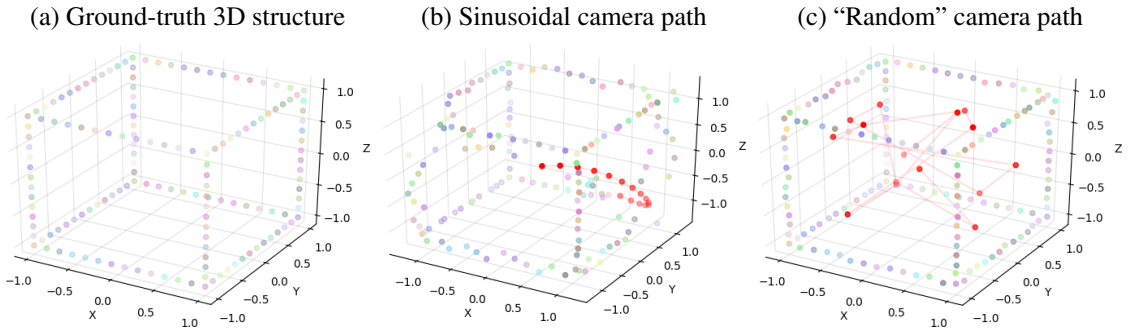


method together with the HHF algorithm for hole filling, shown in red-cyan anaglyphs. More precisely, they relate to the color-plus-depth image pairs shown in Figures 4.4 and 4.7, respectively. We provide synthetic binocular views after moving the virtual camera to the left and right, up and down, as well as forward and backward motion. We are not aware of other open-source methods for producing and visualizing binocular images from our 360° texture plus depth representation. At the moment we have not tested our approach in real VR HMDs, but we plan to do those tests in the future. The readers can watch videos with synthesized binocular views for the scenes in Figures 4.8 and 4.9 at <www.inf.ufrgs.br/~tltsilveira/thesis-videos>.

4.7.5 Capture Guidelines

The theoretical error analysis given by Equations (4.8) and (4.9) indicates that choosing a camera center $\mathbf{C}_j$ close to a 3D point $\mathbf{X}$ helps to reduce the error estimate of that point. Although this is indeed true, we do not recommend using views close to scene objects, since such views tend to present significant projection-induced distortions which

Figure 4.10: Calibrated reconstruction of cube-like 3D structure based on $J = 15$ cameras and $\kappa = 1,000$. Camera path following a (b) sinusoidal path and a (c) "random" path. Ground-truth 3D structure in (a). Source: the author.



might compromise the optical flow computation. Furthermore, we recommend to align the most important content of the scene to camera's horizon, since the dense correspondences in this region produced by optical flow tend to be more reliable (confer Figure 4.3c).

Another point to be noted is that when using only two cameras, 3D points aligned with the epipoles cannot be estimated. In those cases, $\mathbf{X}$, $\mathbf{C}_2$, $\mathbf{x}_1$ and $\mathbf{x}_2$ are all co-linear, so that the depth estimate ϑ in (Equation (4.6)) is undefined. Hence, in practical scenarios, we suggest using three or more views by displacing the camera in orthogonal directions *w.r.t.* the reference view, so that the epipole ambiguity generated by one pair might be solved by other(s). Figure 4.10 illustrates the impact of the camera positioning in the presence of noisy correspondences. For this example, we use $J = 15$ cameras and an average matching error of $2.27°$. The 3D points are estimated using our 3D calibrated reconstruction algorithm. Note that when the captures are approximately aligned (example using a sinusoidal path), the recovered 3D points near the epipoles tend to be much more inaccurate than when considering scattered captures.

4.8 Conclusions of the Chapter

In this chapter, we introduced a pipeline for estimating the 3D scene geometry of indoor scenes from a set of unordered spherical images. The proposed approach, published in Silveira and Jung (2019a), explores sparse feature matching to compute the 5-DoF pose of all the cameras *w.r.t.* a reference view, and then derotate those images so that optical flow can be applied to obtain dense matches. A sparsified set of these matches is used to estimate the full 6-DoF pose of all cameras, and a calibrated 3D reconstruction is performed by minimizing a weighted error in the 3D space. An image-guided filter

tailored to the spherical domain is then applied to impose edge-aware spatial smoothness, generating a depth map registered to the reference image. Finally, the color-plus-depth image pair is used to generate binocular stereo image pairs and new synthesized views 3-DoF+ configurations.

Our experimental results showed that (i) the derotation process based on the Essential matrix indeed allow the use of optical flow methods to obtain dense matchings; (ii) using a subset of the best dense matches to extract the 6-DoF pose of the cameras produces better results than sparse matching using SPHORB; (iii) the proposed weighting scheme based on the photometric-geometric confidence metric preserves better smaller 3D structures than the unweighted version; (iv) the post-processing scheme smooths the depth map while preserving depth discontinuities; and (v) the color-plus-depth image pair obtained with the proposed approach can be coupled with existing DIBR approaches for generating coherent binocular stereo pairs and small head motion parallax.

5 AGGREGATING SPATIAL INFORMATION FOR 6-DOF POSE AND DEPTH ESTIMATION

In this chapter, we explore how to improve the 3D scene reconstruction method presented in Chapter 4 by aggregating information from appearance-consistent segments. In particular, we first adapt a superpixel algorithm to the spherical domain aiming to acquire meaningful image segments for modeling the spatial constraints. This novel algorithm, called spherical simple non-iterative clustering (SSNIC), is introduced in Section 5.1. We explore SSNIC segments for selecting scattered points for 6-DoF pose estimation, going in the same direction of what was explored in Chapter 3 in the context of Epipolar matrices estimation. The intuition behind this analysis and some experimental results are shown in Section 5.2. We then use neighborhood information provided by SSNIC superpixels to improve the point-wise calibrated 3D reconstruction approach that was presented in Section 4.5. This spatially-consistent calibrated reconstruction formulation is detailed in Section 5.3. Finally, Section 5.4 concludes this chapter.

5.1 Oversegmentation of Spherical Images

Superpixel algorithms cluster pixels into perceptually meaningful regions that can be used to replace the common point-wise image representation (ACHANTA et al., 2012). This process is also often called image oversegmentation (LIU; SHEN; LIN, 2015). Classic superpixel algorithms have been applied as fundamental building blocks of several techniques that tackle relevant problems, such as relative two-view pose estimation (XIAO et al., 2019), depth map estimation (ZHANG et al., 2015) and novel view interpolation (TEZUKA et al., 2015).

Similarly to other tasks, image oversegmentation is much more explored in the context of perspective images than omnidirectional. To the best of our knowledge, there are only two approaches for adequately computing superpixels on spherical images. Both extend methods suited for perspective images, namely the simple linear iteration clustering (SLIC) (ACHANTA et al., 2012) and the efficient graph-based segmentation (EGS) (FELZENSZWALB; HUTTENLOCHER, 2004). These domain-adapted techniques are called spherical SLIC (SSLIC) (ZHAO et al., 2018) and PanoEGS (YANG; ZHANG, 2016), respectively. It is important to note that EGS generates irregularly shaped super-

pixels and does not offer precise control of the number of regions and their compactness (ACHANTA et al., 2012). PanoEGS inherited those undesirable characteristics and was outperformed by SSLIC in most common generic segmentation metrics (ZHAO et al., 2018).

5.1.1 Spherical Simple Non-Iterative Clustering

Although SSLIC has a publicly available implementation, we opt to adapt a pinhole-based superpixel algorithm that has proven to outperform the traditional SLIC in many aspects. The simple non-iterative clustering (SNIC) algorithm presents many improvements concerning SLIC that we can take advantage and use on its spherical version. Namely, SNIC contrasts with SLIC mainly because (i) the pixel connectivity is enforced from the beginning; (ii) there is no need for multiple k-means iterations; (iii) there are fewer pixel visits and distance computations; and (iv) it consumes less memory.

All those characteristics are kept in our adaptation to the spherical domain. In fact, we use the original open-source code from Achanta and Süsstrunk (2017)[1] and adapt the key aspects to adequate it to the new image domain. We name this novel algorithm as spherical SNIC (SSNIC).

The basic SNIC/SSNIC algorithm is outlined as follows. We start by selecting and associating an initial centroid location for each of the segments that will be generated based on the target number R of superpixels. Those points have a distance value set to zero and feed a priority queue. While this priority queue is not empty, the element with the minimum distance to some centroid is popped out and, if not labeled, it is assigned to the superpixel associated to that centroid. The superpixel centroid is updated online, and the label-less neighbors of the popped element are pushed to the priority queue after having the distance computed to the current centroid. When the priority queue is empty, meaning that all pixels were assigned to some segment, the algorithm finishes. For simplicity we also present the pseudo-code for SNIC algorithm (ACHANTA; SÜSSTRUNK, 2017) in Algorithm 1.

[1] Source code available in <https://ivrl.epfl.ch/research-2/research-current/research-superpixels/research-snic_superpixels>.

Algorithm 1: SNIC algorithm (ACHANTA; SÜSSTRUNK, 2017).

Input : Image I, R initial centroids $\{\mathbf{x}_r, \mathbf{c}_r\}$, color normalization factor m

Output: Assigned label map L

Initialize $L[:] \leftarrow 0$

for $r \in [1, 2, \ldots, R]$ **do**

  Element $e \leftarrow \{\mathbf{x}_r, \mathbf{c}_r, r, 0\}$

  Push e on priority queue Q

end

while Q *is not empty* **do**

  Pop Q to get e_l

  if $L[\mathbf{x}_l]$ *is* 0 **then**

    $L[\mathbf{x}_l] = r_l$

    Update centroid $\bar{\mathbf{x}}_l$ and $\bar{\mathbf{c}}_l$ online

    for *Each connected neighbor* $\mathbf{x}_n$ *of* $\mathbf{x}_l$ **do**

      Crete element $e_n = \{\mathbf{x}_n, \mathbf{c}_n, k_l, dT(\bar{\mathbf{x}}_l, \bar{\mathbf{c}}_l; \mathbf{x}_n, \mathbf{c}_n)\}$

      if $L[\mathbf{x}_n]$ *is* 0 **then**
        Push e_n on Q

      end

    end

  end

end

return L

According to Zhao et al. (2018), there are some problems when adapting a superpixel algorithm to the spherical domain that we need to handle, such as (i) the distribution of the initial centroids for superpixel growing; (ii) the boundary connectivity and pixel neighborhood; and (iii) the distance between image points and superpixels. We discuss how we deal with these issues in detail next.

The first important question that arises is how to distribute the initial superpixel centroids, also called seeds. Depending on the application, it may be highly desired to spread the seeds uniformly so that superpixel algorithms like SLIC/SNIC and SSLIC/SSNIC can produce well distributed and approximately equal-sized labels. The regular grid approach is widely adopted in traditional superpixel algorithms suited for perspective images (ACHANTA et al., 2012; ACHANTA; SÜSSTRUNK, 2017). However, it is not adequate in the context of equirectangular images because, as already mentioned, they

are non-uniformly sampled (FERREIRA; SACHT; VELHO, 2017).

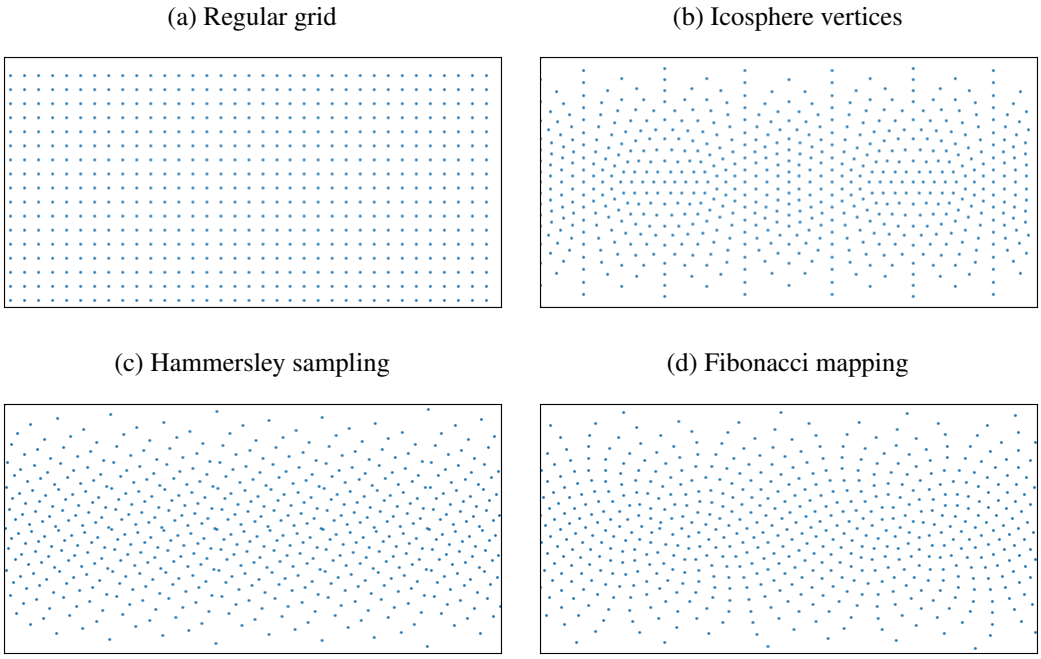
A simple way to generate uniformly distributed points on the sphere surface is to build an icosphere (successive subdivisions of an icosahedron) (EDER; FRAHM, 2019) and select its vertices as seeds. However, as argued by Zhao et al. (2018), this choice restricts the possible number of superpixels to the icosahedral structure for which the quantity of vertices grows in a geometric factor according to the subdivision level. For instance, an icosphere with 0, 1, 2, 3 and 4 subdivisions presents 12, 42, 162, 642 and 2562 vertices, respectively. Because of that, SSLIC adopts an approach that generates Hammersley points (CUI; FREEDEN, 1997) on the plane and further maps those points to the unit sphere via linear scaling and z-preserving radial projection (ZHAO et al., 2018). In practice, however, SSLIC does not guarantee that the number of generated superpixels will be the same as specified.

Instead of Hammersley sampling, we propose to use the spherical Fibonacci mapping (KEINERT et al., 2015), which provides a near-equidistant distribution of a generic number of seeds on the surface of the unit sphere. Figure 5.1 illustrates the four mentioned sampling approaches represented in equirectangular format. It is worth mentioning that although the target number of seeds was set to $R = 600$ for this example, the actual number of points on the regular grid and points corresponding to the icosphere vertices were 595 and 642, respectively. Hammersley sampling and the spherical Fibonacci mapping do generate the correct number of seeds, noting that the Hammersley sampling sometimes proposes very close pairs of points, as shown in Figure 5.1(c).

Another point to be tackled when adapting traditional superpixel algorithms to the spherical domain concerns the intrinsic horizontal circularity of equirectangular images, *i.e.,* the right and left boundaries of the image should be connected. This simple issue demands to redefine SLIC's search region for potential superpixel candidates depending on how close they are from the image boundaries (ZHAO et al., 2018). On the other hand, the association between pixels performed by SNIC is enforced from the beginning, by either 4- or 8-point connectivity. Thus, in practical terms, checking if a candidate pixel location (x, y) of a given segment is out of the image boundaries ($x < 0$ or $x > w$), and if so, adding/subtracting the image width w value generates the circular superpixel connection. Note that candidate positions out of the vertical bounds should be discarded. The whole process of selecting potential candidates to a given superpixel is simplified in the SNIC approach because there is no need for defining a search region as in SLIC.

The last issue that needs to be addressed concerns how to measure the distance

Figure 5.1: Choice of the seeds for spherical superpixel algorithms. Source: the author.



between pixels and superpixels centroids. As in SLIC, SNIC and SSLIC, we also use a distance measure that combines both spatial and appearance distances together. However, instead of considering the L2-norm of integer image coordinates (ACHANTA et al., 2012; ACHANTA; SÜSSTRUNK, 2017), we opt to approximate the geodesic distance on the sphere by using the cosine dissimilarity, given by

$$dS(\bar{\mathbf{x}}, \mathbf{x}_c) = 1 - \bar{\mathbf{x}}^\top \mathbf{x}_c, \tag{5.1}$$

where $\bar{\mathbf{x}}$ and $\mathbf{x}_c$ are points on the unit sphere S^2 that denote the locations of a superpixel centroid and a candidate pixel, respectively. The appearance distance is given by the Euclidean norm between the color components in CIE L*a*b* space of the superpixel centroid and the candidate pixel (ACHANTA et al., 2012; ACHANTA; SÜSSTRUNK, 2017), given by

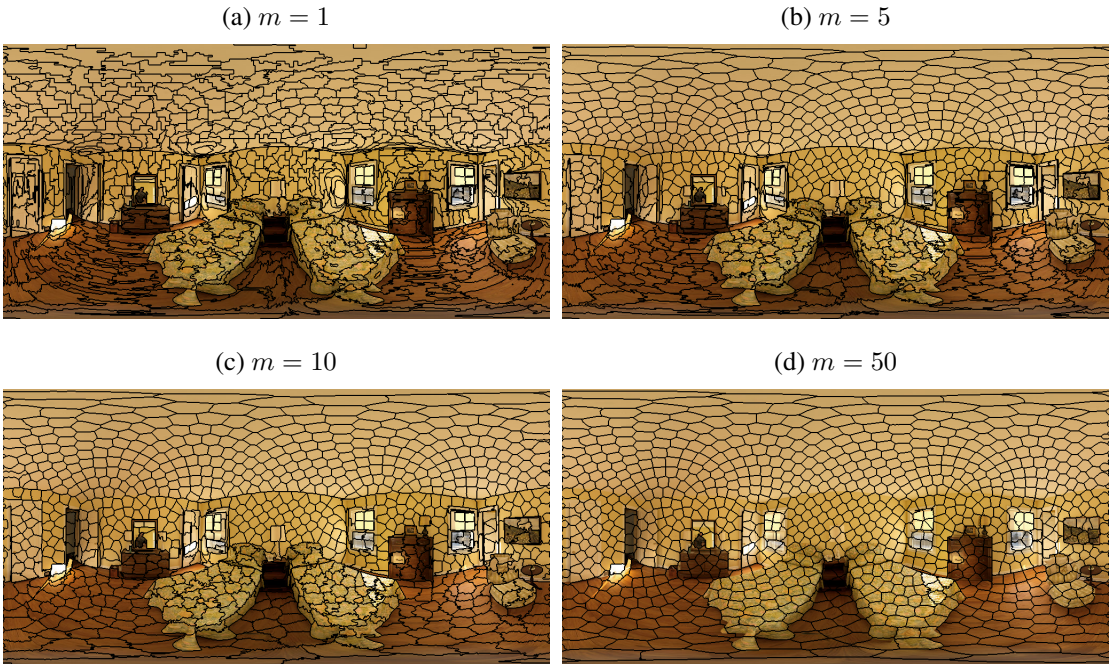
$$dC(\bar{\mathbf{c}}, \mathbf{c}_c) = \|\bar{\mathbf{c}} - \mathbf{c}_c\|_2, \tag{5.2}$$

where $\bar{\mathbf{c}} = [\bar{c}_l\ \bar{c}_a\ \bar{c}_b]^\top$ and $\mathbf{c}_c = [c_l\ c_a\ c_b]^\top$ encode the respective color channel measurements.

Finally, the combined distance between a given superpixel and the candidate pixel $\mathbf{x}_c$ is given by

$$dT(\bar{\mathbf{x}}, \bar{\mathbf{c}}; \mathbf{x}_c, \mathbf{c}_c) = \frac{dS(\bar{\mathbf{x}}, \mathbf{x}_c)}{s} + \frac{dC(\bar{\mathbf{c}}, \mathbf{c}_c)}{m}, \tag{5.3}$$

Figure 5.2: The impact of varying the compactness parameter m in SSNIC algorithm with fixed $R = 1,000$ on a 1024×512 equirectangular image from the SUN360 dataset (XIAO et al., 2012). Source: the author.



where s and m are normalizing factors for spatial and color distances, respectively. Assuming equal probability for expanding each superpixel, we set $s = \sqrt{4\pi/R}$, recalling that R is the number of pixels. Parameter m is the compactness factor, which is user-provided. It controls the compromise between image border adherence and superpixel regularity. Smaller values for m implicates tighter border adherence, whereas higher values for m will result in more regular superpixel regions. Figure 5.2 presents some examples illustrating the effect of varying m on the generated superpixels.

Our SSNIC approach keeps the original SNIC performance inherited from Algorithm 1. For instance, 1280×640 images with a target number of superpixels set to $10, 10^2, 10^3, 10^4, 10^5$, and 10^6 are segmented in 1.72, 1.82, 1.87, 1.95, 2.16 and 2.35 seconds by SSNIC, whereas the official SSLIC implementation takes 7.36, 7.83, 8.28, 20.48, 41.09, and 56.43 seconds, respectively.

5.2 Toward the Selection of Scattered Features for 6-DoF Pose Estimation

Chapter 3 presented a perturbation analysis for the 8-PA, and we concluded that the accuracy of the method for estimating Epipolar matrices is highly dependent on the noise level and the spatial distribution of the matched features on the surface of the sphere.

In this section we aim to analyze the performance of the 1-, 3- and 12-DoF SnP algorithms discussed in Section 4.4 also concerning the feature distribution. Recall that, differently from the 8-PA, SnP tackles the problem of estimating the 6-DoF camera pose of a new capture with respect to a known 3D geometry representation of the scene. For that aim, SnP uses a set of points in the 3D space ("world points") and the corresponding projected points in image coordinates.

The motivation for including the current analysis is that if the distribution of features points matters for the SnP problem as it matters for the Epipolar matrices estimation, then we can force the selection of scattered points on the sphere surface based on superpixel segmentation. In particular, unlike other approaches, the proposed SSNIC method encourages segments that are uniformly distributed over the sphere surface due to the spherical Fibonacci mapping for seed selection. Aiming to assess the validity of this hypothesis, we performed two experiments, considering both synthetic and real feature matching. Theoretical analysis remain as a future work.

Our first experiment uses randomly generated points, similarly to Sections 3.5.1 and 4.7.2. As in Section 4.7.1, we assess the accuracy of SnP algorithms considering ground-truth world points, attempting only to the evaluate the effects of correspondence noise. In the current experiment, noise amount and feature distribution are controlled by the vMF noise model and the synthetic camera FoVs, respectively. For each combination of FoV and noise level, we generate 1,000 experiments, each one containing 100 3D points randomly selected within a 5-10m radius (constrained to the camera FoV). The first camera is placed on and aligned to the origin of the world coordinate system, and the second one was randomly placed within a $[-1, 1]^3$ cube with no rotation. Note that zero relative rotation between the target camera and 3D world representation, *i.e.,* $\mathbf{R} = \mathbf{I}$, is a requirement for the 1- and 3-DoF SnP versions from Section 4.4.

Figure 5.3 shows the average relative errors for translation and rotation components using the three SnP algorithms from Section 4.4. Please note that values in the plots range differently. First and second rows present the translation errors for the 1-DoF SnP algorithm, considering as input the translation direction from ground-truth and the 8-PA estimate, respectively. The third and fourth rows present the translation error for the 3- and 12-DoF SnP versions, and the last row illustrates the rotation error for the 12-DoF SnP algorithm. Columns relate to different noise levels in decreasing order – $\kappa = 500$, $\kappa = 10,000$ and $\kappa = 1,000,000$ – corresponding to an average matching error of $3.21°$, $0.72°$, and $0.07°$, respectively. The rotation errors for the 1- and 3-DoF algorithms are

considered as those of the 8-PA, and because of the extensive analysis presented in Chapter 3, they are suppressed here.

Except for the first row, we can see that the 6-DoF pose estimates get better using any of the SnP formulations as both the noise level decreases and the FoV is expanded. When using the actual (ground truth) direction of the translation vector in the 1-DoF approach (first row), the translation scale can be estimated accurately regardless of the FoV. When the translation direction is initially estimated from the Essential matrix, the error does decrease as the FoV is enlarged. Based on these observations, we conclude that the translation error in the 1-DoF SnP algorithm is mostly bounded by the 8-PA error. Results also show that the FoV apparently presents less influence on the SnP algorithm with three free parameters (third row), and more effect on the SnP version with 12-DoF (last two rows). In particular, one can note an increased rotation error in the 12-DoF SnP algorithm for narrower FoVs, even greater than the translation error. To explain these results, we still need to investigate how matching inaccuracies affect the Procrustes problem solution (for the 12-DoF version), that is needed to ensure actual rotation matrices estimates, as mentioned in Section 4.4.

Based on the insights that come from Figure 5.3, we propose to select correspondences that are spread on the sphere surface to estimate the 6-DoF pose. Note that the current approach for selecting image points for the 6-DoF camera poses estimation is based solely on the confidence metric given in Equation (4.2). Contrarily, here, we select the best matching pair between the reference image and a given supporting image within each SSNIC superpixel of the reference image according to the joint confidence metric.

Our second experiment compares the two strategies for selecting correspondence points in the context of 6-DoF pose estimation: based solely on confidence and exploring superpixels. Table 5.1 presents the results for the average translation and rotation relative errors from the same 100 random views as in Section 4.7.1. For a fair comparison, we set the compactness $m = 0.5$ and the number of superpixels R to 0.1% the number of image points, and filter the correspondences that have low confidence. Note that the selection of poor matches may occur when a superpixel contains only bad matches – which might arise on the spherical image poles. For this experiment, instead of setting a hard threshold we reject correspondences that have less than 75% of the minimum confidence that was selected by the ranking approach explained in Section 4.7.1, which allows us to select approximately the same number of reliable matches. Kruskal-Wallis H-test (KRUSKAL; WALLIS, 1952) indicates that varying the compact-

Figure 5.3: Average FoV-dependent results for the relative translation and rotation errors. First and second rows consider the 1-DoF SnP version with ground-truth and 8-PA translation direction, respectively. Third and four rows present the average relative translation error for the 3- and 12-DoF SnP algorithms, and the last row shows the relative rotation error for the 12-DoF SnP algorithm. Columns relate to different noise levels. From left to right: $\kappa = 500$, $\kappa = 10,000$, $\kappa = 1,000,000$. Source: the author.

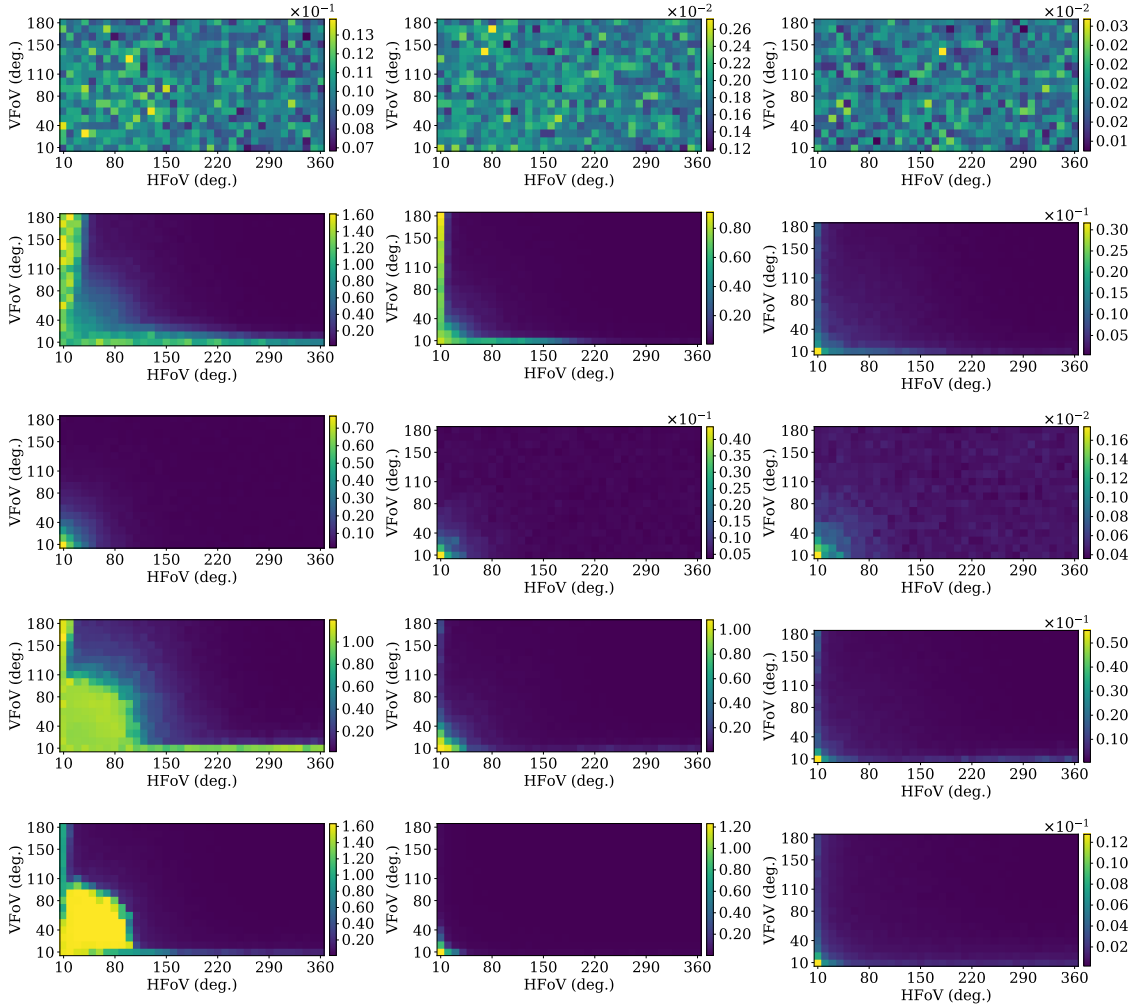


Table 5.1: Comparison of SnP algorithms using SSNIC, SSLIC or sparsified set from DeepFlow (Dense) matches, optionally using the non-linear refinement proposed in Guan and Smith (2017b). We show average relative errors for translation and rotation, and runtime (seconds), from top to bottom. Values are scaled to $\times 10^3$ for a better analysis.

Figure of merit	Dense	SSNIC	SSLIC	Refined dense	Refined SSNIC	Refined SSLIC
	Optimization of 12 parameters					
$\varepsilon_{\mathbf{t}}$	31.9 ± 27.4	34.0 ± 27.0	34.2 ± 26.7	13.7 ± 9.4	14.1 ± 9.3	14.1 ± 9.3
$\varepsilon_{\mathbf{R}}$	3.1 ± 2.5	3.1 ± 2.2	3.1 ± 2.3	1.2 ± 0.7	1.4 ± 0.6	1.4 ± 0.6
Runtime	1.7 ± 0.1	1.6 ± 0.1	1.5 ± 0.1	316.3 ± 21.5	297.3 ± 22.6	298.1 ± 22.4
	Optimization of 3 parameters					
$\varepsilon_{\mathbf{t}}$	16.8 ± 10.3	16.7 ± 9.1	16.8 ± 9.2	13.7 ± 9.4	14.1 ± 9.3	14.1 ± 9.3
$\varepsilon_{\mathbf{R}}$	1.8 ± 0.7	1.8 ± 0.7	1.8 ± 0.8	1.2 ± 0.7	1.4 ± 0.6	1.4 ± 0.6
Runtime	0.6 ± 0.1	0.6 ± 0.1	0.6 ± 0.1	287.4 ± 30.8	271.2 ± 32.4	276.1 ± 32.0
	Optimization of 1 parameter					
$\varepsilon_{\mathbf{t}}$	18.5 ± 9.1	18.7 ± 7.6	18.6 ± 7.8	13.7 ± 9.4	14.1 ± 9.3	14.1 ± 9.3
$\varepsilon_{\mathbf{R}}$	1.8 ± 0.7	1.8 ± 0.7	1.8 ± 0.8	1.2 ± 0.7	1.4 ± 0.6	1.4 ± 0.6
Runtime	0.5 ± 0.1	0.5 ± 0.1	0.5 ± 0.1	294.3 ± 27.4	251.7 ± 24.2	250.7 ± 24.2

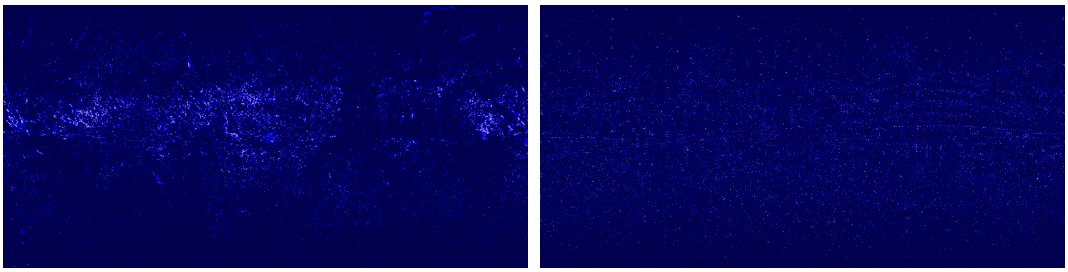
ness $m \in \{0.5, 1.0, 2.0, 5.0, \}$ does not affect the average relative errors at 1% significance. SSNIC computation with $R = 892$ and $m = 0.5$ applied to a 1280×640 equirectangular image takes about 1.85s on the machine configurations specified in Section 4.7. For comparison purposes, we also compute the rotation and translation errors using SSLIC algorithm with the same target number of superpixels and default parameters. One can notice that there is not a significant difference between the results obtained by using SSLIC or SSNIC.

Although the first experiment of this section showed that increasing FoV (*i.e.,* using spatially spread features) tends to improve the 6-DoF pose estimates, the selection strategy based on superpixels did not show a significant accuracy gain when compared to the selection of the features with the best confidence obtained from Equation (4.2). In order to better understand the difference between the two strategies, we have computed the average location of selected correspondences (on the reference image) obtained in our experiments considering the confidence-based (called unconstrained) and the superpixel-constrained ranking approaches. The results are shown in Figure 5.4, where pixels colored as dark blue represents zero occurrence. The brighter the pixel color the higher the number of occurrences in that location. As can be observed, the unconstrained approach for point selecting often yields correspondences around the whole equator line of the equirectangular images, and a reasonably wide vertical FoV. Hence, the features are already well distributed on the unit sphere, which might explain the closeness in the measurements presented in Table 5.1.

It is important to highlight that all the analyses presented in this section were based on a single dataset (*Classroom*). This synthetic scene presents several textured regions, which tend to produce good feature matching using most optical flow methods (such as

Figure 5.4: Average locations of the correspondences selected by (a) the unconstrained and (b) superpixel-constrained ranking-based approaches. Source: the author.

(a) Unconstrained selection (b) Superpixel-constrained selection



DeepFlow, used in this work). As mentioned in Section 4.7, we are not aware of other realistic datasets with multi-view captures that provide ground-truth information in terms of pose and depth. However, we believe that in scenarios with less textural information, the best matches (according to the confidence measure) might be spatially concentrated in a smaller region on the sphere, and the superpixel-based selection scheme might be helpful.

5.3 Spatially-Consistent Multi-view Calibrated Reconstruction

One of the main issues concerning the pipeline for multi-view 3D scene reconstruction presented in Chapter 4 is the lack of spatial consistency during depth estimation. The calibrated reconstruction algorithm based on the 3D error that was presented in Section 4.5 weights different camera views for each image point separately, neglecting any spatial coherence. Spatial consistency refers to the fact that regions around a point of interest from the same object are expected to present similar depth values. In Section 4.6, we explored spatial consistency *a posteriori*, by post-processing the depth maps obtained for each individual pixel using a spatial filter. In this section, we propose to explore the spatial consistency by jointly integrating the information from several pixels and cameras. More precisely, we extend the pixel-wise method presented in Section 4.5 by promoting spatially-consistent depth estimates within each superpixel.

The modified calibrated reconstruction algorithm considers R SSNIC segments extracted from the reference image of our multi-view approach, and estimates individual depth values as the previous approach but at the same time enforcing consistency within the superpixels. More precisely, since superpixel segments are expected to adhere to object boundaries (given proper parameters R and m), they should contain spatially coherent

information in semantic terms. Thus, the adopted reasoning is that, within a given segment, we should not allow large depth discontinuities, and thus we enforce the individual depth estimates subject to a consensus μ around a global depth estimate for that superpixel. Differently from the image-guided filtering approach presented in Section 4.6, the spatially-consistent multi-view calibrated reconstruction algorithm that we explore here acts *a priori*, during the depth estimation.

Here, we use the same notation as Section 4.5, briefly revised next. We refer to $\mathbf{x}_j^i$ as the projection of the i-th 3D point $\mathbf{X}^i$ onto the j-th camera centered in $\mathbf{C}_j$ that presents a weight $w_j^i \geq 0$ computed according to Equation (4.10). Also, we assume that the reference image is indexed by $j = 1$, and the 3D point to be estimated can be written as $\mathbf{X}^i = \vartheta^i \mathbf{x}_1^i$, where ϑ^i is a scalar representing its depth.

The proposed spatially-consistent formulation of the 3D error considers J observations of Q points $\mathbf{x}_j^i$, constrained to a superpixel region, for $j = 1, 2, \ldots, J$, $i = 1, 2, \ldots, Q$. The goal is to minimize a joint error function E_{3Ds} that combines individual average 3D pixel errors and the consensus the average depth μ, given by

$$E_{3Ds} = \sum_{i=1}^{Q} \frac{1}{J-1} \sum_{j=2}^{J} \left\{ w_j^i \left\| (\vartheta^i \mathbf{x}_1^i - \mathbf{C}_j)^\top (\vartheta^i \mathbf{x}_1^i - \mathbf{C}_j) - [\mathbf{x}_j^{i\top} (\vartheta^i \mathbf{x}_1^i - \mathbf{C}_j)]^2 \right\| \right\} + \lambda \sum_{i=1}^{Q} (\vartheta^i - \mu)^2, \tag{5.4}$$

where $\lambda > 0$ controls the weight of the regularization term. When $\lambda \to 0$, Equation (5.4) solves the same problem as Equation (4.5), considering each point $i = 1, 2, \ldots, Q$ individually. As λ increases, the depth distribution within the superpixel tends to be closer to the consensus μ. In fact, the limit solution (as $\lambda \to +\infty$) is $\vartheta^1 = \vartheta^2 = \cdots = \vartheta^Q = \mu$, yielding a constant-depth representation of the superpixel.

Solving for the unknowns $\vartheta^i, i = 1, 2, \ldots, Q$, and μ can be done through the following sparse linear system (after multiplication by $J - 1$):

$$\begin{bmatrix} a^1 + 2\lambda(J-1) & 0 & 0 & \cdots & -2\lambda(J-1) \\ 0 & a^2 + 2\lambda(J-1) & 0 & \cdots & -2\lambda(J-1) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & a^Q + 2\lambda(J-1) & -2\lambda(J-1) \\ 2\lambda(J-1) & 2\lambda(J-1) & \ldots & 2\lambda(J-1) & -2Q\lambda(J-1) \end{bmatrix} \begin{bmatrix} \vartheta^1 \\ \vartheta^2 \\ \vdots \\ \vartheta^Q \\ \mu \end{bmatrix} = \begin{bmatrix} b^1 \\ b^2 \\ \vdots \\ b^Q \\ 0 \end{bmatrix}, \tag{5.5}$$

where $a^i = \sum_{j=2}^{J} w_j^i \left[1 - (\mathbf{x}_1^{i\top} \mathbf{x}_j^i)^2\right]$ and $b^i = \sum_{j=2}^{J} w_j^i \left[\mathbf{x}_1^{i\top} \mathbf{C}_j - (\mathbf{x}_1^{i\top} \mathbf{x}_j^i)(\mathbf{x}_j^{i\top} \mathbf{C}_j)\right]$. Note that the coefficient matrix is almost a diagonal matrix, with the last row and column with non-zero elements. This matrix structure allows to solve this linear system efficiently using a simplified Gaussian elimination method, in a complexity linear with respect to the

number of pixels Q within the superpixel.

For larger λ, this approach leads to a piece-wise constant depth representation of the scene, which might be adequate when the superpixels are small. If larger superpixels are used, another possibility would be to enforce a planar representation for each segment, which extends the concept of piece-wise depths. However, optimizing such alternative formulation leads to a non-linear system that cannot be solved with the same efficiency as explored in Equation (5.5).

Next, we evaluate the results of the proposed spatially-consistent calibrated 3D reconstruction algorithm. Because there is not a trustworthy way for mimicking superpixel segmentation on "random world" simulations as in Section 4.7.2, we skip the analysis of synthetic matching and focus on the experiments relying on real matching performed by DeepFlow. For this aim, we follow the same experimental setup as explained in Section 4.7.3, and analyze the results achieved by the formulation in Equation (5.5) in terms of 3D reconstruction error considering the whole pipeline as described throughout Chapter 4.

The accuracy of the proposed spatially-consistent multi-view calibrated reconstruction algorithm depends basically on three extra parameters that did not exist in the point-wise formulation. These parameters are the number of SSNIC superpixels R, SSNIC compactness factor m, and the regularization parameter λ from Equation (5.5). Although there is an intricate relation between these three parameters, the visual analysis presented next aims to evaluate the impact of changing each parameter independently.

Intuitively, the larger the value of R the finer the image representation. However, in the limit case, we may end up with the point-wise 3D reconstruction algorithm, and the spatial consistency that we are trying to impose is compromised. Recall that if $\lambda \to 0$, then we also will obtain the same results as the in point-wise formulation regardless of the SSNIC parameters R and m. For the 3D reconstruction problem, boundary adherence is more important than superpixels regularity, since the spatial consistency assumes that pixels within a superpixel belong to the same object. Thus, we recommend to select a small value for m. Figures 5.6, 5.7 and 5.8 depict the effect of changing the values of R, m, and λ individually. These results are shown on image RoIs selected to highlight the desired effects, and they are shown in Figure 5.5.

Figure 5.6 illustrates the effect of increasing the number of superpixel regions R while keeping the remaining parameters fixed. For visualization purposes, we set $m = 0.1$ and $\lambda = 10^{-2}$. As expected, a small value for R (*e.g.* 0.1% of the image points, or

Figure 5.5: Reference image indicating the RoIs used in Figures 5.6, 5.7, and 5.8, in blue, green, and red, respectively. Image represented in gray-scale for better visualization. Source: the author.

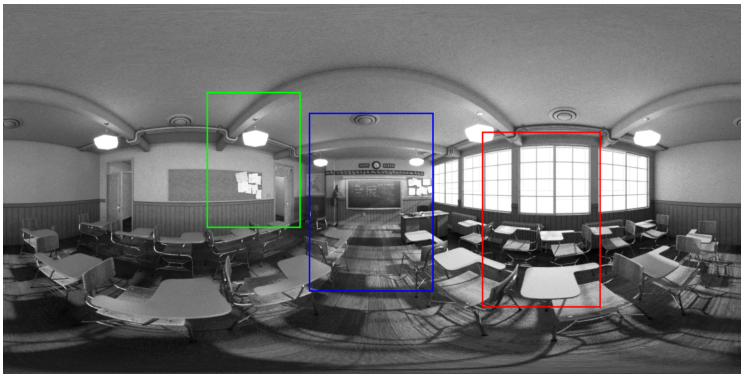
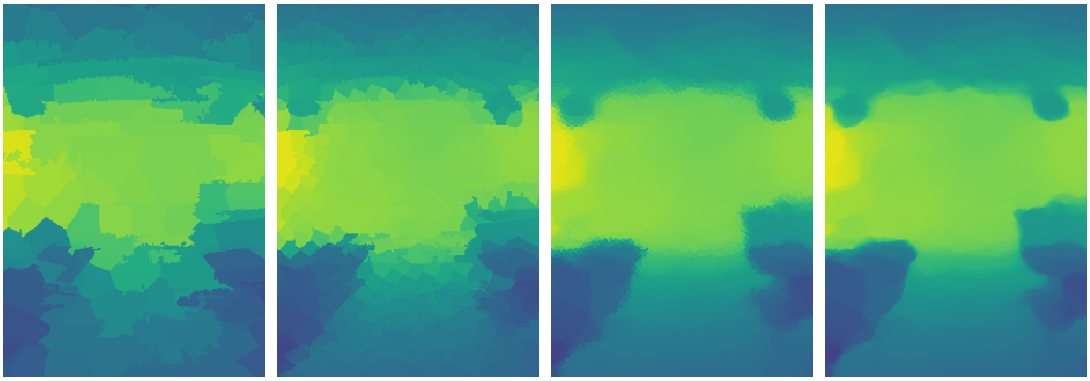


Figure 5.6: Impact of the number of SSNIC superpixels R on the spatially-consistent calibrated 3D reconstruction algorithm. Results focused on a RoI for better visualization. Source: the author.

(a) $R = 819$ (b) $R = 4,096$ (c) $R = 40,960$ (d) Pixel-wise



$R = 819$) will generate regions that mix information from several scene objects and the underlying 3D reconstruction results may be poor. On the other hand, using a larger value for R, set as 5% of the number of image pixels ($R = 40,960$), leads to visually accurate results without apparent jagged artifacts.

The impact of varying the SSNIC compactness factor m while keeping the other parameters fixed is illustrated in Figure 5.7. We set $R = 4,096$ and $\lambda = 10^{-2}$ for visualization purposes only. Recall that small values for m encourage border adherence, as explained in Section 5.1.1. This phenomenon is inherited from the superpixel model formulation, and appears in the same format in our 3D reconstruction pipeline (confer the lamp contour in Figure 5.7). In the explored 3D reconstruction context, border adherence is preferred than segments regularity.

Finally, Figure 5.8 shows the depth estimates obtained by varying parameter λ only. For this example, we set $R = 4,096$ and $m = 0.1$. As pointed out before, the

Figure 5.7: Impact of the SSNIC compactness m on the spatially-consistent calibrated 3D reconstruction algorithm. Results focused on a RoI for better visualization. Source: the author.

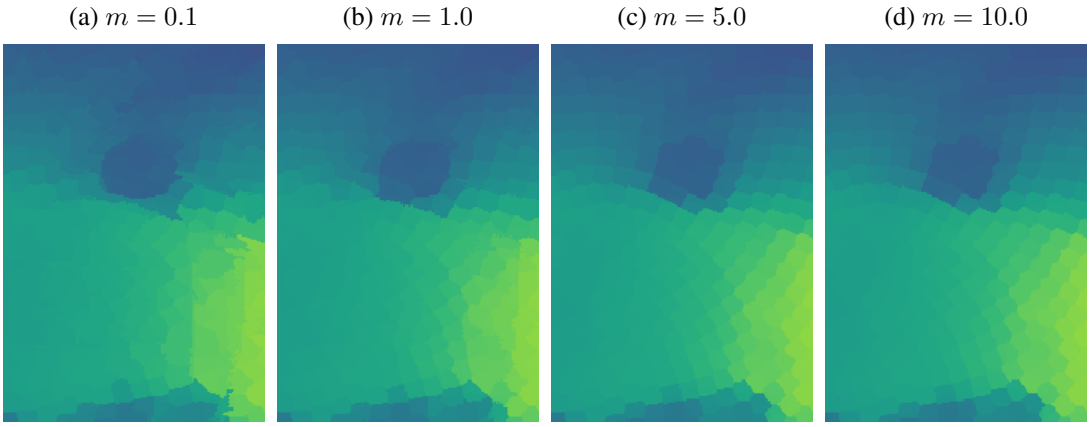
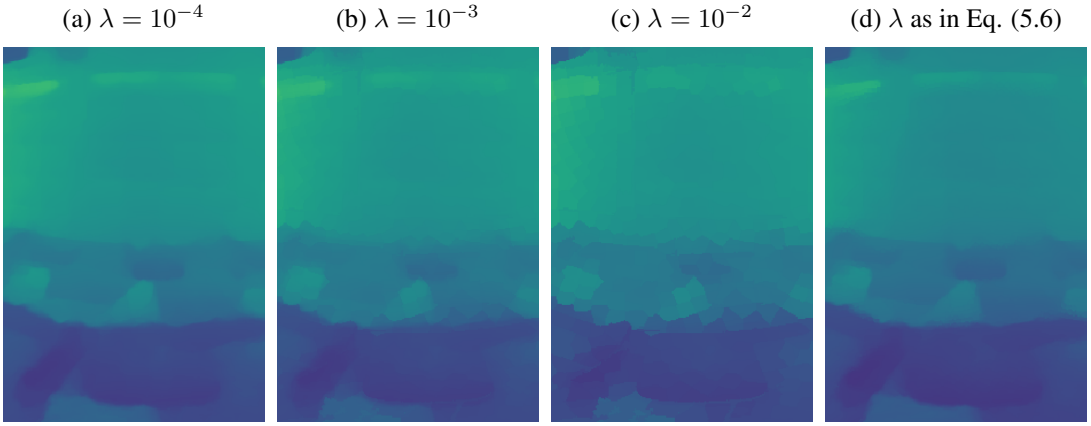


Figure 5.8: Impact of the parameter λ on the spatially-consistent calibrated 3D reconstruction algorithm. Results focused on a RoI for better visualization. Source: the author.



spatially-consistent depth estimation procedure approaches the pixel-wise one as λ tends to zero. On the other hand, the consensus depth tends to dominate when parameter λ increases, generating superpixel segments with constant depth.

In Figures 5.8 (a)–(c), the value of λ is fixed for all superpixels. However, we believe that λ should be small for textured segments, because of potential mixture of object information, and thus point-wise depth estimation should be encouraged. Smoother regions should present a consistent depth estimate regardless of the other parameters, and thus a large value for λ is preferred. We promote the desired behavior by empirically setting λ as the squared inverse variance of the luminance color channel of each superpixel

Table 5.2: Average relative error of the 3D scene reconstruction using the pixel-wise approach and the spatially-consistent algorithm based on SSLIC and SSNIC segments. Values are scaled to 10 for better analysis.

Number of cameras	Method					
			SSNIC		SSLIC	
	Pixel-wise	Pixel-wise + DT filter	Spat.-cons.	Spat.-cons. + DT filter	Spat.-cons.	Spat.-cons. + DT filter
3	1.515 ± 0.321	1.469 ± 0.279	1.452 ± 0.262	1.410 ± 0.253	1.457 ± 0.271	1.443 ± 0.259
5	1.277 ± 0.346	1.201 ± 0.283	1.161 ± 0.275	1.139 ± 0.267	1.167 ± 0.281	1.141 ± 0.271
7	1.111 ± 0.189	1.074 ± 0.177	1.001 ± 0.149	0.996 ± 0.144	1.021 ± 0.151	1.001 ± 0.149
9	1.051 ± 0.186	1.020 ± 0.172	0.993 ± 0.158	0.978 ± 0.153	0.997 ± 0.162	0.982 ± 0.156

segment, given by

$$\lambda = \lambda(\{c_l^1, c_l^2, \ldots, c_l^Q\}) = \frac{1}{\sigma\Big(\{c_l^1, c_l^2, \ldots, c_l^Q\}\Big)^2 + \epsilon}, \tag{5.6}$$

where $c_l^1, c_l^2, \ldots, c_l^Q$ are the luminances of the Q pixels within a given segment, and ϵ is a very small constant that avoids division by zero in cases of totally homogeneous regions. In our further experiments, we set R to 5% the number of superpixels and $m = 0.1$. Figure 5.8 (d) illustrates the behavior of λ according to Equation (5.6).

Table 5.2 presents the average relative error and standard deviation for $J \in \{3, 5, 7, 9\}$, considering 30 simulations using the same random images from Section 4.7.3. One can note that the results achieved by imposing spatial consistency during depth estimation (using either SSLIC or SSNIC) improves the point-wise results for all the considered values for J. Note that although SSLIC and SSNIC leads to similar 3D errors, SSNIC is much faster than SSLIC, and thus it is adopted in this work as part of our main pipeline. Using SSLIC instead of SSNIC for computing 40,960 superpixels would increase the pipeline runtime about 40%.

Also, one can see that the spatially-consistent approach achieved slightly better average results than the post-processing (pixel-wise + DT filter) option, described in Chapter 4. Although both the *a priori* and *a posteriori* approaches are used for enforcing spatial consistency, they act in different stages of the proposed pipeline, so that they can be applied together. In our tests, a small additional gain was obtained by applying the guided DT filter after estimating the depth using the superpixel-based calibrated 3D reconstruction algorithm.

We observed that the spatially-consistent version of the calibrated reconstruction algorithm presented in this section tends to produce more coherent results in the presence of "few" outliers per superpixel than the point-wise strategy. This is probably the main cause of the reduction in the relative 3D error shown in Table 5.2. In particular, our

SSNIC-based approach tends to produce better estimates near the poles of the equirectangular images – where the segments are larger in terms of pixel count, and thus can better account for few outliers. Figure 5.9 exemplifies these results when the recovered 3D geometries from the *Classroom* scene are seen as point clouds (lateral and top-down views are provided). These results indicate that the spatially-consistent estimates present a more regular shape (cuboid-like), despite the apparent artifacts (clustered depth values) in Figures 5.9(b) and (e).

Finally, Figure 5.10 illustrates the obtained results when applying the spatially-consistent calibrated 3D reconstruction algorithm after being post-processed with DT filter on the same real datasets described in Section 4.7.3. As reported in Table 5.1 and shown in Figure 5.9, there are only subtle differences that accounts for a final average accuracy improvement. All the results consider the same parameters as specified before.

Figure 5.9: Example of a typical 3D reconstruction from the proposed method using $J = 3$ cameras. Point clouds shown in external (a), (b), (c) side and (d), (e), (f) top views. Results for (a), (d) pixel-wise (+ DT filter) and spatially-consistent calibrated 3D reconstruction algorithms (b), (e) before and (c), (f) after DT filtering. Source: the author.

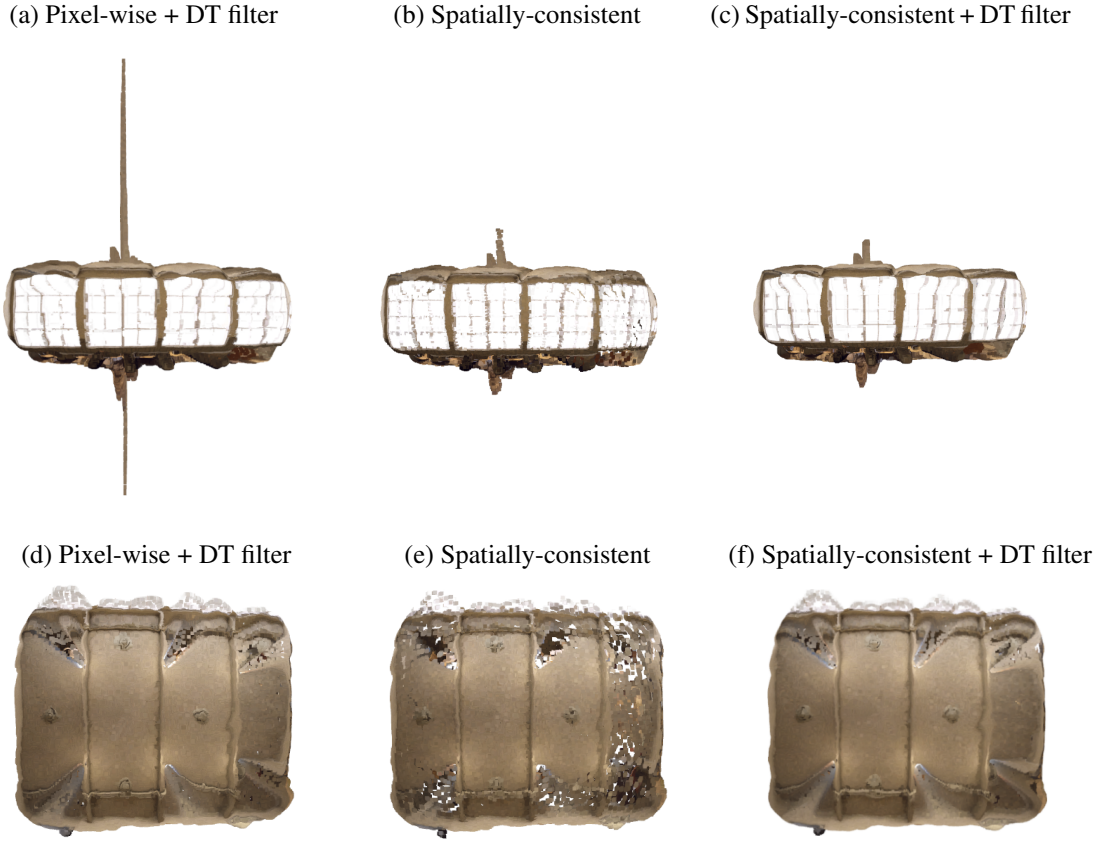
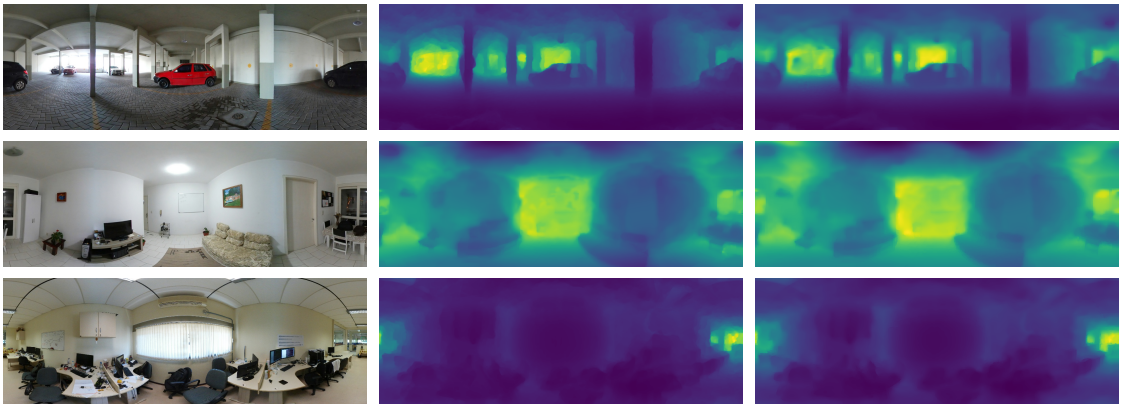


Figure 5.10: Examples of results produced by our approach. From left to right: the reference image and estimated depth maps using the point-wise and the spatially-consistency approaches, respectively. The depth estimates are post-processed using DT filter. Source: the author.



5.4 Conclusions of the Chapter

In this chapter, we presented a novel superpixel algorithm suited for omnidirectional images that encourages image segments to be uniformly distributed on the sphere surface, named spherical simple non-iterative clustering (SSNIC). Then, we conducted an experimental analysis based on synthetic feature matching for determining if, as in the Epipolar matrix estimation case, the linear solutions for spherical-n-point problem are also affected by the features distribution. Based on the obtained (positive) results, we proposed to select a representative pair of correspondences from each SSNIC superpixel for enforcing nearly-uniform spatial distribution, and assessed this approach against an unconstrained selection of the best feature set using our joint photometric-geometric confidence formulation. The obtained results indicate that the unconstrained selection of points tended to cover a full horizontal FoV of the images from our dataset, so that we could not improve its performance by enforcing superpixel-based selection of features.

Finally, we have proposed to enforce spatial consistency *a priori*, *i.e.,* during depth estimation, using SSNIC segments. This approach differs from the *a posteriori* solution adopted in the previous chapter, which uses an image guided filter in a post-processing step. Our results indicate that this approach can be effective for removing incoherent matches within semantically grouped regions, and it presented smaller average errors when estimating the 3D scene geometry.

6 DENSE 3D RECONSTRUCTION FROM SINGLE SPHERICAL IMAGES

In this chapter we provide a *framework* for inferring depth from a single spherical image, in such a way that it is generic and can be coupled to any single-image depth technique, such as (LIU et al., 2016; EIGEN; FERGUS, 2015; EIGEN; PUHRSCH; FERGUS, 2014; GODARD; Mac Aodha; BROSTOW, 2017), going in the direction of what was discussed in Su and Grauman (2017). Note that because the depth estimation task from a single image is an ill-posed problem, the inference process based solely on trained data may fail depending on several aspects related to the discrepancies between training and test data, such as (presence or absence of) contextual information, aspect ratio, illumination changes, camera setup, image resolution, etc. We do not recommend using a single image for depth inference because there are no geometric constraints to base the procedure so that the framework we introduce here is intended to be used only in *extreme* cases, where it is not possible to capture more than one image.

The pipeline of our approach, shown in Figure 6.1, starts by extracting multiple *overlapping* tangent planar projections with smaller FoVs from the spherical image. Afterward, we apply a monocular planar depth estimation algorithm to each one of the planar patches and back-project the associated depth maps to the adequate locations on the sphere. In a third moment, we minimize the depth discrepancies along the pairwise intersections on the sphere and perform alpha-blending to obtain the final spherical depth map. The methodology presented in this chapter was published in Silveira, Dal'aqua and Jung (2018).

Figure 6.1: Pipeline of the proposed approach for inferring depth from a single spherical image. Source: the author.

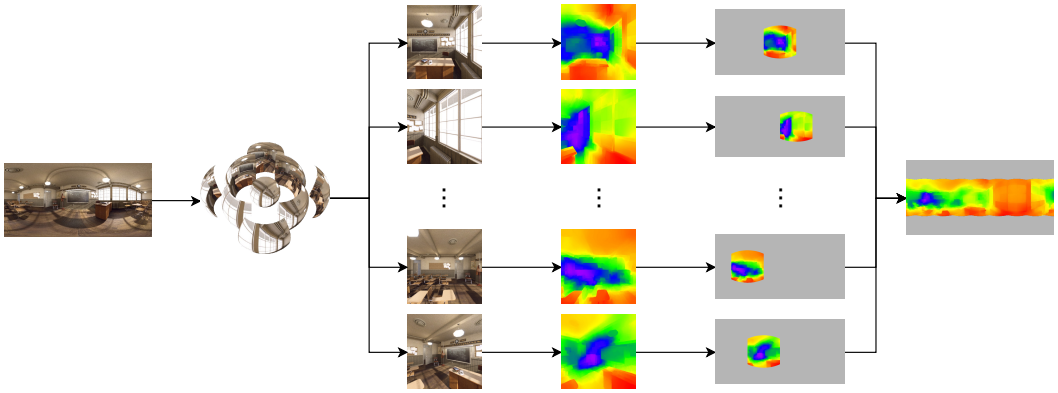
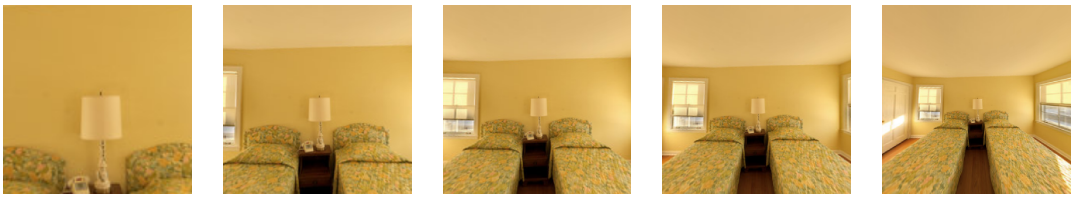


Figure 6.2: Planar projections of spherical sections with different FoV θ values. Source: the author.

(a) $\theta = 45°$ (b) $\theta = 75°$ (c) $\theta = 90°$ (d) $\theta = 100°$ (e) $\theta = 120°$



6.1 Sectioning the Sphere and Planar Projection

The first part of our method consists of partitioning the sphere into N_s angular sections with horizontal FOV $\theta = 360°/N_s$ (so that each section contains an approximately "perspectively-projected RoI"), and then applying planar monocular depth estimation algorithms to each RoI. Aiming to enforce the coherence between the estimates, we select $2N_s$ sections with horizontal FOV θ and angular displacement of $\theta/2$ so that each section overlaps approximately a half with its two adjacent sections. The sections in south and north poles[1] are discarded here, since their planar version represent uncommon conditions in the CNN training sets, causing incoherent depth estimates. Furthermore, these sections typically relate to the floor and ceiling of the scene, as mentioned in Chapter 4.

Given an angular section with horizontal FoV θ, and for simplicity vertical FoV $\phi = \theta$, and also the sphere projection center (θ_C, ϕ_C), we project the information within this section to a square planar image. This is accomplished by selecting the intensity values located at the positions (θ, ϕ) on the surface of the sphere that pass through the tangent plane to it and (θ_C, ϕ_C). Intensities in real-valued pixel positions of the source image are obtained via bilinear interpolation. Figure 6.2 illustrates different choices for the FoV $\theta = \{45°, 75°, 90°, 100°, 120°\}$ using an image from the SUN360 dataset (XIAO et al., 2012).

6.2 Planar Monocular Depth Estimation

Since our aim is to provide a framework so that any planar monocular depth estimation method can be applied, we decided to use them as black boxes. Although the

[1] It is fair to assume that the equator line of most equirectangular images is aligned to the world's horizon. Moreover, a derotation pre-processing (PATHAK et al., 2016b) might lead any spherical image to this condition.

authors in Liu et al. (2016), Eigen and Fergus (2015), Eigen, Puhrsch and Fergus (2014), Godard, Mac Aodha and Brostow (2017) made their code available, only the first and the last approaches are considered here. We consider indoor trained models, namely the “NYU v2 dataset" from Liu et al. (2016) and the “Eigen split" dataset from Godard, Mac Aodha and Brostow (2017), according to notations of the respective papers. We did not use the other two methods, namely Eigen and Fergus (2015), Eigen, Puhrsch and Fergus (2014), because of severe aspect ratio and resolution constraints.

Liu et al. (2016) propose to formulate the depth estimation as a deep continuous conditional random fields (CRF) learning problem, without depending on geometric priors or any extra information. The authors jointly explore CNNs and a continuous CRF, which they call deep convolutional neural field, by solving a maximum a posteriori inference problem. Their method takes into account unary and pairwise potentials for encouraging, respectively, to regress the depth values and the similarity in neighboring superpixels, similarly to Eigen and Fergus (2015).

Unlike the other works (LIU et al., 2016; EIGEN; PUHRSCH; FERGUS, 2014; EIGEN; FERGUS, 2015), the unsupervised method from Godard, Mac Aodha and Brostow (2017) does not need the ground-truth depth maps for training their model: instead, they use calibrated and rectified stereo pairs. Godard, Mac Aodha and Brostow (2017) propose a fully convolutional model that exploits epipolar geometry constraints to learn how to produce depth in such a way that the difference of the left image projected to the right camera and the actual right view (and vice versa) are minimized. In fact, the loss function of their architecture tries to simultaneously optimize an appearance matching cost, a smoothness term on the estimated disparities, and a left-right consistency checking term. For testing, they use only a single image.

6.3 Reprojecting and Fusing Planes to the Sphere

The process described in Section 6.1 is applied in the inverse direction to each of the individually estimated depth maps, reprojecting them to the locations where the corresponding color patches where extracted from. More precisely, let $\mathfrak{I}$ be the input equirectangular image, and $\mathbf{S}_k$ and $\mathbf{D}_k$ denote the reprojected patches and the corresponding estimated depth images, for $k = 1, \ldots, N_s$. Each pair of adjacent reprojections $\mathbf{S}_k$ and $\mathbf{S}_{k+1}$ (and the respective depth maps) overlap, as well as $\mathbf{S}_{N_s}$ and $\mathbf{S}_1$ (circular cover).

At each intersection of adjacent reprojections $\mathbf{S}_k$ and $\mathbf{S}_{k+1}$, there are two indepen-

dently estimated depth maps (denoted by $\mathbf{D}_L^k$ and $\mathbf{D}_R^k$). Also, let $\mathcal{P}_k$ be the set of pixel positions (i, j) in the overlapping region (in the equirectangular domain), and $\mathcal{R}_k$ the set containing the row indices of those pixels.

Our goal here is to smoothly stitch these two depth maps for each overlapping region $\mathcal{P}_k$, and at the same time capture the structures present in the image. For that purpose, we want to find weights $w_L^k(i)$ and $w_R^k(i)$ for each row $i \in \mathcal{R}_k$ that minimize the depth cost error C_D^k along the overlap region, given by

$$C_D^k = \sum_{(i,j)\in\mathcal{P}_k} \left[w_L^k(i) D_L^k(i,j) - w_R^k(i) D_R^k(i,j)\right]^2. \tag{6.1}$$

However, selecting $\mathbf{w}_L^k$ and $\mathbf{w}_R^k$ independently for each row may lead to undesired discontinuities on adjacent rows of the resulting depth map. To cope with this issue, we add a regularization cost term C_R^k given by

$$C_R^k = \sum_{(i,j)\in\mathcal{P}_k} \chi(i,j) \left\{ \left[w_L^k(i) - w_L^k(i+1)\right]^2 + \left[w_R^k(i) - w_R^k(i+1)\right]^2 \right\}, \tag{6.2}$$

where $\chi(i, j)$ is a weight that should be large when neighboring pixels (i, j) and $(i+1, j)$ are similar, and small otherwise. We propose to use

$$\chi(i,j) = \begin{cases} \nu, \text{ if } \|I(i,j) - I(i+1,j)\| < \tau \\ \upsilon, \text{ otherwise} \end{cases}, \tag{6.3}$$

where τ defines a color similarity threshold, and $\upsilon < \nu$ are the regularization weights. We have tested continuous functions for $\chi(\cdot)$ (with monotonic decay w.r.t. to $\|I(i, j) - I(i + 1, j)\|$), but results were similar.

Thus, the final cost function is given by $C_T^k = C_D^k + C_R^k$ for each overlapping region $\mathcal{P}_k$. To avoid the trivial solution, we actually solve the homogeneous linear systems $\nabla C_T^k = \mathbf{0}$ subject to $\|\mathbf{w}^k\| = 1$, where $\mathbf{w}^k$ contains all the unknowns (weights). Similarly to the 8-PA, the solution is obtained by the least right singular vector.

Since each section $\mathbf{S}_k$ overlaps with its two adjacent sections, it also presents two sets of row-wise weights $\mathbf{w}_R^k$ and $\mathbf{w}_L^{k-1}$ on its right and left portions, respectively. A direct multiplication of those weights by the corresponding depth values $\mathbf{D}_k$ do minimize the depth differences along adjacent overlapping sections according to Equation (6.2), but might generate discontinuities within the section (along the boundaries of the overlapping

regions). To cope with this issue, a row-wise linear interpolation scheme is applied to generate a continuous weight map for each depth map $\mathbf{D}_k$ related to $\mathbf{S}_k$.

The final step in the proposed approach is to generate a single depth value for the overlapping regions $\mathcal{P}_k$, since the depth values $\mathbf{D}_L^k$ and $\mathbf{D}_R^k$ (consider those values *after* the multiplication by the weights) will still present discrepancies. In this work, we used alpha-blending along the rows to obtain the final depth value $\Gamma(i,j)$, given by

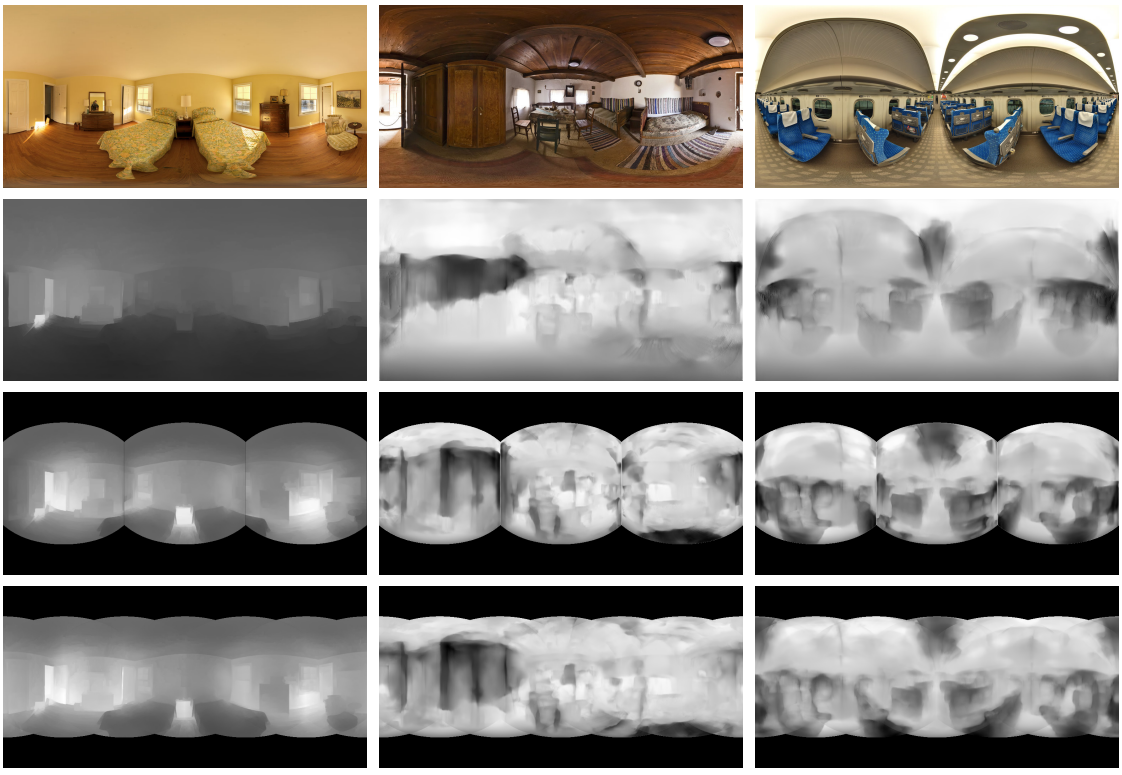
$$\Gamma(i,j) = (1 - \iota_i(j))\, D_L^k(i,j) + \iota_i(j) D_R^k(i,j) \tag{6.4}$$

for

$$\iota_i(j) = (j - j_L^i)/(j_R^i - j_L^i), \tag{6.5}$$

where j_L^i and j_R^i are the lowest and the highest column values at each row i, respectively. The depth values on non-overlapping regions have only one depth value, which is copied to the final depth map $\mathbf{\Gamma}$.

Figure 6.3: From top to bottom: color image, and depth estimates from the application on the equirectangular image, and disjoint and overlapping sections with $\theta = 120°$. From left to right: results from (LIU et al., 2016), and VGG and ResNet-based models in (GODARD; Mac Aodha; BROSTOW, 2017). Source: the author.



6.4 Experimental Results on Monocular Spherical Depth Estimation

In general, selecting smaller values for the FoV θ presents less distortion on the tangent planes, but the resulting images do not contain enough contextual information used by the baseline CNNs. In this paper, we provide the results for $\theta = 90^\circ$ and $\theta = 120^\circ$. The former is often used to project the sphere to a cube-map representation, which allows traditional algorithms to be applied because of the attenuated distortions (KOPF, 2016). The latter is selected because a wider FoV is essential in providing more contextual information (ZHANG et al., 2014). We found that $\theta = 120^\circ$ as a good compromise between distortion and contextual information. The other free parameters are $\nu = 200$, $\upsilon = 50$ and τ is 10% the maximum RGB color difference, set experimentally.

Due to the lack of spherical image databases with ground-truth for depth, we evaluate our method quantitatively by considering spherical captures from the realistic *Classroom* scene, as in the other chapters. To the best of our knowledge, we were the first authors to propose an alternative for estimating depth from spherical images, although some works had already focused on 3D layout extraction from a single panorama a few years before (ZHANG et al., 2014). Up to the paper submission (SILVEIRA; DAL'AQUA; JUNG, 2018), we were not aware of Yang, Liu and Kang (2018), Zioulis et al. (2018), and thus we did not compare the results with none of them. We highlight that both competing strategies are much more complex than our approach and may lead to better results.

The spherical depth maps obtained by using our approach are compared to the ground-truth by the scale-invariant mean squared error (SIMSE), as done in Eigen, Puhrsch and Fergus (2014) in the context of depth estimation of planar images. Three baseline CNN models are considered in our tests, namely the method proposed by Liu et al. (2016), as well as the VGG and ResNet-based models from Godard, Mac Aodha and Brostow (2017). All configurable parameters were kept unaltered from the original papers. Table 6.1 presents the average SIMSE values for three test cases: depth estimation on the equirectangular images, on disjoint sections of the sphere, and by the proposed framework with $\theta = 90^\circ$ and $\theta = 120^\circ$. The results are computed discounting the values in the south and north poles depending on the selected θ values for a fair comparison. Note that the best results are achieved by the proposed scheme regardless of the chosen planar depth estimator. Moreover, applying planar methods to disjoint sections does not guarantee better results than if they were directly applied to the full images, as can be seen from the results of VGG and ResNet. Also, smaller errors were found for $\theta = 120^\circ$, indicating that

Table 6.1: Average SIMSE values.

Mode	Method	$\theta = 90°$	$\theta = 120°$
Full image	Liu et al. (2016)	13.177 ± 4.720	10.202 ± 3.804
	Godard, Mac Aodha and Brostow (2017) (VGG)	13.865 ± 4.619	10.828 ± 3.713
	Godard, Mac Aodha and Brostow (2017) (ResNet)	13.722 ± 4.603	10.684 ± 3.669
Disjoint sections	Liu et al. (2016)	13.030 ± 4.856	10.306 ± 4.007
	Godard, Mac Aodha and Brostow (2017) (VGG)	18.373 ± 5.972	11.235 ± 3.702
	Godard, Mac Aodha and Brostow (2017) (ResNet)	17.948 ± 5.834	11.022 ± 3.699
Overlap. sections	Liu et al. (2016)	12.394 ± 4.704	9.567 ± 3.768
	Godard, Mac Aodha and Brostow (2017) (VGG)	12.634 ± 4.478	10.015 ± 3.486
	Godard, Mac Aodha and Brostow (2017) (ResNet)	12.315 ± 4.444	9.802 ± 3.451

the CNNs can deal with small distortions since they have enough contextual information.

We have also applied the proposed approach to real indoor scenes from the SUN360 database (ZHANG et al., 2014). Figure 6.3 illustrates the application of the proposed framework in three different scenarios, each one submitted the three test cases guided by one of the considered CNN-based methods. Note that our framework is able to resolve the relative depth from several structures (the beds, doors, seats, etc.) that are not clearly distinguished by the other approaches. In particular, several objects are not found when the planar methods are directly applied to the equirectangular images, and globally inconsistent values are exhibited if they are applied to disjoint sections. However, we emphasize that the results from our approach are completely dependent on those of the planar monocular depth estimation methods, as can be clearly seen in Figure 6.3.

6.5 Conclusions of the Chapter

In this chapter, we proposed a framework for inferring depth from a single spherical image based on overlapping planar projections, which can benefit from any depth estimation approach suited for planar images (baseline method). We performed tests by plugging three state-of-the-art CNN-based baseline algorithms, and compared the results on both synthetic and real spherical images. Quantitative results indicated that our method outperforms two common strategies for adapting planar methods to the spherical domain: the application of a planar method (i) directly to equirectangular images or (ii) to multiple disjoint planar sections, mapped back to the sphere.

7 FINAL REMARKS

7.1 Conclusions

In this Dissertation, we have addressed the problem of dense 3D reconstruction using two or more spherical images, focusing our analysis on indoor scenes. We have also presented a framework for inferring depth from a single spherical image.

Our multi-view approach considers that the input images are temporally unordered and the 3D camera poses are unknown. Given a reference view, our method first estimates the 5-DoF camera pose of an additional camera, and then creates an initial 3D reconstruction based on the stereo pair. From this intermediate scene representation, the proposed approach estimates the full 6-DoF camera poses of all remaining cameras (if more than two are available), which are used to produce a dense depth map fully registered to the reference image. Post-processing techniques such as image-guided filtering are applied to further refine the results, and, thanks to the color-plus-depth representation, traditional DIBR techniques can be explored for providing 3-DoF+ for VR applications. We further investigate and conclude that adding spatial constraints via image oversegmentation (achieved by adapting a superpixel approach to the spherical domain) can lead to better results in terms depth estimates.

Aiming to support our choices in the main pipeline, we have theoretically and experimentally shown that the linear 8-PA is capable of producing more accurate estimates for the Essential matrix when using wide FoV image pairs compared to typical perspective/pinhole-based cameras, which present much narrower FoV than spherical cameras. Although being a self-contained result, these conclusions allow us to suppress the application of the popular but expensive iterative non-linear refinement approaches based on BA and still obtain compelling results in our multi-view method.

Additionally, we have introduced a framework for adapting any arbitrary algorithm for inferring depth from a single perspective image to the spherical domain. As such, the performance of our proposal is limited by those baseline techniques, and, because the ill-posedness nature of the monocular depth estimation problem, we do recommend relying instead on multi-view geometric constraints for achieving better results.

Next we revise the hypotheses listed in Section 1.3, and discuss their validity based on the results presented along this Dissertation.

Hypothesis (i) says that “using features matched over the sphere can yield bet-

ter pose estimation results than using narrow-FoV localized features". We provide both theoretical and experimental results in Sections 3.2 and 3.5 that testify that Hypothesis (i) is indeed true at least when considering the classic 8-PA. Some insights are given in Section 5.2 for the case of the SnP problem.

Our second hypothesis is that "it is possible to obtain progressively more accurate 3D scene reconstructions by adding more spherical images". The results presented in Section 4.7 experimentally corroborate Hypothesis (ii), considering both synthetic and real feature matching, when assessing the multi-view 3D calibrated reconstruction algorithm and the whole pipeline from Chapter 4.

Hypothesis (iii) claims that "with a fixed number of views, a proper positioning of the cameras can improve 3D scene reconstruction". Section 4.5 presented the mathematical formulation of our calibrated 3D reconstruction algorithm for the multi-view problem that exposes the problem of the epipoles in spherical imaging. Section 4.7.5 discussed the problem of having nearly straight-aligned image captures, and exhibited some examples indicating that our third hypothesis is indeed true.

Hypothesis (iv) states that "standard optical flow algorithms can be used to obtain dense features for estimating both the 3D camera pose and dense geometry, provided the camera views are close enough". In Section 4.3 we explored a dense optical flow suited for planar images in the context of omnidirectional imaging. We proposed to compute a confidence metric for each matching, and used both the correspondences and confidences for pose and 3D geometry estimation (confer Sections 4.7.1 and 4.7.3), validating Hypothesis (iv). Measuring how far the captures can be to each other remains as future work.

Our fifth hypothesis is that "it is possible to weight each one of the matched features so that a better 3D scene reconstruction is obtained than if all of them contribute the same". Sections 4.5 and 5.3 explored ways to encourage good matches and penalize bad ones, with and without aggregating nearby spatial information. Although we have presented the results for one out of many possible weighting schemes, we could show that Hypothesis (v) is indeed true.

Finally, Hypothesis (vi) asserts that "a domain-adapted superpixel algorithm can be used to improve the average 6-DoF camera poses and depth estimates in a multi-view 3D reconstruction pipeline". We start by proposing the spherical simple non-iterative clustering, as described in Section 5.1, and use it for the referred tasks. Although we have concluded that the spatial distribution of the features impacts on the 6-DoF pose

estimates using synthetic matching, we could not prove these results in practice when using our superpixel algorithm for selecting spread features when compared to a subset of the best matches (according to the confidence metric). The second part of Hypothesis (vi) was validated in Section 5.3.

7.2 Future Works

Spherical imaging is changing how traditional problems of image processing and computer vision areas are tackled, offering a number of advantages regarding the classic pinhole-based scene capture model. In this Dissertation, we have presented some analyses and solutions in this emerging field, focusing on the 3D geometry recovery problem. However, there is still plenty of room for improvements. In the following, we list some of our intentions for future work:

(i) to explore other spatial/geometric constraints in our multi-view 3D reconstruction method, as matching layout information extracted from the reference and supporting images via the approaches in Zou et al. (2018), Yang et al. (2019);

(ii) to better explore the potential of SSNIC in terms of generic segmentation metrics, and compare it to the literature. We also intend to extend a recent graph-based superpixel algorithm (WEI et al., 2018) to the spherical domain, and explore it in pure segmentation applications and in our multi-view 3D reconstruction method. Graph-based approaches present the advantage of being independent of data topology;

(iii) to perform visualization tests using the proposed approach for 3-DoF+ with users equipped with HMD VR headsets, and compute subjective metrics from the users feedback. We will do our best for comparing our results with other approaches;

(iv) to explore techniques like the kernel transformer network (SU; GRAUMAN, 2019) for adapting large displacement and dense CNN-based approaches for computing the optical flow in perspective images, such as DeepFlow (WEINZAEPFEL et al., 2013), to the spherical domain;

(v) to extend our perturbation analysis relating the feature spreading and noise to other linear solutions involving camera pose and 3D geometry recovery, such as the DLT and 1-, 3- and 12-DoF SnP problems;

(vi) to compile a systematic literature review on the 3D geometry/depth estimation problem from one, two or multiple spherical images and publish a survey journal paper.

7.3 Published Papers

Below, we list the published papers related to this Dissertation.

(i) Silveira, T. L. T. and Jung, C. R.; *Dense 3D Scene Reconstruction from Multiple Spherical Images for 3-DoF+ VR Applications*; Proceedings of the IEEE Conference on Virtual Reality and 3D User Interfaces (VR), p. 9-18; 2019.

(ii) Silveira, T. L. T. and Jung, C. R.; *Perturbation Analysis of the 8-Point Algorithm: a Case Study for Wide FoV Cameras*; Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), p. 11757-11766; 2019.

(iii) Oliveira, A. Q., Silveira, T. L. T., Walter, M. and Jung, C. R.; *On the Performance of DIBR Methods when Using Depth Maps from State-of-the-Art Stereo Matching Algorithms*; Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), p. 2272-2276; 2019.

(iv) Silveira, T. L. T., Dal'Aqua, L. P. and Jung, C. R.; *Indoor Depth Estimation from Single Spherical Images*; Proceedings of the IEEE International Conference on Image Processing (ICIP); p. 2935-2939; 2018.

(v) Silveira, T. L. T. and Jung, C. R.; *Evaluation of Keypoint Extraction and Matching for Pose Estimation Using Pairs of Spherical Images*; Electronic Proceedings of the Conference on Graphics, Patterns and Images (SIBGRAPI); p. 374-381; 2017.

REFERENCES

ABDEL-AZIZ, Y.; KARARA, H. Direct Linear Transformation from Comparator Coordinates into Object Space Coordinates in Close-Range Photogrammetry. **Urbana, IL: American Society of Photogrammetry**, p. 1–18, 1971.

ACHANTA, R. et al. SLIC superpixels compared to state-of-the-art superpixel methods. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 34, n. 11, p. 2274–2281, 2012. .

ACHANTA, R.; SÜSSTRUNK, S. Superpixels and polygons using simple non-iterative clustering. In: **Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017**. [S.l.: s.n.], 2017.

AGARWAL, S. et al. Bundle adjustment in the large. **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**, v. 6312 LNCS, n. PART 2, p. 29–42, 2010.

AGGARWAL, R.; VOHRA, A.; NAMBOODIRI, A. M. Panoramic Stereo Videos with a Single Camera. In: **2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.]: IEEE, 2016. p. 3755–3763.

AKIHIKO, T.; ATSUSHI, I.; OHNISHI, N. Two-and three-view geometry for spherical cameras. **Proc. of the Sixth Workshop on Omnidirectional Vision, Camera Networks and Non- classical Cameras**, v. 105, p. 29–34, 2005.

ALCANTARILLA, P.; NUEVO, J.; BARTOLI, A. Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces. **Procedings of the British Machine Vision Conference 2013**, p. 13.1–13.11, 2013.

ALCANTARILLA, P. F.; BARTOLI, A.; DAVISON, A. J. KAZE Features. In: **Lecture Notes in Computer Science**. [S.l.]: Springer Berlin Heidelberg, 2012. p. 214–227.

ALIBOUCH, B. et al. A phase-based framework for optical flow estimation on omnidirectional images. **Signal, Image and Video Processing**, v. 10, n. 2, p. 285–292, 2016.

ALIBOUCH, B. et al. Optical flow estimation on omnidirectional images: An adapted phase based method. In: **Image and Signal Processing**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. p. 468–475.

ANDERSON, R. et al. Jump: Virtual Reality Video. **ACM Trans. Graph. Article**, v. 3516, n. 1312, p. 978–1, 2016.

ANTOINE, J.-P. et al. Wavelets on the sphere: implementation and approximations. **Applied and Computational Harmonic Analysis**, v. 13, n. 3, p. 177 – 200, 2002. Available from Internet: <http://www.sciencedirect.com/science/article/pii/S1063520302005079>.

ATAPOUR-ABARGHOUEI, A.; BRECKON, T. P. A comparative review of plausible hole filling strategies in the context of scene depth image completion. **Computers & Graphics**, Elsevier, v. 72, p. 39–58, 2018.

BARRON, J. T.; POOLE, B. The Fast Bilateral Solver. In: **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**. [S.l.: s.n.], 2016. v. 9907 LNCS, n. c, p. 617–632.

BRUNTON, A.; LANG, J.; DUBOIS, E. Efficient multi-scale stereo of high-resolution planar and spherical images. **Proceedings - 2nd Joint 3DIM/3DPVT Conference: 3D Imaging, Modeling, Processing, Visualization and Transmission, 3DIMPVT 2012**, p. 120–127, 2012.

CAI, T. T.; ZHANG, A. et al. Rate-optimal perturbation bounds for singular subspaces with applications to high-dimensional statistics. **The Annals of Statistics**, Institute of Mathematical Statistics, v. 46, n. 1, p. 60–89, 2018.

CHANG, A. et al. Matterport3D: Learning from RGB-D Data in Indoor Environments. In: **Proceedings of the International Conference on 3D Vision (3DV)**. [S.l.: s.n.], 2017.

CRUZ-MOTA, J. et al. Scale invariant feature transform on the sphere: Theory and applications. **International Journal of Computer Vision**, v. 98, n. 2, p. 217–241, 2012.

CSURKA, G. et al. Characterizing the uncertainty of the fundamental matrix. **Computer vision and image understanding**, San Diego: Academic Press, c1995-, v. 68, n. 1, p. 18–36, 1997.

CUI, J.; FREEDEN, W. Equidistribution on the sphere. **SIAM Journal on Scientific Computing**, v. 18, n. 2, p. 595–609, 1997.

DENG, X. et al. Automatic spherical panorama generation with two fisheye images. In: **Proceedings of the World Congress on Intelligent Control and Automation (WCICA)**. [S.l.: s.n.], 2008.

DRMAC, Z. On principal angles between subspaces of euclidean space. **SIAM Journal on Matrix Analysis and Applications**, SIAM, v. 22, n. 1, p. 173–194, 2000.

EDER, M.; FRAHM, J.-M. Convolutions on Spherical Images. **IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops**, may 2019. Available from Internet: <http://arxiv.org/abs/1905.08409>.

EIGEN, D.; FERGUS, R. Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-scale Convolutional Architecture. In: **2015 IEEE International Conference on Computer Vision (ICCV)**. [S.l.]: IEEE, 2015. p. 2650–2658.

EIGEN, D.; PUHRSCH, C.; FERGUS, R. Depth map prediction from a single image using a multi-scale deep network. In: **International Conference on Neural Information Processing Systems**. [S.l.: s.n.], 2014. v. 2, p. 2366–2374.

FARNEBÄCK, G. Two-frame motion estimation based on polynomial expansion. **Image Analysis**, v. 2003, n. 1, p. 363–370, 2003.

FELZENSZWALB, P. F.; HUTTENLOCHER, D. P. Efficient graph-based image segmentation. **International Journal of Computer Vision**, v. 59, n. 2, p. 167–181, Sep 2004. Available from Internet: <https://doi.org/10.1023/B:VISI.0000022288.19776.77>.

FERREIRA, L. S.; SACHT, L.; VELHO, L. Local Moebius transformations applied to omnidirectional images. **Computers & Graphics**, Elsevier Ltd, v. 68, p. 77–83, 2017.

FISCHLER, M. A.; BOLLES, R. C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. **Communications of the ACM**, v. 24, n. 6, p. 381–395, 1981.

FUENTES-PACHECO, J.; RUIZ-ASCENCIO, J.; RENDÓN-MANCHA, J. M. Visual simultaneous localization and mapping: a survey. **Artificial Intelligence Review**, v. 43, n. 1, p. 55–81, 2012.

FURUKAWA, Y.; PONCE, J. Accurate, Dense, and Robust Multi-View Stereopsis. In: **2007 IEEE Conference on Computer Vision and Pattern Recognition**. IEEE,

2007. v. 32, n. 8, p. 1–8. Available from Internet: <http://ieeexplore.ieee.org/document/4270271/>.

GASTAL, E. S. L.; OLIVEIRA, M. M. Domain transform for edge-aware image and video processing. **ACM Trans. Graph.**, ACM, New York, NY, USA, v. 30, n. 4, p. 69:1–69:12, 2011.

GAUTAMA, T.; HULLE, M. A. V. A phase-based approach to the estimation of the optical flow field using spatial filtering. **IEEE Transactions on Neural Networks**, v. 13, n. 5, p. 1127–1136, Sep. 2002.

GAVA, C. C.; STRICKER, D.; YOKOTA, S. Dense Scene Reconstruction from Spherical Light Fields. In: **2018 25th IEEE International Conference on Image Processing (ICIP)**. IEEE, 2018. p. 4178–4182. Available from Internet: <https://ieeexplore.ieee.org/document/8453486/>.

GLUCKMAN, J. M.; NAYAR, S. K. Ego-motion and omnidirectional cameras. In: **IEEE International Conference on Computer Vision**. [S.l.: s.n.], 1998. p. 999–1005.

GODARD, C.; Mac Aodha, O.; BROSTOW, G. J. Unsupervised Monocular Depth Estimation with Left-Right Consistency. In: **Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2017. p. 270–279.

GUAN, H. **Local Features, Structure-from-motion and View Synthesis in Spherical Video**. Thesis (PhD), 2017.

GUAN, H.; SMITH, W. A. P. BRISKS: Binary Features for Spherical Images on a Geodesic Grid. In: **Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2017. p. 1–9.

GUAN, H.; SMITH, W. A. P. Structure-From-Motion in Spherical Video Using the von Mises-Fisher Distribution. **IEEE Transactions on Image Processing**, v. 26, n. 2, p. 711–723, feb 2017.

HADFIELD, S. J.; LEBEDA, K.; BOWDEN, R. HARD-PnP: PnP Optimization Using a Hybrid Approximate Representation. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 8828, n. c, 2 018.

HARRIS, C.; STEPHENS, M. A Combined Corner and Edge Detector. **Procedings of the Alvey Vision Conference 1988**, p. 23.1–23.6, 1988. Available from Internet: <http://www.bmva.org/bmvc/1988/avc-88-023.html>.

HARTLEY, R.; ZISSERMAN, A. **Multiple View Geometry in Computer Vision**. [S.l.]: Cambridge, 2003.

HARTLEY, R. I. In defense of the eight-point algorithm. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 19, n. 6, p. 580–593, jun 1997.

HUANG, J. et al. 6-DOF VR videos with a single 360-camera. In: **2017 IEEE Virtual Reality (VR)**. [S.l.]: IEEE, 2017. p. 37–44.

HUANG, T.-C.; TSENG, Y.-H. Indoor Positioning and Navigation Based on Control Spherical Panoramic Images. **ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences**, XLI-B6, n. July, p. 251–258, 2016.

IM, S. et al. **All-Around Depth from Small Motion with a Spherical Panoramic Camera**. Cham: Springer International Publishing, 2016. (Lecture Notes in Computer Science, v. 9907).

JEONG, J. et al. 3DoF+ 360 Video Location-Based Asymmetric Down-Sampling for View Synthesis to Immersive VR Video Streaming. **Sensors**, v. 18, n. 9, 2018. Available from Internet: <http://www.mdpi.com/1424-8220/18/9/3148>.

JUNG, J. et al. **Update on N17618 v2 CTC on 3DoF+ and Windowed 6DoF**. Villar Dora, Italy, 2018. MPEG123/m43571.

KEINERT, B. et al. Spherical fibonacci mapping. **ACM Transactions on Graphics**, v. 34, n. 6, p. 1–7, oct 2015. Available from Internet: <http://dl.acm.org/citation.cfm?doid=2816795.2818131>.

KIM, H.; HILTON, A. 3D scene reconstruction from multiple spherical stereo pairs. **International Journal of Computer Vision**, 2013.

KIM, H.; HILTON, A. Block world reconstruction from spherical stereo image pairs. **Computer Vision and Image Understanding**, 2015.

KIRISITS, C.; LANG, L. F.; SCHERZER, O. Decomposition of optical flow on the sphere. **GEM - International Journal on Geomathematics**, v. 5, n. 1, p. 117–141, 2014.

KO, H. et al. Robust uncalibrated stereo rectification with constrained geometric distortions (USR-CGD). **Image and Vision Computing**, Elsevier B.V., v. 60, p. 98–114, 2017.

KOPF, J. 360-degree video stabilization. **ACM Transactions on Graphics**, v. 35, n. 6, p. 1–9, nov 2016.

KRUSKAL, W. H.; WALLIS, W. A. Use of ranks in one-criterion variance analysis. **Journal of the American statistical Association**, Taylor & Francis Group, v. 47, n. 260, p. 583–621, 1952.

LAI, P. K. et al. Real-Time Panoramic Depth Maps from Omni-directional Stereo Images for 6 DoF Videos in Virtual Reality. **2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)**, p. 405–412, 2019. Available from Internet: <https://ieeexplore.ieee.org/document/8798016/>.

LEUTENEGGER, S.; CHLI, M.; SIEGWART, R. Y. BRISK: Binary Robust invariant scalable keypoints. In: **2011 International Conference on Computer Vision**. [S.l.]: IEEE, 2011. p. 2548–2555.

LEVENBERG, K. A Method for the Solution of Certain Non-Linear Problems in Least. **Quarterly of Applied Mathematics**, v. 2, n. 278, p. 164–168, 1944.

LIM, J.; BARNES, N.; LI, H. Estimating relative camera motion from the antipodal-epipolar constraint. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 32, n. 10, p. 1907–1914, 2010.

LIU, F.; SHEN, C.; LIN, G. Deep Convolutional Neural Fields for Depth Estimation from a Single Image. In: **Proc. IEEE Conf. Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2015.

LIU, F. et al. Learning Depth from Single Monocular Images Using Deep Convolutional Neural Fields. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 38, n. 10, p. 2024–2039, oct 2016.

LO, I.-c.; SHIH, K.-t.; CHEN, H. H. Image Stitching for Dual Fisheye Cameras. **2018 25th IEEE International Conference on Image Processing (ICIP)**, IEEE, p. 3164–3168, 2018.

LONGUET-HIGGINS, H. C. A Computer Algorithm for Reconstructing a Scene from Two Projections. In: **Readings in computer vision: issues, problems, principles, and paradigms**. [S.l.: s.n.], 1987. p. 61–62.

LOWE, D. G. Distinctive Image Features from Scale-Invariant Keypoints. **International Journal of Computer Vision**, v. 60, n. 2, p. 91–110, nov 2004.

LUKIERSKI, R.; LEUTENEGGER, S.; DAVISON, A. J. Rapid free-space mapping from a single omnidirectional camera. In: **2015 European Conference on Mobile Robots, ECMR 2015 - Proceedings**. [S.l.: s.n.], 2015.

MAIR, E.; SUPPA, M.; BURSCHKA, D. Error propagation in monocular navigation for zinf compared to eightpoint algorithm. In: IEEE. **Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on**. [S.l.], 2013. p. 4220–4227.

MERIKOSKI, J. K.; SARRIA, H.; TARAZAGA, P. Bounds for singular values using traces. **Linear Algebra and its Applications**, Elsevier, v. 210, p. 227–254, 1994.

MIKOLAJCZYK, K.; SCHMID, C. A performance evaluation of local descriptors. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 27, n. 10, p. 1615–1630, Oct 2005.

MOLNAR, J. et al. 3D reconstruction of planar patches seen by omnidirectional cameras. **2014 International Conference on Digital Image Computing: Techniques and Applications, DICTA 2014**, 2015.

MOREAU, J.; AMBELLOUIS, S.; RUICHE, Y. 3D reconstruction of urban environments based on fisheye stereovision. In: **8th International Conference on Signal Image Technology and Internet Based Systems, SITIS 2012r**. [S.l.: s.n.], 2012.

MOREL, J.-M.; YU, G. ASIFT: A New Framework for Fully Affine Invariant Image Comparison. **SIAM Journal on Imaging Sciences**, v. 2, n. 2, p. 438–469, 2009.

MOSTEGEL, C. et al. Scalable Surface Reconstruction from Point Clouds with Extreme Scale and Density Diversity. In: **IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2017. p. 904–913.

MÜHLICH, M.; MESTER, R. The role of total least squares in motion analysis. In: SPRINGER. **European Conference on Computer Vision**. [S.l.], 1998. p. 305–321.

NAYAR, S. K. Catadioptric Omnidirectional Camera *. In: **Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 1997. p. 482–488.

NELSON, R. C.; ALOIMONOS, J. Biological Cybernetics Finding Motion Parameters from Spherical Motion Fields. v. 273, n. Ullman 1981, p. 261–273, 1988.

NISTÉR, D. An efficient solution to the five-point relative pose problem. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 26, n. 6, p. 756–770, 2004.

OLIVEIRA, A. Q. de et al. On the performance of dibr methods when using depth maps from state-of-the-art stereo matching algorithms. In: **Proceedings of the 44th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)**. [S.l.: s.n.], 2019. p. 2272–2276.

OLIVEIRA, A. Q. de; WALTER, M.; JUNG, C. R. An artifact-type aware dibr method for view synthesis. **IEEE Signal Processing Letters**, v. 25, n. 11, p. 1705–1709, Nov 2018.

O'ROURKE, S.; VU, V.; WANG, K. Random perturbation of low rank matrices: Improving classical bounds. **arXiv preprint arXiv:1311.2657**, 2013.

Thomas Wesley Osborne, Todor Georgiev Georgiev and Sergiu Radu Goma. **Wide field of view array camera for hemispheric and spherical imaging**. 2014. US9819863B2. Qualcomm Inc. US9819863B2.

OZYESIL, O. et al. A Survey of Structure from Motion. p. 305–364, 2017. Available from Internet: <http://arxiv.org/abs/1701.08493>.

PAGANI, A. et al. Dense 3D Point Cloud Generation from Multiple High-resolution Spherical Images. **International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST)**, p. 1–8, 2011.

PAGANI, A.; STRICKER, D. Structure from Motion using full spherical panoramic cameras. In: **2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)**. [S.l.]: IEEE, 2011. p. 375–382.

PALMA, G. et al. Scalable non-rigid registration for multi-view stereo data. **Journal of Photogrammetry and Remote Sensing**, v. 142, p. 328–341, 2018.

PATHAK, S. et al. 3D reconstruction of structures using spherical cameras with small motion. In: **2016 16th International Conference on Control, Automation and Systems (ICCAS)**. [S.l.]: IEEE, 2016. p. 117–122.

PATHAK, S. et al. Spherical Video Stabilization by Estimating Rotation from Dense Optical Flow Fields. **Journal of Robotics and Mechatronics**, v. 29, n. 3, p. 566–579, jun 2017.

PATHAK, S. et al. Virtual Reality with Motion Parallax by Dense Optical Flow-Based Depth Generation from Two Spherical Images. p. 1–6, 2017.

PATHAK, S. et al. Distortion-Robust Spherical Camera Motion Estimation via Dense Optical Flow. In: **2018 25th IEEE International Conference on Image Processing (ICIP)**. [S.l.]: IEEE, 2018. p. 3358–3362.

PATHAK, S. et al. Rotation Removed Stabilization of Omnidirectional Videos Using Optical Flow. **The Abstracts of the international conference on advanced mechatronics : toward evolutionary fusion of IT and mechatronics : ICAM**, v. 2015.6, n. 1, p. 51–52, 2015. Available from Internet: <https://www.jstage.jst.go.jp/article/jsmeicam/2015.6/0/2015.6{_}51/{_}a>.

PATHAK, S. et al. A decoupled virtual camera using spherical optical flow. In: **2016 IEEE International Conference on Image Processing (ICIP)**. [S.l.]: IEEE, 2016. p. 4488–4492.

PATHAK, S. et al. Dense 3D reconstruction from two spherical images via optical flow-based equirectangular epipolar rectification. In: **2016 IEEE International Conference on Imaging Systems and Techniques (IST)**. [S.l.]: IEEE, 2016. p. 140–145.

PATHAK, S. et al. Optical Flow-Based Epipolar Estimation of Spherical Image Pairs for 3D Reconstruction. **SICE Journal of Control, Measurement, and System Integration**, v. 10, n. 5, p. 476–485, 2017.

PERONA, P.; MALIK, J. Scale-space and edge detection using anisotropic diffusion. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE, v. 12, n. 7, p. 629–639, 1990.

PETSCHNIGG, G. et al. Digital photography with flash and no-flash image pairs. In: ACM. **ACM transactions on graphics (TOG)**. [S.l.], 2004. v. 23, n. 3, p. 664–672.

RADGUI, A. et al. Optical flow estimation from multichannel spherical image decomposition. **Computer Vision and Image Understanding**, v. 115, n. 9, p. 1263–1272, 2011.

RADKE, R. J. **Computer Vision for Visual Effects**. [s.n.], 2012. Available from Internet: <http://ebooks.cambridge.org/ref/id/CBO9781139019682>.

ROSTEN, E.; DRUMMOND, T. Machine Learning for High-Speed Corner Detection. In: **Computer Vision – ECCV 2006**. [S.l.: s.n.], 2006. v. 1, p. 430–443.

RUBLEE, E. et al. ORB: An efficient alternative to SIFT or SURF. In: **2011 International Conference on Computer Vision**. [S.l.]: IEEE, 2011. p. 2564–2571.

RUSINKIEWICZ, S.; LEVOY, M. Efficient variants of the ICP algorithm. In: **Proceedings of International Conference on 3-D Digital Imaging and Modeling, 3DIM**. [S.l.: s.n.], 2001.

SCHARSTEIN, D.; SZELISKI, R.; ZABIH, R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. **Proceedings - IEEE Workshop on Stereo and Multi-Baseline Vision, SMBV 2001**, v. 47, n. 1, p. 131–140, 2001.

SCHONBERGER, J. L.; FRAHM, J.-M. Structure-from-Motion Revisited. **2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**, p. 4104–4113, 2016.

SCHWARZ, S.; HANNUKSELA, M. M. Perceptual quality assessment of hevc main profile depth map compression for six degrees of freedom virtual reality video. In: IEEE. **Image Processing (ICIP), 2017 IEEE International Conference on**. [S.l.], 2017. p. 181–185.

SEITZ, S. et al. A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms. In: **2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR'06)**. IEEE, 2006. v. 1, p. 519–528. Available from Internet: <http://ieeexplore.ieee.org/document/1640800/>.

SILVEIRA, T. L. T.; DAL'AQUA, L.; JUNG, C. R. Indoor Depth Estimation From Single Spherical Images. In: **IEEE International Conference on Image Processing (ICIP)**. [S.l.: s.n.], 2018.

SILVEIRA, T. L. T. D.; JUNG, C. R. Evaluation of Keypoint Extraction and Matching for Pose Estimation Using Pairs of Spherical Images. **2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)**, p. 374–381, 2017.

SILVEIRA, T. L. T. da; JUNG, C. R. Dense 3D Scene Reconstruction from Multiple Spherical Images for 3-DoF+ VR Applications. In: **Proceedings of the 26th IEEE Conference on Virtual Reality and 3D User Interfaces (VR)**. [S.l.: s.n.], 2019. p. 9–18.

SILVEIRA, T. L. T. da; JUNG, C. R. Perturbation Analysis of the 8-Point Algorithm: a Case Study for Wide FoV Cameras. In: **Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2019. p. 11757–11766.

SOLH, M.; ALREGIB, G. Hierarchical hole-filling for depth-based view synthesis in ftv and 3d video. **IEEE Journal of Selected Topics in Signal Processing**, IEEE, v. 6, n. 5, p. 495–504, 2012.

SONG, S. et al. Im2Pano3D: Extrapolating 360 Structure and Semantics Beyond the Field of View. v. 1, 2017. Available from Internet: <http://arxiv.org/abs/1712.04569>.

SPEARMAN, C. The proof and measurement of association between two things. **The American Journal of Psychology**, v. 15, n. 1, p. 72–101, 1904.

STEWART, G. W. Matrix perturbation theory. Citeseer, 1990.

SU, H. et al. Render for CNN: Viewpoint estimation in images using CNNs trained with rendered 3D model views. In: **Proceedings of the IEEE International Conference on Computer Vision**. [S.l.: s.n.], 2015. p. 2686–2694.

SU, Y.-C.; GRAUMAN, K. Learning Spherical Convolution for Fast Features from 360 Imagery. In: **Conference on Neural Information Processing Systems**. [S.l.: s.n.], 2017. p. 529–539.

SU, Y.-C.; GRAUMAN, K. Kernel Transformer Networks for Compact Spherical Convolution. In: **Conference on Computer Vision and Pattern Recognition**. [s.n.], 2019. p. 9442–9451. Available from Internet: <http://arxiv.org/abs/1812.03115>.

SUR, F.; NOURY, N.; BERGER, M.-O. Computing the uncertainty of the 8 point algorithm for fundamental matrix estimation. In: **19th British Machine Vision Conference-BMVC 2008**. [S.l.: s.n.], 2008. p. 10.

SZELISKI, R. **Computer Vision: Algorithms and Applications**. 1st. ed. Berlin, Heidelberg: Springer-Verlag, 2010.

TEZUKA, T. et al. View synthesis using superpixel based inpainting capable of occlusion handling and hole filling. In: **2015 Picture Coding Symposium (PCS)**. [S.l.: s.n.], 2015. p. 124–128.

THATTE, J. et al. Stacked Omnistereo for virtual reality with six degrees of freedom. In: **2017 IEEE Visual Communications and Image Processing (VCIP)**. IEEE, 2017. p. 1–4. Available from Internet: <http://ieeexplore.ieee.org/document/8305085/>.

TIAN, T.; TOMASI, C.; HEEGER, D. Comparison of approaches to egomotion computation. **Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition**, n. 2, p. 315–320, 1996.

TOMASI, C.; KANADE, T. **Detection and Tracking of Point Features**. [S.l.], 1991.

TOMASI, C.; MANDUCHI, R. Bilateral filtering for gray and color images. **Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)**, p. 839–846. Available from Internet: <http://ieeexplore.ieee.org/document/710815/>.

TONG, J.; NING, X. Depth measurement by omni-directional camera. In: **2013 IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems, IEEE-CYBER 2013**. [S.l.: s.n.], 2013.

TORII, A.; HAVLENA, M.; PAJDLA, T. From Google Street View to 3D City models. **Iccvw**, p. 2188–2195, 2009.

TORR, P. H.; MURRAY, D. W. The development and comparison of robust methods for estimating the fundamental matrix. **International journal of computer vision**, Springer, v. 24, n. 3, p. 271–300, 1997.

TROIANI, C. et al. 1-Point-based monocular motion estimation for computationally-limited micro aerial vehicles. In: **2013 European Conference on Mobile Robots**. [S.l.]: IEEE, 2013. p. 13–18.

TUKEY, J. W. **Exploratory data analysis**. [S.l.]: Reading, Mass., 1977.

VALGAERTS, L. et al. Dense versus sparse approaches for estimating the fundamental matrix. **International Journal of Computer Vision**, v. 96, n. 2, p. 212–234, 2012.

WANG, R. Singular vector perturbation under gaussian noise. **SIAM Journal on Matrix Analysis and Applications**, SIAM, v. 36, n. 1, p. 158–177, 2015.

WANG, W.; TSUI, H. An SVD Decomposition of Essential Matrix with Eight Solutions for the Relative Positions of Two Perspective Cameras. **International Conference on Pattern Recognition**, v. 1, p. 1362–365 vol.1, 2000. Available from Internet: <http://dx.doi.org/10.1109/icpr.2000.905353>.

WEDIN, P.-Å. Perturbation bounds in connection with singular value decomposition. **BIT Numerical Mathematics**, Springer, v. 12, n. 1, p. 99–111, 1972.

WEGNER, K. et al. Depth Estimation from Stereoscopic 360-Degree Video. **Proceedings - International Conference on Image Processing, ICIP**, p. 2945–2948, 2018.

WEI, X. et al. Superpixel hierarchy. **IEEE Transactions on Image Processing**, v. 27, n. 10, p. 4838–4849, Oct 2018.

WEINZAEPFEL, P. et al. DeepFlow: Large displacement optical flow with deep matching. **Proceedings of the IEEE International Conference on Computer Vision**, n. Section 2, p. 1385–1392, 2013.

WELCH, B. L. The generalization of 'student's' problem when several different population variances are involved. **Biometrika**, v. 34, n. 1-2, p. 28–35, 1947. Available from Internet: <+http://dx.doi.org/10.1093/biomet/34.1-2.28>.

WENG, J.; HUANG, T. S.; AHUJA, N. Motion and structure from two perspective views: Algorithms, error analysis, and error estimation. **IEEE Transactions on Pattern Analysis & Machine Intelligence**, IEEE, n. 5, p. 451–476, 1989.

WEYL, H. Das asymptotische verteilungsgesetz der eigenwerte linearer partieller differentialgleichungen (mit einer anwendung auf die theorie der hohlraumstrahlung). **Mathematische Annalen**, Springer, v. 71, n. 4, p. 441–479, 1912.

WON, C.; RYU, J.; LIM, J. SweepNet: Wide-baseline Omnidirectional Depth Estimation. p. 6073–6079, 2019. Available from Internet: <http://arxiv.org/abs/1902.10904>.

WOOD, A. T. Simulation of the von mises fisher distribution. **Communications in Statistics - Simulation and Computation**, v. 23, n. 1, p. 157–164, 1994.

XIAO, G. et al. Superpixel-Guided Two-View Deterministic Geometric Model Fitting. **International Journal of Computer Vision**, Springer US, v. 127, n. 4, p. 323–339, 2019. Available from Internet: <https://doi.org/10.1007/s11263-018-1100-8>.

XIAO, J. et al. Recognizing scene viewpoint using panoramic place representation. In: **2012 IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.]: IEEE, 2012. p. 2695–2702.

XU, B. et al. Optical flow-based video completion in spherical image sequences. In: **2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)**. [S.l.]: IEEE, 2016. p. 388–395.

XU, B. et al. Spatio-Temporal Video Completion in Spherical Image Sequences. **IEEE Robotics and Automation Letters**, v. 2, n. 4, p. 2032–2039, oct 2017.

XU, W. X. W.; MULLIGAN, J. Robust relative pose estimation with integrated cheirality constraint. **2008 19th International Conference on Pattern Recognition**, n. m, p. 6–9, 2008.

YANG, H.; ZHANG, H. Efficient 3D Room Shape Recovery from a Single Panorama. In: **2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.]: IEEE, 2016. v. 2016-Decem, p. 5422–5430.

YANG, J.; LI, H.; JIA, Y. Optimal essential matrix estimation via inlier-set maximization. **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**, v. 8689 LNCS, n. PART 1, p. 111–126, 2014.

YANG, Q. Stereo matching using tree filtering. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, 2015.

YANG, S.-T. et al. DuLa-Net: A Dual-Projection Network for Estimating Room Layouts from a Single RGB Panorama. In: **IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [s.n.], 2019. p. 3363–3372. Available from Internet: <http://arxiv.org/abs/1811.11977>.

YANG, Y.; LIU, R.; KANG, S. B. Automatic 3D Indoor Scene Modeling from Single Panorama. In: **Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition**. [S.l.]: IEEE, 2018. p. 5430.

YI, K. M. et al. Learning to find good correspondences. In: **Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2018. p. 2666–2674.

ZBONTAR, J.; LECUN, Y. Stereo matching by training a convolutional neural network to compare image patches. **J. Mach. Learn. Res.**, JMLR.org, v. 17, n. 1, p. 2287–2318, jan. 2016. Available from Internet: <http://dl.acm.org/citation.cfm?id=2946645.2946710>.

ZHANG, C. et al. MeshStereo: A Global Stereo Model with Mesh Alignment Regularization for View Interpolation. In: **2015 IEEE International Conference on Computer Vision (ICCV)**. [S.l.]: IEEE, 2015. p. 2057–2065.

ZHANG, Y. et al. PanoContext: A whole-room 3D context model for panoramic scene understanding. In: **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**. [S.l.: s.n.], 2014.

ZHANG, Z. et al. Benefit of large field-of-view cameras for visual odometry. **Proceedings - IEEE International Conference on Robotics and Automation**, v. 2016-June, p. 801–808, 2016.

ZHAO, Q. et al. Spherical Superpixel Segmentation. **IEEE Transactions on Multimedia**, v. 20, n. 6, p. 1406–1417, 2018.

ZHAO, Q. et al. SPHORB: A Fast and Robust Binary Feature on the Sphere. **International Journal of Computer Vision**, Springer US, v. 113, n. 2, p. 143–159, 2014.

ZHOU, C. et al. Fast, Accurate Thin-Structure Obstacle Detection for Autonomous Mobile Robots. In: **The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops**. [S.l.: s.n.], 2017. p. 1–10.

ZIOULIS, N. et al. OmniDepth: Dense Depth Estimation for Indoors Spherical Panoramas. In: . [S.l.: s.n.], 2018. p. 453–471.

ZOU, C. et al. LayoutNet: Reconstructing the 3D Room Layout from a Single RGB Image. 2018. Available from Internet: <http://arxiv.org/abs/1803.08999>.