Evaluation de la qualité des vidéos panoramiques assemblées

Sandra Nabil

Communauté
UNIVERSITÉ Grenoble Alpes

Thèse

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES

Spécialité : **Mathématiques et Informatique**

Présentée par

Sandra NABIL

Thèse dirigée par **James CROWLEY**
et codirigée par **Frédéric DEVERNAY**

préparée au sein **du Laboratoire Jean Kuntzmann à l'INRIA Rhône-Alpes**
et de **l'École Doctorale MSTII**

Evaluation de la qualité des vidéos panoramiques assemblées

Quality Evaluation for Stitched Panoramic Videos

Thèse soutenue publiquement le **27 Novembre 2018**,
devant le jury composé de :

M., Rémi RONFARD
Directeur de Recherche, INRIA Grenoble, France, Président
Mme, Luce MORIN
Professeur, INSA Rennes, France, Rapporteur
M., Patrick LE CALLET
Professeur, Polytech Nantes/Université de Nantes, France, Rapporteur
M., Antoine MELER
Ingénieur de Recherche, GoPro Francin, France, Examinateur
M., James CROWLEY
Professeur, Grenoble INP, France, Examinateur
M., Frédéric DEVERNAY
Chargé de recherche, INRIA Grenoble, France, Examinateur

"To my beloved parents and sister"

This research has been made possible by the funding of Live360TV project and by the use of equipment provided by ANR Equipment for Excellence Amiqual4Home (ANR-11-EQPX-0002).

Acknowledgements

I am thankful to everyone who helped me achieve this step in my research career and education.

Specifically, thanks to my thesis advisors James Crowley and Frédéric Devernay for allowing me to have this opportunity and for their valuable advise during those 3 years and for the great suggestions for this manuscript and for the defense presentation.

Thanks to all the jury members, Luce Morin and Patrick Le Callet for reviewing my manuscript, for their insightful comments before and during the defense as well as their appreciation of my work. Thanks to Antoine Meler for the feedback on my work and the inspiring questions at my defense. Thanks to Rémi Ronfard for perfectly leading the discussion during the defense.

Thanks to Raffaella Balzarini for her great help in the human-study experiment and for her feedback on my work. Working with her was always a pleasure.

Thanks to Nadine Mandran for guiding me through the first steps of designing the user-experiment.

Thanks to Stan Borkowski for helping us build a panoramic camera rig.

Thanks to Stéphane Gamet and the team in Kolor GoPro for lending us GoPro Omni camera.

Thanks to colleagues and friends who took the time to participate in the user experiment, helped by being part in the video datasets or gave me feedback on my work.

Thanks to Christine for her helping me taking the datasets in Cairo and for being a great friend.

Thanks to my awesome parents and my amazing sister for their continuous support and love. Without them I would have achieved nothing.

Abstract

High quality panoramic videos for immersive VR content are commonly created using a rig with multiple cameras covering a target scene. Unfortunately, this setup introduces both spatial and temporal artifacts due to the difference in optical centers as well as the imperfect synchronization. Traditional image quality metrics cannot be used to assess the quality of such videos, due to their inability to capture geometric distortions. In addition, the lack of a reference to these panoramic videos represents another challenging problem.

In this thesis, we propose two approaches for the objective assessment of panoramic videos which offer alternatives to the ground truth and are able to quantify the geometric deformations that are present in stitched videos. These methods are then validated with an empirical experiment that combines human error annotation and eye-tracking.

The first approach suggests to compare the overlapping regions of matched pairs prior to blending using a method initially proposed for novel view synthesis and which models three visibility features based on the human visual perception. A combined map is created by pooling the values of the intermediate maps. The second approach investigates the use of the original videos as a reference for the output panorama. We note that this approach is not directly applicable, because each pixel in the final panorama can have one to N sources corresponding to N input videos with overlapping regions. We show that this problem can be solved by calculating the standard deviation of displacements of all source pixels from the displacement of the panorama as a measure of distortion. This makes it possible to compare the difference in motion between two given frames in the original videos and motion in the final panorama. Salience maps based on human perception are used to weight the distortion map for more accurate filtering.

An empirical experiment is then conducted with human participants to validate the proposed objective evaluation methods. The experiment aimed

at answering the question of whether humans and the objective methods detect and measure the same errors, and exploring which errors are more salient to humans when watching a panoramic video.

The methods described have been tested and validated and they yield promising results which show a high correlation between the proposed algorithms and the humans perception as well as provide interesting findings regarding human-based perception for quality metrics. They also open the way to new methods for optimizing video stitching guided by those quality metrics.

Résumé

Des vidéos panoramiques de haute qualité pour un contenu VR immersif sont généralement créées à l'aide d'une plate-forme avec plusieurs caméras couvrant une scène cible. Malheureusement, cette configuration introduit à la fois des artefacts spatiaux et temporels dus à la différence entre les centres optiques et à la synchronisation imparfaite. Les métriques de qualité d'image traditionnelles ne peuvent pas être utilisées pour évaluer la qualité de telles vidéos, en raison de leur incapacité à capturer les distorsions géométriques. De plus, l'absence de référence à ces vidéos panoramiques représente un autre problème épineux. Dans cette thèse, nous proposons deux approches pour l'évaluation objective de vidéos panoramiques qui offrent une alternative à la vérité de terrain et permettent de quantifier les déformations géométriques présentes dans les vidéos assemblées. Ces méthodes sont ensuite validées par une expérience empirique associant annotation d'erreur humaine et eye-tracking.

La première approche suggère de comparer les régions qui se chevauchent de paires appariées avant le mélange en utilisant une méthode initialement proposée pour la synthèse de vues nouvelle et qui modélise trois caractéristiques de visibilité basées sur la perception visuelle humaine. Une carte combinée est créée en regroupant les valeurs des cartes partielles.

La seconde approche étudie l'utilisation des vidéos originales comme référence pour le panorama en sortie. Nous notons que cette approche n'est pas directement applicable, car chaque pixel du panorama final peut avoir une à N sources correspondant à N vidéos d'entrée avec des régions qui se chevauchent. Nous montrons que ce problème peut être résolu en calculant l'écart type des déplacements de tous les pixels source par rapport au déplacement du panorama en tant que mesure de la distorsion. Cela permet de comparer la différence de mouvement entre deux images données dans les vidéos d'origine et le mouvement dans le panorama final. Des cartes de saillance basées sur la

perception humaine sont utilisées pour pondérer la carte de distorsion pour un filtrage plus précis. Une expérience empirique est ensuite menée avec des participants humains pour valider les méthodes d'évaluation objectives proposées. L'expérience visait à déterminer si les humains et les méthodes objectives détectaient et mesuraient les mêmes erreurs, et à déterminer quelles erreurs étaient les plus saillantes pour les humains lorsqu'ils regardaient une vidéo panoramique.

Les méthodes décrites ont été testées et validées et elles donnent des résultats prometteurs montrant une forte corrélation entre les algorithmes proposés et la perception humaine, ainsi que des résultats intéressants concernant la perception humaine des métriques de qualité. Ils ouvrent également la voie à de nouvelles méthodes d'optimisation de l'assemblage vidéo guidées par ces métriques de qualité.

Contents

Chapter 1

Introduction

1.1 Historical Background

Panorama creation is an ancient concept that has attracted people years BC. Before the invention of cameras, when the only means of photography was using paintings or even murals, people have always been interested in creating wide field of view content.

One of the earliest panoramas was found in the ruins excavated in villa of the mysteries Pompeii as early as 20 AD. It was later expressed through artistic paintings to archive historical events or to express the richness of a landscape. Museums and art galleries have examples of panoramic paintings (see figure 1.1) such as "Joseph pardons his Brothers" for instance, probably 1515, which is now in the National gallery of London or "La marche du Grand Seigneur avec sa garde de janissaires et de spahis", 1834, which can be seen in Louvre museum, Paris. The term "panorama" itself, from Greek "pan"=all and "horama"=view, was first introduced by the Irish painter Robert Barker in 1972 [Wikipedia contributors, 2018] to describe one of his paintings that he projected on a sphere.

Figure 1.1: Examples of panoramic paintings that existed before panoramic photography.

Later on, with the invention of photography using daguerrotypes[1] by Louis Daguerre in the 19th century, panoramas were created using a composition of daguerrotypes (see figure 1.2). In 1843, the first panoramic camera was patented by Puchberger using a technique called swing lens, consisting in a lens swinging around a vertical axis to take a wide field of view. Later in 1858, Chevalier and Porro invented panoramic cameras independently. However, it was not until using photo-theodolites[2] 1895 by Finsterwalder and Zeiss that the panoramic photography has become successful. With the emergence of digital cameras in the 1990s, panoramic cameras started to be available in the market [Luhmann, 2004].

Figure 1.2: Left: Panorama created using 8 daguerrotypes in 1848. Right: Swing lens used for early panorama shooting. (figures taken from [Luhmann, 2004]).

Many contributions have been made in the past to provide 360° images and videos, especially for the purpose of movie making. In 1897, Raoul Grimoin-Sanson patented Cineorama which consisted of a circular screen along with 10 synchronized projectors. It was later used to project frames of a movie in a circular panorama presenting a simulation of the experience of a hot air balloon in an expedition in Paris in 1990. Photorama is an example of a procedure and device taking 360° photos and projecting them onto a cylindrical projection, this was invented by Louis Lumière back in 1900. Later, in 1955, the Walt Disney company had an early panoramic video system consisting of 9 cameras with a projector between each pair of cameras and 9 screens placed in a circle to display the projected movie.

[1] The first publicly available photography methodology in which photos were created on a silver copper plate placed inside a camera and fumed with mercury to create a permanent image.

[2] A camera combined with a theodolite, an optical instrument to measure angles between the world and the camera.

Given the importance of panoramic photography and panoramic videos, this technology has continued to develop and panoramic videos is currently a highly active field of research that has many applications and that continue to advance with large steps.

1.2 Motivation

Panoramic videos enable us to capture the richness of the surrounding world. Producing 360° panoramic videos is an interesting tool to provide content for immersive environments and virtual reality and can be useful for several applications such as video games, film making, sports events, tourism and more.

These applications however require a high precision, resolution and they are intolerant to errors, especially when seen using a stereo head-mounted display (HMD). Unfortunately, avoiding stitching errors in panoramic videos is nearly impossible. This is due to the required capture setup that usually involves a panoramic rig with multiple overlapping covering a wide scene. The placement of these cameras in different positions creates parallax errors that appear in the form of ghosting, misalignment or deformation. In addition, it is not always possible to perfectly synchronize these devices together.

Aligning panoramic video frames with image stitching algorithms is not directly applicable to videos due to the multiple cameras used in panoramic video setup which cause parallax. In addition applying stitching on a frame-by-frame basis results in temporal incoherence.

Our goal in this thesis is to understand and quantify those errors using spatial and temporal metrics. We could not rely on traditional quality metrics since they are not designed to capture geometric deformations. In addition, panoramic videos do not have a reference to compare to, instead they are synthesized out of the input videos. Therefore, we present solutions for these problems using objective quality metrics and validate them using a human-centered study.

1.3 Achievements and Contributions

The main achievements and contributions of this thesis are:

- The creation of a panoramic rig and the creation of a dataset for panoramic videos using this rig and an Omni GoPro camera.
- The implementation of state-of-the-art video stitching algorithm of Perazzi et al. [Perazzi et al., 2015].
- A spatial quality metric for panoramic videos that predicts errors prior to blending.
- An optical flow-based temporal quality metric for panoramic videos.
- An error annotation interface for subjective quality metric for panoramic videos.
- A human experiment validation for the proposed objective quality metric, based on the annotation interface and on eye tracking.

1.4 Organization of the Manuscript

Chapter 2 lays the foundation and the background knowledge necessary to understand the rest of the manuscript and the contributions of the thesis. It contains basics of panoramic photography and image stitching which are the baseline to understanding video stitching algorithms. Panoramic photography challenges are discussed along with suggestions of techniques to overcome them. Image stitching basics include a discussion of motion models that are used to establish relationships between images. Image alignment techniques are presented along with camera parameters estimation. Methods for blending images to create a seamless panorama are reviewed afterwards. The chapter also introduces the topic of panoramic videos including the capture systems, the added implications and why image stitching techniques are not directly applicable to videos.

This leads to **chapter 3** which discusses issues of concern in panoramic videos and provides a survey of the state-of-the-art video stitching methods. A large number of research efforts are dedicated to overcome the problem of parallax errors in panoramic videos. The solutions presented are categorized into mesh optimization methods, content-aware seam selection and variations of the blending function. Temporal inconsistency is another issue of concern that is addressed in this chapter and stabilization or temporal alignment are suggested to produce coherent panoramic videos. Another set of methods

are covered whose goal is to create monoscopic or stereoscopic virtual reality content and therefore focus on producing high quality, high-resolution and real-time panoramic videos. Afterwards, the chapter elaborates on the video stitching method implemented within this thesis. This chapter gives the motivation behind this thesis contributions since it shows the difficulty of overcoming visual artifacts in panoramic videos and eventually the importance of designing metrics to assess the quality of these videos.

Chapter 4 presents the proposed spatial quality metric and the improved optical flow-based temporal quality metric after presenting the related state-of-the-art objective quality metrics. The proposed spatial metric works on areas of overlap between pairs of panoramic images within the same frame in time using salience features proposed by Conze et al. [Conze et al., 2012]. A composite distortion map is then built and a blend mask is applied as a weight map to give more importance to pixels on the overlap area. The temporal metric works on comparing the final panoramic output frame with the source input videos. A distortion map is proposed based on optical flow between two consecutive frames to compare the relative motion flow change at each pixel between the output frame and the individual inputs using standard deviation. A saliency map is calculated based on texture, contrast and orientation of gradients [Conze et al., 2012] and a combined map is obtained using a weighted sum of the distortion and the salience maps to represent the hypothesis that attention will be drawn to a salient region or a distortion.

Chapter 5 describes the human-centered experiment conducted to validate the proposed objective quality methods. The chapter starts with a background on the human visual system with focus on the parts that control attention and fixation, both of which are used in the user experiment. Visual attention is then defined with a review of important experiments in the literature and their findings conducted by cognitive scientists and psychologists. The related state-of-the-art of subjective quality metrics are then discussed followed by the protocol for the proposed human-centered study based on annotations and eye-tracking. The method conducted represents a study of the human perception to errors using an annotation interface and human visual attention using eye-tracking and was used to validate the objective metrics. Overall, the chapter explains another important contribution which is the human-centered study from design to analysis with the necessary background and related work.

The results of the proposed methods of chapters 4 and 5 are presented

in **chapter 6**. Description of the capture devices and the datasets acquired and used for these experiments are presented first, then the results of the algorithms on different datasets are shown visually. Analysis of the gaze data and annotations is shown and compared to the proposed metric. A discussion and an interpretation of those results is made for all the proposed methods.

Finally, **chapter 7** summarizes and concludes the thesis content, the proposed methods and their results. It also explains the limitations of the suggested approaches with possible solutions.

There are two **appendices** in this manuscript. The first one explains briefly panoramic image projections which are mentioned in chapter 2 and gives examples of the popular projection surfaces used for compositing different views into a single panorama. The second appendix contains further results of the proposed metrics that were not essential to explain the findings in chapter 6 but which serve as additional examples.

Chapter 2

Image Stitching: from Capture to Display

Computational photo stitching for panoramas and mosaics has been covered extensively in the research literature. Nowadays, nearly every digital camera or smartphone has the capability of capturing high quality panoramic images. Understanding the basic knowledge of this topic and the related literature is necessary as part of this thesis. The following sections explain panoramic photography and image stitching in details.

2.1 Overview of Panoramic Photography

Panoramic photography consists in taking multiple overlapping photos in the intent of generating a single composite seamless wide-angle or panoramic photo. The process used to combine these input images together is called image stitching and the resulting image is called a panorama or a mosaic. In order to appreciate the importance of panoramas, a brief understanding of camera lenses vs. human eyes is of interest.

A comparison is made in [Mchugh, 2005] mainly on 3 aspects that are of interest here. The first one is the angle of view, which is demonstrated in figure 2.1. Each one of the human eyes has an angle of view of around 180° as seen in the image to the left 2.1 with an overlap of 130° between both eyes. Although this might seem like a very wide angle of view similar to the fisheye camera lens shown to the right of figure 2.1, the actual central vision is around 40°-60° comparable to a standard camera lens, whereas the peripheral vision

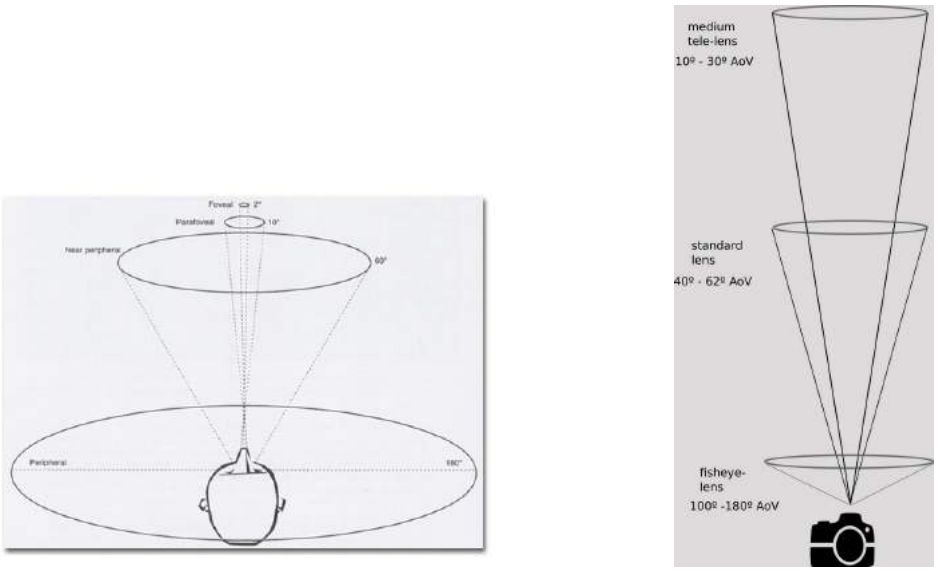


Figure 2.1: Angles of view: human eye vs. camera lenses.

is only useful for sending motion and large-scale objects [Mchugh, 2005]. The camera lenses shown in figure 2.1 vary between 10° for telelens which allows viewing distant objects and up to very wide angle lenses like fisheye lenses which can only capture closer objects and can suffer from lens distortion.

Another important feature to discuss is resolution. The human eye has a visual acuity of 20/20 equivalent to 52 megapixel camera, well beyond most digital cameras with 5-20 megapixels [Mchugh, 2005]. However, it is important to note that our visual acuity decreases gradually as we move from the central vision. We need to look at several regions in order to capture the clarity of a wide range scene.

In addition, the human eye outperforms cameras in dynamic range, which is the adaptation to light and which is the camera's ISO. A human eye has the capability of a dynamic range (24 f-stops [Mchugh, 2005]) and is able to resolve a very high resolution equivalent to 52 megapixel [Mchugh, 2005].

This comparison is meant to let us appreciate the importance of panoramas. A panorama is the output of multiple powerful lenses to capture wide angle views with high resolution and with higher dynamic range.

A good panoramic photo should be seamless, which means that input image transitions cannot be perceived when looking at it. To avoid seams, shooting techniques need to be considered as well as some concepts that will

be briefly covered below.

Parallax Error Parallax error is the difference in positions of two perceived images taken from different view points. This error is a result of taking multiple photos from different viewpoints. In order to avoid this problem, a photographer must move the camera around its optical center/nodal point or no-parallax point. Figure 2.2 shows the difference between turning the camera in a trajectory and what is meant by turning the camera around its center. The resulting images show the difference in positions of the palm tree with respect to the background building when taken from two optical centers in the left versus the same position to the right where camera was rotated around its nodal point. While this seems simple, it is not easy to determine this point and turn a camera around it. To overcome this difficulty, camera designers have provided an accessory called gimbal shown in figure 2.3, upon which the camera can be fixed and turned in all the directions. Using a tripod is usually advised for fixing the gimbal and obtain more stable photos.

Exposure Difference Another important problem that can cause visible seams in the output panorama is the exposure difference which appears as a difference in illumination between scenes. To avoid this problem, it is common to set the camera exposure to a manual mode and choose a fixed exposure for all scenes rather than letting the camera shoot with automatic mode.

2.2 Image Stitching

Once photos are taken separately, the next step is to combine those images together using image stitching. Image stitching algorithms have an established history and such algorithms come embedded in most digital camera and smartphone nowadays. The following sub-sections are meant to be a tutorial on image stitching basics starting from the perspective camera, motion models, image alignment, projection surfaces and finally blending images together.

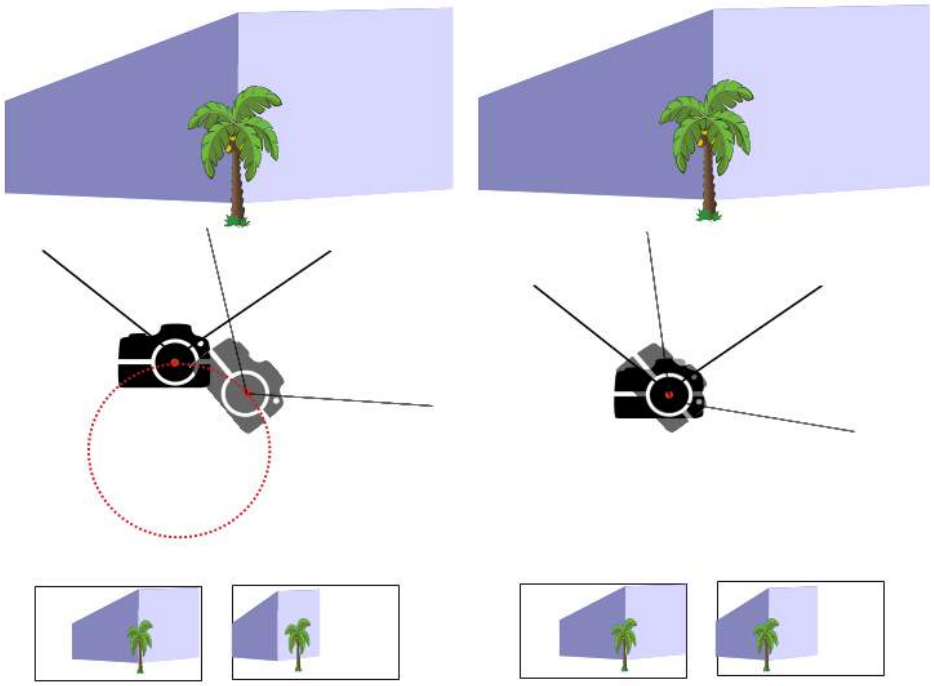


Figure 2.2: Panoramic shooting techniques. Left will create parallax, right is turning around the nodal point, therefore will be parallax-free.

Figure 2.3: Camera gimbal for a parallax-free panorama

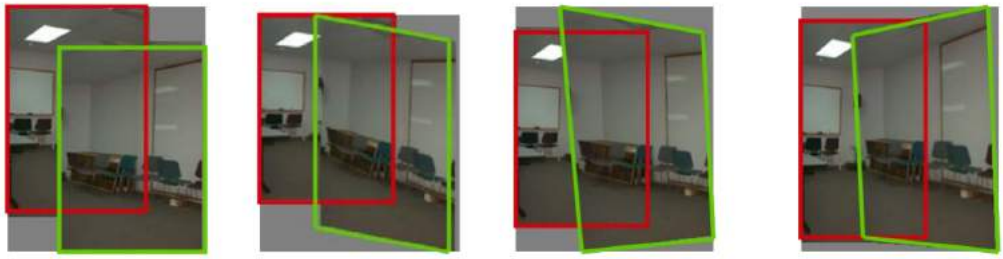


Figure 2.4: Basic transformations applied in image stitching [Szeliski, 2010]. From left to right: translation, affine, perspective and 3D rotation.

2.2.1 Motion Models

Motion models are transformation matrices relating, in our case, one image to another for registration. Figure 2.4 shows examples of motion models that are used in image stitching.

While a transformation can be just a translation and rotation in 2D (i.e. a rigid transformation), in practice to obtain a panoramic image from photos taken freely in the 3D space, projective transformations are used.

Figure 2.5 illustrates the projective transformations between two images in a number of situations. In panoramic photography, we are most concerned the case shown in figure 2.5 (b) in which the two images were taken from the same optical center. Whereas, figure 2.5 (a) is an example of a homography used if the camera was not rotated around its nodal point or in the case of multiple cameras that take the scene simultaneously as we will see later in panoramic videos.

Perspective Camera Model

Camera calibration is crucial in many computational photography applications. Image stitching is not an exception. The first step to be able to make correspondences between images is to recover the camera parameters. Before explaining this step, we introduce the perspective camera.

The perspective camera model or the pinhole camera model is based on 3D projection which maps a point in 3D to 2D plane. In other words, it maps a point in the real world to a pixel in the image plane. A simple yet very important model of the pinhole camera was first introduced by the italian artist and architect Flippo Brunelleschi in the early 15th century [Forsyth

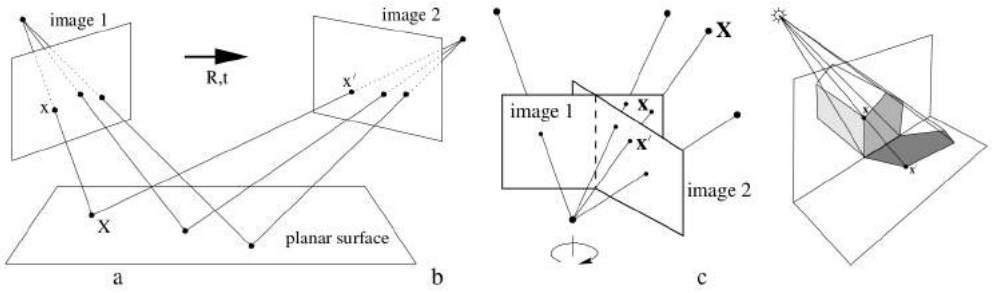


Figure 2.5: Examples of the projective transformation between 2 images from the reference book "Multiple View Geometry" [Hartley and Zisserman, 2003]. (a) Projective transformation (homography) calculated of two images taken from different camera positions onto a common plane. (b) A projective transformation between 2 images taken from the same optical center. (c) A projective transformation between one plane of a 3D object and its shadow on another plane.

and Ponce, 2011](see figure 2.6). After he painted Florence Baptisery with the help of a vanishing point, he used a camera obscura/pinhole camera model to compare the accuracy of his drawing to the actual building. He did that by making a hole in the middle of his drawing and placed a mirror in front of the painting while standing in front of the actual building. Then by moving the mirror into and outside his field of view, he saw the reflection of his drawing through the hole and the building itself and he could verify that his painting was very realistically accurate [Wikipedia Contributors,].

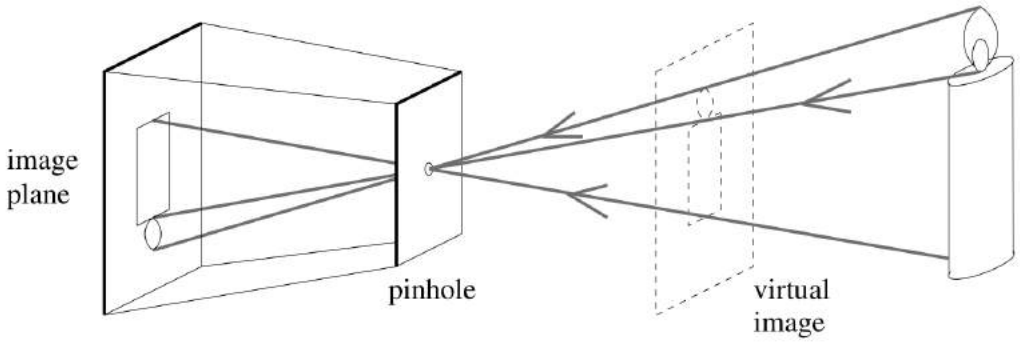


Figure 2.6: Pinhole Imaging Model as demonstrated in "Computer Vision: A Modern Approach" [Forsyth and Ponce, 2011].

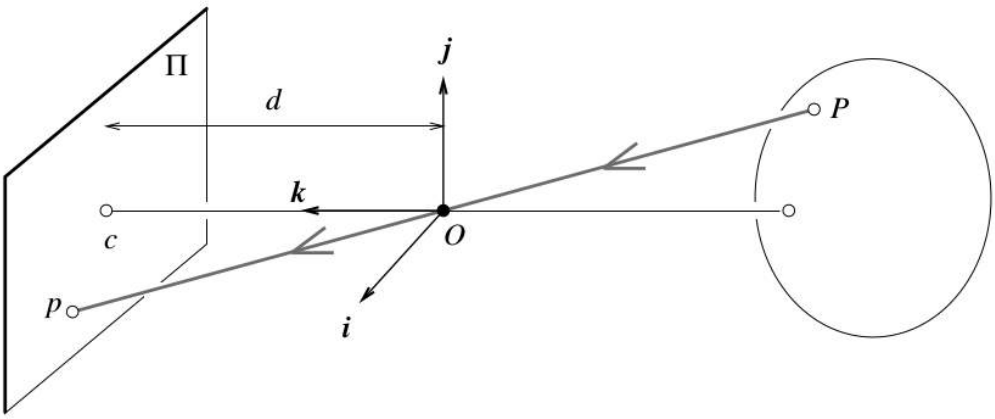


Figure 2.7: Perspective projection notations [Forsyth and Ponce, 2011].

Figure 2.7 introduces the mathematical notations of a perspective projection using the pinhole camera model. P represents a 3D point with coordinates (X,Y,Z) and is aligned with the point O which represents the camera center (pinhole origin) and with p with coordinates (x,y,z) which is the 2D point on the image plane. This collinearity implies that $\vec{Op} = \lambda\vec{OP}$. And since p lies on the image plane, we have $z = d$. Therefore:

$$x = \lambda X, y = \lambda Y, d = \lambda Z \tag{2.1}$$

$$=> \lambda = \frac{x}{X} = \frac{y}{Y} = \frac{d}{Z} \tag{2.2}$$

Camera Parameters

In order to map the 3D point to the 2D space, we use homogeneous coordinates. Homogeneous coordinates are used in projective geometry to represent N dimensions to $N+1$ dimensions by adding a new dimension w. For example, in our case if we need to represent point p in the 3D space, we convert a point from its cartesian coordinates (X,Y) to (x,y,w) in homogeneous coordinates such that $X = \frac{x}{w}$ and $Y = \frac{y}{w}$. Homogeneous coordinates refer to the fact that they are scale invariant since any point (aX,bY) will still be represented with the same coordinates in homogeneous space. Figure 2.8 assumes a normalized image plane which is at a unit distance from the pinhole camera origin [Forsyth and Ponce, 2011]. This allows us to define the point p on the image plane or the physical retina as $\hat{p} = (\hat{x},\hat{y},1)$ in homogeneous coordinates.

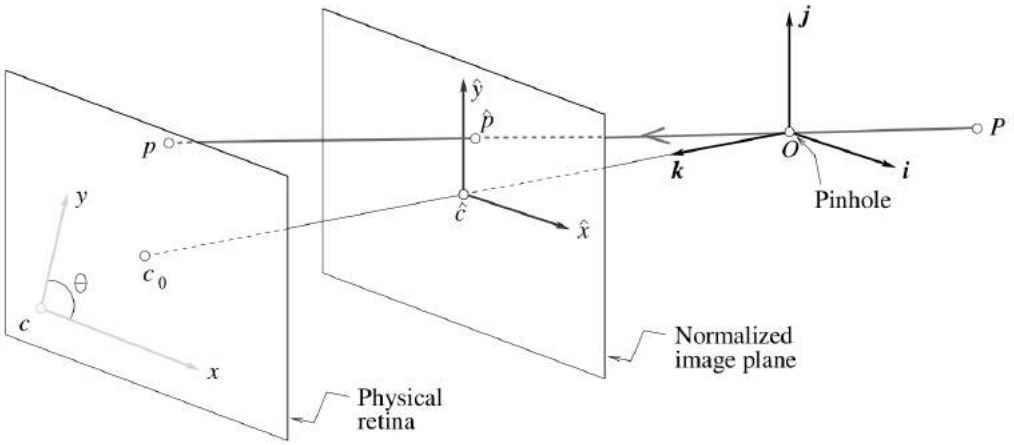


Figure 2.8: Physical and normalized image coordinate systems [Forsyth and Ponce, 2011].

Building on this concept, we introduce two types of camera parameters that are needed in order to know the mapping between the real world and the image plane, which are called intrinsic and extrinsic camera parameters. The intrinsic parameters concern the camera internal parameters which are the focal length, the camera optical center and a skew parameter. The extrinsic parameters are the position of the camera with respect to the scene, which can be defined through rotation and translation.

In order to calculate a point p on the image plane using the perspective camera model we use the following equation:

$$p = \mathbf{K}[\mathbf{R}t]P \tag{2.3}$$

$$\mathbf{K} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

where P is the 3D point in the world coordinates, $\mathbf{K}$ is the intrinsic camera matrix with focal length f_x and f_y and camera center c_x and f_y and the skew parameter s. The extrinsic camera parameters are used to map 3D to 2D coordinates and are represented by [$\mathbf{R}$ t] where $\mathbf{R}$ is a rotation matrix and t is a translation vector.

2.2.2 Pair-wise image alignment: Pixel-based vs. Feature-based

In order to use motion models for alignment, we first need to find matches between images. There are generally two ways to do this:

Direct/pixel-based method

Direct methods can use a brute-force technique or a block matching technique to find image correspondences. The function used for matching is usually a cost function that minimizes an error metric between two given images. The error metric can be defined on the pixel intensities using methods like sum of square differences or a bias and gain model that exploit exposure difference between images. It can also be done through correlation to maximize the product between a pair of images. More details are found in [Szeliski, 2004].

The disadvantage of pixel-based methods is mainly in their difficulty to converge especially with images that have only a small intersection (usually 20% to 50% in panoramas). Therefore, feature-based methods are generally used for image stitching methods.

Pairwise feature-based method

Pairwise feature-based approaches rely on detecting features in an image and then matching feature descriptors using a function such as nearest neighbours. A good feature is usually one that avoids the aperture problem explained in figure 2.9, which happens when a feature of one image is not sufficiently discriminating and thus is more difficult to be matched to only one feature in another image.

Multi-scale oriented patched (MOPS), gradient local orientation histogram (GLOH), scale-invariant feature transform (SIFT) [Lowe, 2004], speeded-up robust features(SURF) [Bay et al., 2008] are all efficient feature descriptors that are considered sufficiently robust, with SIFT and SURF being the most widely used.

Once features have been detected and added to descriptors, it is possible to compare images to find the best matches. A 2D homography can then be calculated using a model fitting technique such as RANSAC [Fischler and Bolles, 1981] which iteratively attempts to find the plane that maximizes the number of inliers.

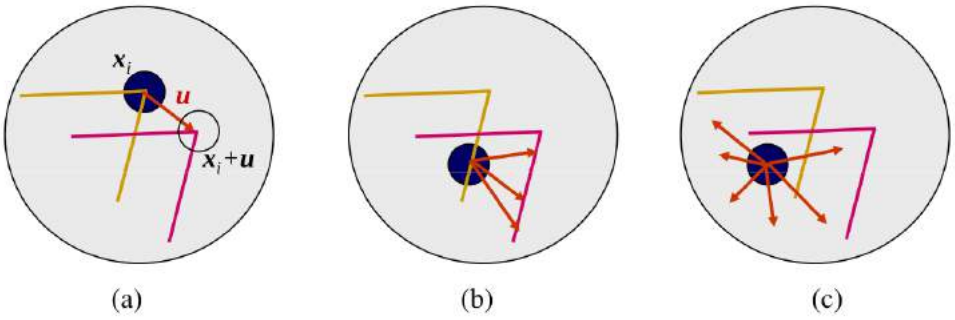


Figure 2.9: Demonstration of the aperture problem in feature matching [Szeliski, 2010]. Figure (a) shows an example of a good feature point that can easily be tracked since it falls on a corner. Figure (b) is a point on a line that is not sufficiently unique and it is a typical example of the aperture problem. Finally, figure (c) is defined on a texture-less region and is impossible to track.

2.2.3 Global alignment: Bundle Adjustment

So far, we have obtained a number of pairwise matches. But a panorama can have a large number of viewpoints that need to be globally matched with the least error. A global alignment is thus a necessity to have a globally coherent output.

Bundle adjustment is a method for global alignment that solves for all camera parameters jointly rather than pairs of cameras. Images are added one at a time to the bundle adjuster which initializes this image with the same camera parameters of its best matching image. It then updates the parameters with Levenberg-Marquardt algorithm [Levenberg, 1944] which is a non-linear least square fitting method to optimize the image positions with respect to each other.

2.2.4 Image projection

At this point, images are not yet placed together. In order to do this, a projection surface is chosen according to the horizontal and vertical fields of view covered when filming and images are projected one by one to that surface with the knowledge of the camera parameters and the projection surface's coordinate system such as cylindrical, conical, fisheye or more. For

more discussion of the projection surfaces, please refer to appendix A.

2.2.5 Blending

Placing images into a common surface is not the final step, instead a seamless single image is required. This process is called blending, which is, as the name implies, merging images together. A good blending method would create as smooth as possible image transition while preserving important details of the original images [Burt and Adelson, 1983]. To achieve this, there are two important factors, the first is the blending function and the second is the choice of a blending mask. In this section, we present various blending methods from simple averaging to more advanced blending techniques along with a discussion of blend masks.

Basic blending function: Alpha blending/ Feathering

In order to understand blending, it is important to consider the basic case of a weighted sum that can be also called feathering or alpha blending [CMU, 2005]. Given two intersecting images that need to be blended into one image, the easiest way to do this is do an average weight in the pixels of intersection.

$$I_{blend} = I_{left} + I_{right}$$
$$I(x, y) = (\alpha R, \alpha B, \alpha G, \alpha)$$

When α is set to 0.5, the blending function works as a simple average, otherwise α can be chosen according to the image that has more importance. The problem with this approach is it causes a high ghosting effect, hence choosing a window for blending can reduce this effect.

Multi-band blending

Burt and Adelson [Burt and Adelson, 1983] were the first to introduce a blending function that produces seamless output. Although published as early as 1983, their method is still the pioneer in image blending for panoramas. The strength of the method relies on the use of image pyramid equivalent to the method proposed by Crowley [L. Crowley, 1981]. Their blending function is based on weighted average method described earlier in equation 2.2.5 and demonstrated by figure 2.10. They observed that the width of the transition zone between two images is ideally chosen according to the

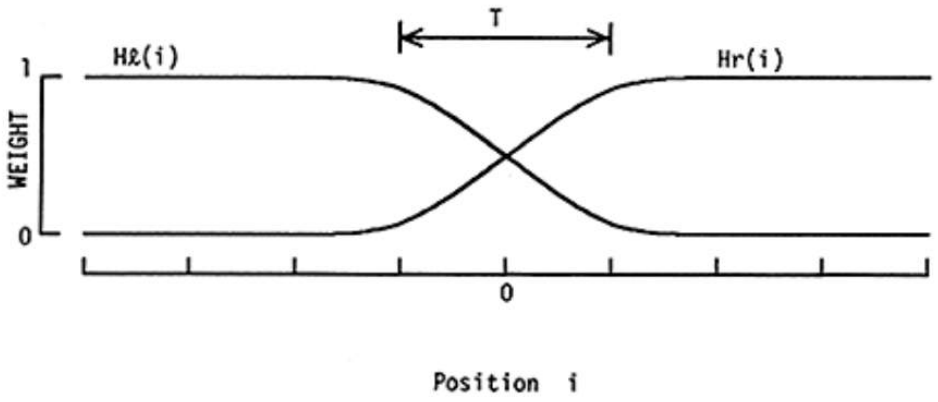


Figure 2.10: A demonstration of 1D weighted average function that can be used to avoid seams [Burt and Adelson, 1983].

size of image features. This is because, if it is small compared to image features, the transition will not look smooth enough, while in the opposite case ghosting artifacts will appear. Therefore, they suggested to subdivide each image to a sequence of low-pass filtered images in order to target the variety of features sizes. At each level of the sampled image, images are blended using weighted average within a transition zone that is proportional to the size of the feature at this level. Once all levels have been visited, the results of blending at each stage is summed up.

To formulate the problem mathematically, let us assume there are two

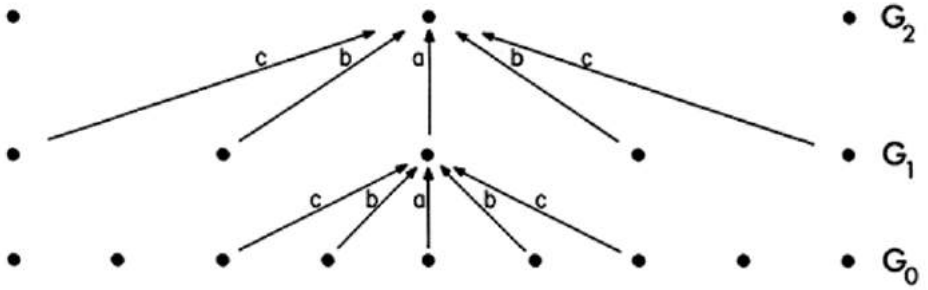


Figure 2.11: 1D example of the down-sampling reduce function [Burt and Adelson, 1983].

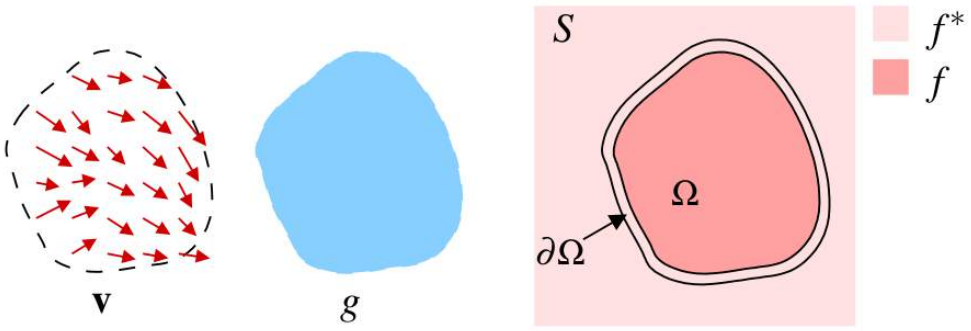


Figure 2.12: Guided interpolation notations [Pérez et al., 2003].

images, one to the left I_l and the other on its right I_r. Let H_l and H_r be weighting function that decreases monotonically from left to right (see figure 2.10 and right to left respectively to give more weight to the I_l or I_r respectively and less the farther away. Assuming i is a given pixel location and u the pixel at the transition line, a blended image I_{lr} obtained from weighted averaging I_l and I_r can be expressed as follows:

$$I(i) = H_l(i-u)F_l(i) + H_r(i-u)F_r(i) \tag{2.5}$$

If we decompose an image I to $G^0.....G^N$, we can apply equation 2.5 at each stage. This is done by a $REDUCE$ function (see 2.11 for a 1D example) from level 0 to level N:

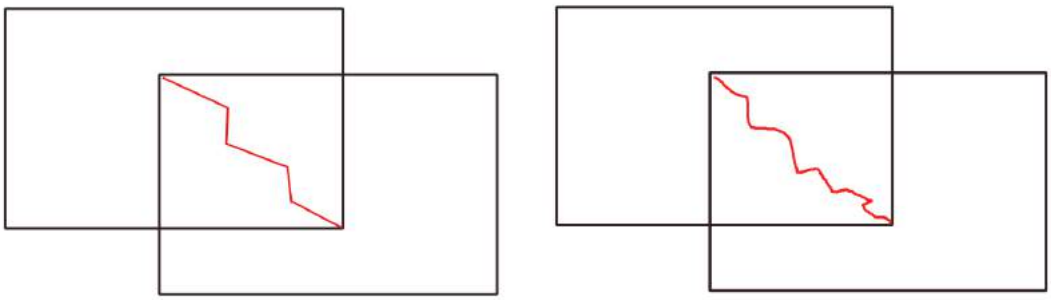
$$G_l = REDUCE(G_{l-1}) \tag{2.6}$$

Then all levels are summed up again to obtain the final image:

$$G_l = EXPAND(G_{l-1}) \tag{2.7}$$

Poisson blending

A more recent blending method is Poisson blending published by Perez et al. in 2003 [Pérez et al., 2003]. The method is especially suited for image compositing in the purpose of altering an image content. Figure 2.12 explains the formulation and notation. Given an image S, we would like to paste another image in the region Ω, we would like to obtain a seamless image at the boundary $\delta\Omega$. First, we define f^* as a known scalar function representing



(a) (b)

Figure 2.13: Simplified demonstration of seam selection for blending a pair of images: (a) Voronoi mask. (b) graph seam cut.

the intensities of image $S - \Omega$ and f being the unknown and defined over Ω. Then we calculate the gradient of the sub-image Ω and we obtain the vector field v. Now the minimization function can be solved using Poisson equation with Dirichlet condition:

$$\min_f \iint_\Omega |\Delta f - v|^2 with f|_{\delta\Omega} = f^*|_{\delta\Omega} \tag{2.8}$$

A discrete version is derived by assuming S and Ω are finite sets of points and where the gradient will be calculated through a window of neighbours that are subtracted from the pixel value.

Seam selection

We have discussed the most important blending functions. The second factor is choosing a good transition zone that we sometimes call a blending mask or seam selection. It is possible to just choose the middle line of the overlapping region, thus it is highly likely that a seam will appear due to the regularity of the line. A better approach is using a Voronoi diagram, which can be sufficient in some cases, however it ignores the image contents and therefore can cause the loss of some information from the original images. Alternatively, graph-cut is a content-aware seam selection approach that uses dynamic programming to optimize the cut with respect to image contents.

Although, graph-cut is the best solution for a panoramic image, it is not usable in panoramic videos, since contents will be different from one frame to another and hence a lot of flickering will be produced temporally. Figure 2.13

show an approximation of what a voronoi mask will look like versus a graph cut.

2.2.6 Recognizing Panoramas

So far we discussed the steps of image stitching with its different approaches. In this section, we present the first work that has been done for panorama recognition by Brown and Lowe in 2003 [Brown and Lowe, 2003]. An automatic panorama construction was presented in this work, which means given a number of images, identify images that match to form a panorama without the need to specify an order or any other input restriction. This approach has become the standard and is the one available in nearly every stitching software or library such as Hugin [PanoTools developers and contributors,] and OpenCV's stitching module.

We have already discussed two possible alignment techniques, the direct one which is pixel-based and the feature-based one. This paper uses a Scale-Invariant Feature Transform (SIFT) [Lowe, 1999, Lowe, 2004] keypoint detector for feature extraction which is partially invariant to affine transformation. By assuming the panorama was taken by rotating the camera around its optical center, the authors define each camera parameter using 3 rotation angles $\theta = [\theta_1, \theta_2, \theta_3]$ and a single focal length f. Therefore, a pairwise homography relates each pair of images: $u_i = \mathbf{H}_{ij} u_j$.
Afterwards, a feature matching step is done where each feature vector in an image is matched to k nearest features using a k-d tree. Each image is then matched to m potential matching images that share the largest number of feature matches. RANSAC [Fischler and Bolles, 1981] is used to optimally fit a plane to the feature points, where the highest number of feature points agree afterwards a probabilistic Bayesian model is applied to verify the match. Connected sets are then matches to identify all images belonging to a certain panorama and reject others. At this stage, matching has depended on a pair-wise feature-based homography. This is likely to accumulate errors since no global alignment has been considered. As explained earlier, bundle adjustment is used to optimize the camera parameters using Levenberg-Marquardt [Levenberg, 1944]. Finally, blending has been done using multi-band Laplacian pyramid by Burt and Adelson [Burt and Adelson, 1983]. Results of this approach has been tested and shown on several examples in the paper including 360° panoramas.

2.3 Extension to Panoramic Videos

The very intuitive method for creating panoramic videos is to use the baseline image stitching algorithms on each frame of the panoramic video. However, this solution is far from being robust. The first reason is due to the way we capture panoramic videos, which is usually done using multiple lenses that cannot be put at the same optical center, which creates high parallax errors. Another reason is the lack of synchronization between the different cameras which causes visual artifacts in the final video. Finally, it is important to note that solving those problems in a frame by frame basis will not be sufficient and will result in temporal inconsistency and flickering when moving across the video.

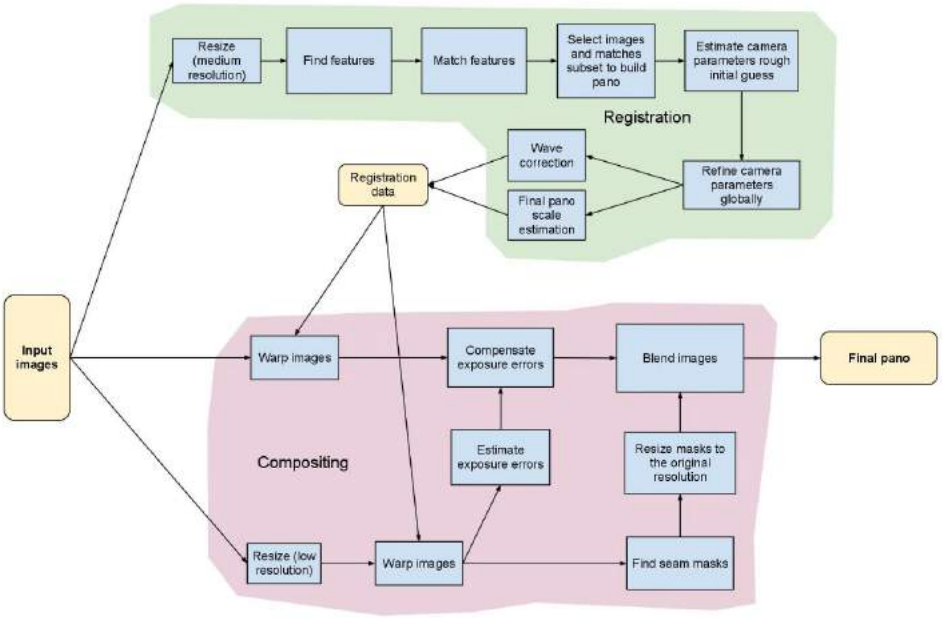


Figure 2.14: Pipeline of the stitching module in OpenCV.

Despite all these drawbacks, the image stitching baseline algorithms are an essential part of any video stitching algorithm. Two widely used available open-source libraries are OpenCV and PanoTools. OpenCV implements a pipeline very similar to that described in [Brown and Lowe, 2007] and is

shown in OpenCV documentation through figure 2.14.

Panotools provides excellent stitching options by grouping several libraries that implement the different steps of stitching. It can be used with scripts or through its graphical interface called Hugin [PanoTools developers and contributors,]. It is an open-source library, therefore source code is also available.

Methods dedicated to produce panoramic videos are covered in chapter 3, which are based on the baseline methods explained in this chapter.

2.4 Summary

In this chapter, basic panorama photography and image stitching has been covered. We explored the different challenges of creating panoramas such as parallax errors and exposure differences. The stitching steps have been explained from feature extraction and matching, then projection to a compositing surface and finally blending. In the last section, panoramic video issues have been discussed including capture systems, processing panoramic videos and different display systems. The next chapter will offer more profound details about important concepts in panoramic videos as well as cover in details some of the most successful state-of-the-art research.

Chapter 3

Video Stitching Design Challenges

In this chapter, we discuss various aspects and challenges in panoramic videos and their solutions in the literature with a focus on one method implemented within this thesis as a basis for our work.

3.1 Panoramic video capture and display

3.1.1 Capture Systems

As explained in chapter 2 we can perfectly shoot a panoramic image with a single camera by rotating it around its optical center and thus avoiding parallax errors. An exception to this will be a scene with high movement which can still suffer from ghosting errors when the object move between different viewpoints.

Unlike a still image, filming a panoramic video needs all views to be taken simultaneously so we can have a panoramic frame of all the scene at each time t. Therefore, the first challenge is establishing an acquisition system that can take multiple videos at the same time covering a large field of view up to 360° spherical videos.

There exists a high market competition in the design of panoramic cameras today. Google, Facebook, GoPro and more companies have cameras available for purchase for shooting panoramic videos varying in size, price and number of cameras. Figure 3.1 show examples of 3 cameras from Ricoh,

(a) (b) (c)

Figure 3.1: Commercial 360 degree cameras. (a) Ricoh Theta with two fish-eye lenses. (b) Omni GoPro with 6 cameras. (c) Surround 360 by Facebook with 17 cameras.

Figure 3.2: An unstructured camera setup used by Perazzi et al. [Perazzi et al., 2015]

GoPro and Facebook that vary from 2 cameras to 17 cameras. Hand-held Ricoh Theta and Omni GoPro are available for purchase while Facebook's Surround360 is made for research with an assembling guide available online. All these cameras have a particular setup and are usually synchronized but they have their limitations in resolution, field of view and parallax errors. It is also possible to build one's own rig and fixing multiple cameras on it such as in figure 3.2. This offers a more flexible option with respect to the camera types, their number and their positions, however there is a number of additional issues to be dealt with such as synchronization and camera parameters estimation.

Synchronization Problem

Synchronizing a number of cameras for video shooting is not straightforward. A hardware synchronization mechanism is embedded in commercial cameras whereas an experimental setup needs a solution. Disney researchers [Wang et al., 2014] propose a temporal alignment approach based on feature extraction and matching along with an interactive tool that allows an artist to choose frames that are in correspondence to enforce a certain path. Nguyen and Lhuillier [Nguyen and Lhuillier, 2016] suggest a bundle adjustment method that estimates time offsets for to achieve synchronization in addition to the refinement of 3D camera parameters. Finally, it is possible to include the sound of a clap while shooting the videos and use the accompanying audio file for synchronization.

3.1.2 Display Devices

The current options for panoramic video display include regular screens with a plugin that allows mouse navigation such as the ones present on YouTube and Facebook today, as well as offline photo viewers like Windows 10 Photo Viewer.

Another option and a better one is the head-mounted displays(HMD) which are worn on the head and eyes and allow an immersive environment simply by moving the head in different directions as seen in figure 3.3. Google cardboard is another cheap solution for viewing panoramic videos on a smartphone.

Finally, more expensive alternatives include cave automatic virtual environment (CAVE) systems which consists of a three to six walls with projectors

in a room-sized cube. Users wear a HMD display to view the stereoscopic scene and can optionally have a hand controller for navigation. Figure 3.3 shows a photo of UCL's CAVE which has 5 projection walls. Dome movie theatres is another display option which allows a large audience to have a fully immersive experience (see HP's Antarctic dome in figure 3.3).

While every type of display can offer a different experience, it appears that HMD offers the best one according to MacQuarrie and Steed 2017 [MacQuarrie and Steed, 2017]. In a user study, they compared HMD, a regular TV and SurroundVideo system based on a number of factors including a participant sense of engagement, spatial awareness, their feelings of enjoyment or fear as well as their attention and feeling overwhelmed about missing an event. Their results varied on each metric/display with a global preference to HMD. Preparing content for panoramic videos for these display devices is an open research topic that is still facing a lot of challenges. Unlike narrower fields of views, with 360 degree videos, users are required to choose their viewing points, which totally depend on the content being viewed. Zoric et al. 2013 [Zoric et al., 2013] conducted a user study to understand the challenges of designing interactive panoramic videos. Their main conclusion was about the benefit and the design implication of giving users the possibility of controlling the view, instead of being obliged to see a subjective viewpoint imposed by the cameraman. Co-located and remote social viewing where people can share their chosen view with others was another important aspect that could be of interest to many users.
Alternatively, some methods offer a solution to this problem by automatically choosing a view and therefore produces a natural video from the panoramic 360° spherical video. An example is the work presented by Hu et al. [Hu et al., 2017] which propose an intelligent agent that will automatically choose viewing angles for sports viewing. The method works well for videos having clearly salient objects but becomes less efficient when there is equal saliency within the video. A similar idea is that of Su et al. [Su et al., 2016] who produce a regular video from 360° video by usinga dataset of 360 videos and learning an optimal human-like camera trajectory using dynamic programming. These methods do not allow a free viewpoint for the user and are not meant to be used for a HMD.

Figure 3.3: Top: Oculus Rift to the left and Google Cardboard to the right, used for viewing panoramic videos. Bottom: projection display rooms, UCL's CAVE to the left and HP's Antarctic Dome to the right.

3.2 Video Stitching

Video stitching involves the process of taking multiple videos covering a large scene to provide a single seamless video. Depending on the application, this video can be just a wide angle video or a spherical 360° video. It can also be a 2D video or a stereoscopic video.

In chapter 2, we covered the basic image stitching with a brief discussion of the challenges introduced by panoramic videos. We give more details along with existing solutions in this chapter. We start with an overview of a typical video stitching software as described by VR post production specialist, Stephen Les [Les, 2015]. His description is based on his use for two of the most popular commercial video stitching software, Kolor Autopano and VideoStitch. The regular workflow is shown in figure 3.4 which shows how challenging the task is. The process involves a set of pre-processing steps using other software applications such as *360CamMan* and *Adobe Premiere* which would organize and analyze the videos prior to importing them to the stitching software. Afterwards, an attempt to auto synchronization is made that will have to be done manually in the case of failure. The clips are eventually sent to the automatic video stitcher, which might succeed or not. In difficult cases, processing frame by frame will be the alternative. The pipeline ends by a set of post processing steps such as video stabilization and exposure compensation and finally further post processing can be done according to the application.

Figure 3.5 shows the different components present in most video stitching software. Components with red color are the ones that are handled differently than the basic image stitching which we will focus on in this chapter. Camera parameter estimation is not as straightforward as in the case of still panoramic photos, since panoramic videos are usually taken using multiple cameras that can be structured or unstructured. Parallax is almost unavoidable in panoramic videos for the same reason: multiple capture devices impossible to place such that optical centers are in the same position, therefore parallax compensation or removal is a very important aspect in video stitching. And finally, since it is no longer a single photo but a series of frames in time, video issues need to be dealt with, such as stabilization. Temporal artifacts are produced due to the frame-by-frame processing and the complex deformation steps involved in each time frame.

Therefore, we explain these challenges in details with the state of the art solutions.

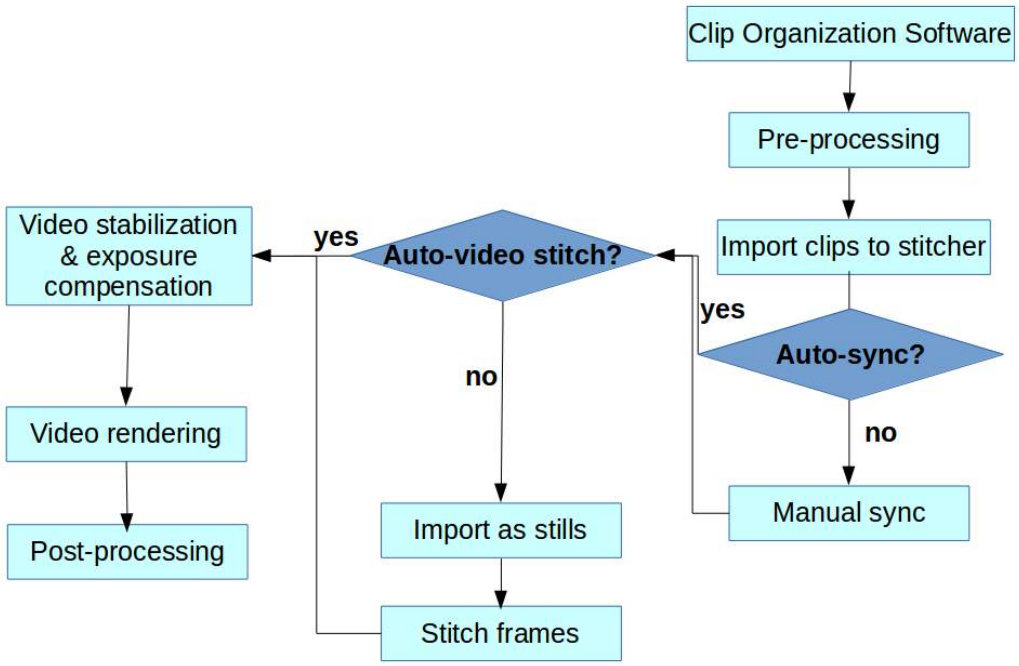


Figure 3.4: Workflow of commercial software for video stitching as described by Stephen Les in [Les, 2015]

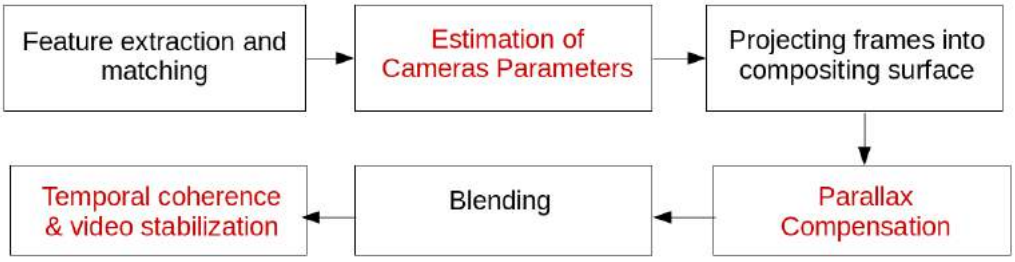


Figure 3.5: Different components of video stitching.

3.2.1 Camera Calibration

As with many computer vision and computational photography applications, panoramic video camera calibration is a crucial pre-processing step. We discuss here categories of methods used in the state of the art video stitching methods for multiple camera calibration.

Feature-based Calibration

A large category of video stitching in the literature use homographies to estimate camera parameters [Zheng et al., 2008, El-Saban et al., 2010, Perazzi et al., 2015, Jiang and Gu, 2015, Kim et al., 2017]. This process involves the extraction of features from different views then match pairs of images. A model fitting approach such as RANSAC [Fischler and Bolles, 1981] is used to obtain the best plane fitting features and allows the calculation of a 2D perspective transform (homography) between both matched image planes (see Figure 2.5(a) from chapter 2 for an explanation of the projective transformation of 2D images taken from different viewpoints). The pair-wise homography is used to estimate camera parameters and bundle adjustment is used to minimize the accumulated error over all cameras. The main advantage of this approach is being fast and simple and yields good results in most cases.

Checkerboard-like Techniques

Estimating camera parameters based on homographies is subject to inaccuracy since it depends on the step of the feature extraction and matching step. As an alternative, a classical approach can be used which consists in moving a checkerboard object or a similar pattern in front of the cameras to detect corners from different positions and accordingly estimate camera parameters [Tsai, 1987, Heikkila and Silven, 1997, Zhang, 2000]. Variations of this algorithm are employed depending on the camera setup. Bundle adjustment is also used to optimize parameter estimation globally for all cameras. This method is adopted in a number of video stitching algorithms [Lee et al., 2016, Ho et al., 2017] to avoid errors resulting from the feature-based approach.

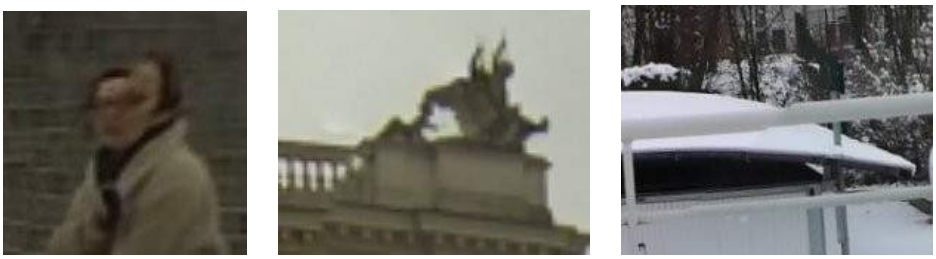


Figure 3.6: Deformation, ghosting and misalignment resulting from parallax problem (dataset by Perazzi et al. [Perazzi et al., 2015] and our own dataset.)

3.2.2 Removing parallax errors and temporal artifacts

Parallax is the difference in the apparent object position in two images taken from two different positions. Ideally, we would like to shoot panoramas by rotating a camera around its optical center. Unfortunately, it is nearly impossible in panoramic videos due to the capture setup (figures 3.1 and 3.2). The resulting error is the most common and highest disturbing artifact in panoramic videos as shown in figure 3.6. Therefore, most of the video stitching methods address this problem in their solutions. Usually the solution falls into one or more of the following categories: mesh optimization, seam cut selection, the blending function or error metric minimization. We discuss these categories with examples in the following subsections.

Grid/Mesh Optimization

Similar to a 3D mesh, it is possible to treat an image as a mesh by dividing it using a grid and treating intersections of lines as vertices that can be optimized in space and time. This idea has been adopted in different ways by a number of video stitching methods in the state of the art for parallax errors removal and temporal consistency.

An example of this approach was suggested by Jiang and Gu [Jiang and Gu, 2015]. After extracting and matching SIFT features, pair-wise homographies are calculated in the spatial domain H_i^S (where i is the camera number) relating neighboring camera views then in the temporal domain $H_{i,t}^T$ (where i is the camera number and t is the frame number in time) to relate frames in time(see figure 3.7. Each image is then divided into an $M1xM2$ grid and vertices are optimized based on a number of terms that enforce alignment and smoothness both in the spatial domain between neighboring cameras

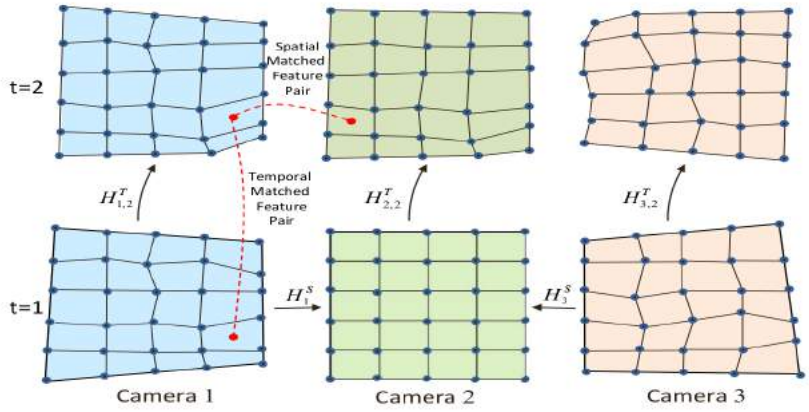


Figure 3.7: Spatial-temporal local warping method by Jiang et al. [Jiang and Gu, 2015].

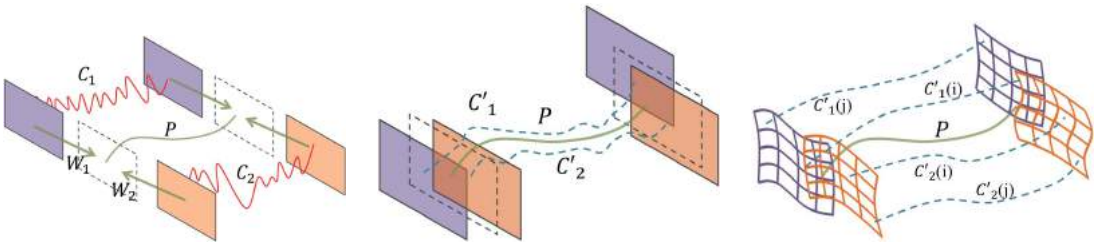


Figure 3.8: Three steps of joint optimal stitching and stabilization by Guo et al. [Guo et al., 2016]

and temporally for each video. More precisely, they use two terms to enforce local alignment of matched features spatially and temporally, this is used to remove the effects of parallax. Their global alignment terms helps keep the regions with no matched features to remain unchanged, while the smoothness terms are added to avoid temporal distortion. In their final cost function, they allow the terms to be weighted to differently according to the user preference and the scene content.

Similarly, Guo et al. [Guo et al., 2016] proposed a method that jointly solves the video stitching and stabilization of videos by optimizing a grid in space and time to overcome parallax. Since grid-based methods highly depend on the detected features' accuracy, they utilize a more sophisticated feature extraction and matching method. First, they divide a frame into a

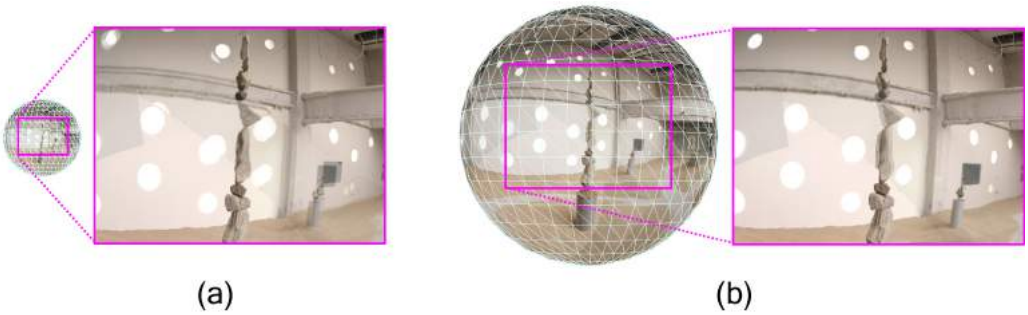


Figure 3.9: Parallax error in (a)2D versus (b)3D spherical projection surfaces [Lee et al., 2016].

5*5 grid and then for each region they choose a local threshold that depends on this region's texture. Feature matching is done between views of the same video and in time, which mean they track features through the video. In addition, they combine stitching and stabilization at the same step where they generate an optimal trajectory for each camera path by optimizing a cost function consisting of 3 terms enforcing smoothness between neighboring camera views and frames in time. Subsequently, frames are then warped towards this optimal path. They finally divide again each frame into 16*16 grid where each cell is treated as a separate camera path that is optimized towards the previously calculated optimal camera path. Figure 3.8 explain the 3 steps of joint stitching and stabilization.

Lee et al. [Lee et al., 2016] propose a 3D grid-based method that differs in many aspects to the methods discussed earlier. Their mesh is generated through triangulation after projecting the their videos on a 3D sphere. They justify their use of a 3D sphere since calculation depth can help remove parallax error as shown in the example taken from their paper (see figure 3.9. A set of points are chosen using feature detection in regions of overlap that are then projected and converted to the spherical coordinated are used in to control the minimization function together with a smoothness term. The method then calculate a salience map from user annotation to improve resolution in regions with higher importance such as a human's face. Figure 3.10 explains the steps of the algorithm.

Another method that relies on 3D reconstruction was presented by Lin et al. [Lin et al., 2016] for video stitching from hand-held phone cameras. After camera calibration, they reconstruct their scene in 3D in the overlap region.

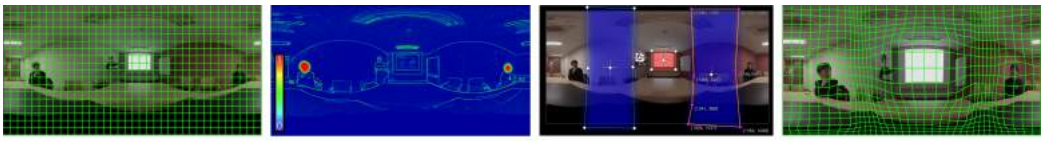


Figure 3.10: Deformation of 3D mesh and salience detection in [Lee et al., 2016].

They adopt a mesh-based warping method similar to the previous methods, but with a difference in the optimization's cost function terms. Since they work in the 3D space, they employ an epipolar term together with a line, a feature and a coherence terms.

Smart Seam Selection

As discussed in chapter 2, Voronoi diagrams and graph cuts are widely used for seam selection. However, to remove parallax in challenging situations in panoramic videos, more adapted approaches have been presented in the state of the art.

Li et al. [Li et al., 2015] proposed a solution to stitching wide angle synchronized videos. Their main idea was based on the choice of two seams rather than one in the region of overlap between two images. They constrain their seam selection spatially and temporally and optimize their solution using dynamic programming. Based on the chosen seams, they build the remaining steps of their stitching algorithm. Features are extracted as edges present on the seams, since object boundaries are likely to have visible parallax errors. Afterwards, they warp images towards each other based on the selected features and do a deformation propagation to make the warping smooth. Finally, they use Poisson Image editing to blend the images together. Their approach seem to outperform many other algorithms in particular cases, mainly with indoor scenes and small distance between optical centers. Their method does not consider the case of multiple image overlaps where double-seam selection might not work and will likely accumulate errors.

As mentioned earlier, Jiang et al. [Jiang and Gu, 2015] associate their grid-based optimization method with a spatial-temporal seam finding method. Their graph-cut method uses pixels in the overlap as nodes and discriminate between two edges, the spatial and temporal ones. They optimize their cost

function such that they choose the best seam cut both spatially and temporally. They also associate higher weights to edges that contains salient features such as humans so they avoid cutting through them.

Kim et al. [Kim et al., 2017] presented a method for smart seam selection and content-aware blending. Their seam selection is based on a number of measurements including whether the seam has an object or not and whether it cuts through an edge. To avoid temporal artifacts due to seam selection at each frame, a partial seam-update function is applied which stretches the seam if an object has been detected. In the same way, blending is restricted to the detection of objects on the seam line.

Blending Function

Blending is an essential method in image and video stitching that aims to produce a seamless output panorama out of the composite input images/frames. The most widely used blending technique for stitching was established in 1983 by Burt and Adelson [Burt and Adelson, 1983] which was explained in chapter 2. Although the method performs well in most of the situations, parallax and motion are still a challenge when it comes to blending. Therefore, some research has been dedicated to provide more content-aware blending methods or temporal methods.

An early panoramic video stitching was presented in 2008 by Zheng et al. [Zheng et al., 2008]. They work on two input video streams taken by webcams. The method pursue the basic stitching of feature extraction and matching. Their main contribution is a new blending function that aims to reduce parallax errors. Their function is basically an alpha blending approach, where they define α as a non-linear function. They show their results on various degrees of motion in scenes. Their future work was planned to extend the approach to several cameras as well as handle exposure difference.

Su et al. [Su et al., 2018] focus their solution on occlusion detection, which as they state, is the main reason for ghosting artifacts. In order to detect occlusion, they create a binary map in which they identify occlusions as feature point pixels that are not identical. They optimize their maps using dynamic programming. They use the occlusion maps to determine a blending strip in the area of overlap between a pair of images. They apply a spatial–temporal Bayesian view synthesis approach to generate their final view.

Temporal alignment

El Saban et al. [El-Saban et al., 2010] exploit temporal redundancy in panoramic videos to speed-up the stitching process. They propose the use of a reference frame where SURF features were detected and matched once, then the same correspondence and alignment were used across the video. After the steps of frame matching, they do feature tracking using optical flow between frames in time. This allows a temporal alignment in addition to the spatial alignment. Their dataset was collected using free-moving hand-held mobiles, thus they do not impose a structure for the camera setup. Their approach is claimed to work well but a lot of advancements have been done since then. Also, the paper shows no visual results so it is hard to judge the effectiveness of their method.

An approach specifically designed for videos taken using two fisheye lenses was done by Ho et al. [Ho et al., 2017]. The challenge with fisheye-based 360 videos is the small overlap region between the two views. Therefore, the authors proposed a rigid moving least squares minimization function to find image correspondences between the two views. This generates a transformation matrix that deforms one image to the other using control points. A score is given to the stitching based on a number of measurements obtained from an empirical experiment. This score is used to identify bad matches that can cause jitters in the resulting video.

3.2.3 Generating high resolution real-time videos for virtual reality content

To this point, we addressed the way video stitching state of the art methods offered solutions to specific challenges. Most of the previous method involve expensive computations that focus mostly on solving parallax errors or temporal instability. in the aim of reducing visual defects. In this section, we discuss a number of cutting-edge methods that attempted to provide solutions for virtual reality (VR) content, which means they will be seen with special immersive equipments (see figure 3.3). These methods usually have different design implications:

- Videos need to be 360° to allow immersion.
- Rendering need to be realtime.

(a) (b)

Figure 3.11: Real-time 360° high-resolution videos solutions. (a) Rich360 solving parallax errors and loss of details by using a deformable 3D sphere and increasing resolution in salient regions [Lee et al., 2016]. (b) Foveated stitching mimics human vision by giving less resolution in the foveal vision and higher acuity in central vision depending on the calculated head direction [Lee et al., 2017]

- Stereo videos can add further feeling of presence and provide a better experience.
- Visual artifacts are less tolerated since they can cause visual discomfort when seen with a VR headset.
- High resolution is crucial, at least in salient regions.

The upcoming subsections provide a survey of a variety of works aimed for VR applications.

Monocular Videos

As previously mentioned in this chapter, authors in [Lee et al., 2016] suggest a framework, Rich360, which focuses on reducing parallax by projecting the video on a 3D sphere that is optimized by mesh deformation. Higher resolution is made in regions of salience importance. Their final goal is to create high quality content for VR.

Another 360 video stitching framework for real-time VR has been proposed by Lee et al. [Lee et al., 2017]. Using a client-server architecture, projection and blending maps are calculated for each camera on the server side, while the client side collects gaze information from head movements captured using Google cardboard's sensor. This information is used to calculate an acuity map which gives a high weight to the central vision and a low weight to the foveal region of the eye. Another saliency map is calculated and both are combined to produce high resolution in the salient regions. The

Figure 3.12: Facebook Surround 360: first and second generations.

method focuses on rendering high-resolution real-time VR but does not provide solution to parallax removal. See figure 3.11 for a demonstration of this method versus Rich360.

Stereo Videos

We agree that a third dimension can add a lot to feelings of presence in immersive environments. However, it is a more challenging problem that requires the generation of two videos for each eye and thus needs special equipment. It also needs more sophisticated approaches for stitching and blending. In addition, it is less tolerant to any stitching error since it is perceived within a 3D headset that can cause discomfort and fatigue for observers if a visual artifact is detected.

There is more than one approach to calculate depth maps to provide stereoscopic panoramic videos. The most popular approaches involve shooting videos using a stereoscopic cameras or using LIDAR scanners to calculate depth maps using light field information. We discuss some recent solutions in the following subsections.

Facebook Surround 360

One of the most interesting panoramic system capture was Surround 360 announced by Facebook in spring 2016 [Cabral, 2016] and open-sourced in summer of the same year [Briggs, 2016]. The system is a combined hardware and software for producing panoramic 360 cameras. The rig, shown in figure 3.12 left consists of 17 cameras Point Grey cameras covering a 360° x

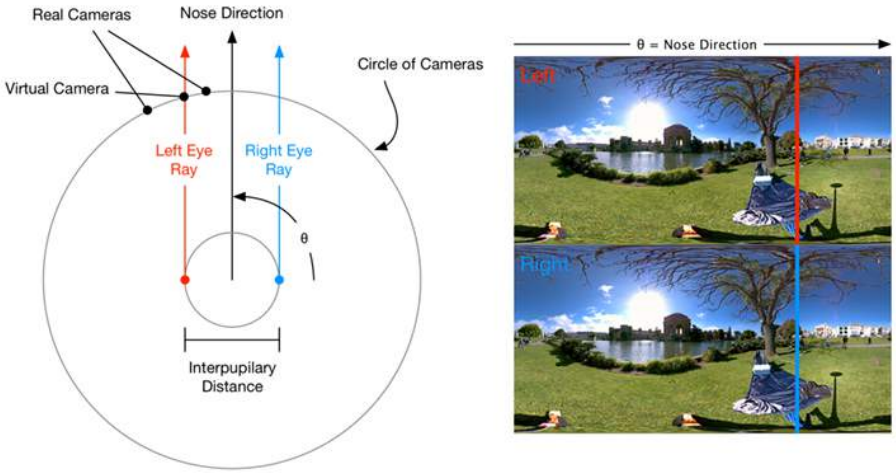


Figure 3.13: Interpolation of virtual camera in Surround 360 and projection to equirectangular surface where every column of pixels is rendered in the direction of the nose. Figure taken from [Briggs, 2016]

180° view, each of which can provide 8K resolution quality. The criteria for choosing the camera was meant to provide efficient filming without getting overheated while functioning for hours.

Their open-source project contain a guide on how to build the rig and all the camera requirements. It also contain the stitching software along with sample data for testing. In order to generate stereoscopic content, the method suggests a view interpolation to generate a virtual camera for each eye between real cameras. To render pixels in the final view, they generate equirectangular maps where each pixel represents a ray going from the interpolated left and right virtual cameras in the direction of the nose (see figure 3.13 for clarification).

The processing pipeline described in [Briggs, 2016] starts in the hardware, where the Image Signal Processor (ISP) of the camera converts raw sensor data into RGB images and apply gamma correction. Afterwards, camera calibration is done using classic checkerboard approach for all the side cameras. Each image is then projected individually into an equirectangular surface. Optical flow is calculated in overlapping regions of two views for parallax compensation. Optical flow is used for the novel view interpolation for the virtual camera. At the top of the rig, there is only one camera, so monocular

Figure 3.14: Removal of the tripod pole and generation of a new view using two cameras at the bottom of the Facebook Surround 360 rig. Figure taken from [Briggs, 2016]

video is generated and optical flow is calculated between the top view and the generated side panorama and used for warping the top to the side views. At the bottom, there are two cameras which are meant to be used to remove the tripod pole (see figure 3.14).

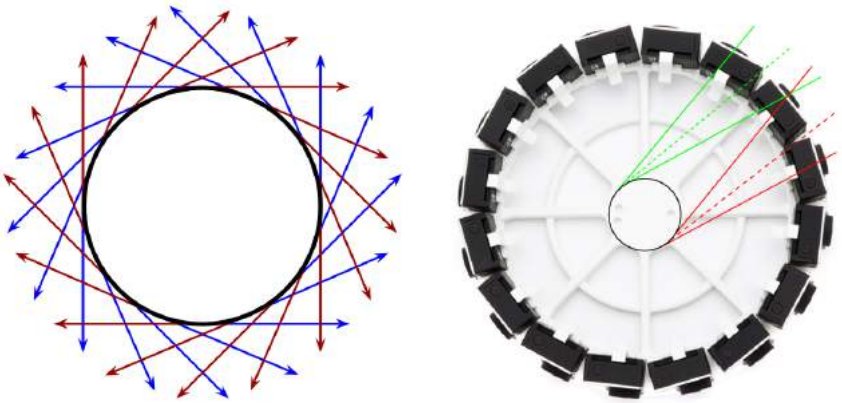
A second generation of Surround 360 has been released in 2017. The new camera is more portable than the first rig and it has 24 lenses, therefore is more capable. It is also meant to be available for purchase unlike the initial one.

Google Jump

Google has also published a method [Anderson et al., 2016] for stereo video generation from videos taken by Google Jump. The method is very similar to that of Facebook since it also relies on optical flow for parallax compensation and temporal stability. However, since their system uses GoPro cameras with rolling shutter they treat the problem in 2D optical flow correspondence manner than the 1D stereo problem. They project their videos later to the omnidirectional stereo circle seen in 3.15.

Samsung 360 Round

Samsung released a new 360° camera with 17 camera fisheye lenses arranged in stereo pairs and a single camera in the top (see figure 3.16). Limonov et al. [Limonov et al., 2018] devised a pipeline that aims to stitch videos taken from this camera and render it in realtime to be seen with a VR



(a) (b)

Figure 3.15: (a) Omnidirectional stereo (ODS) projection rendering stereo rays in all directions using tangents to the circle. Left and right image rays are shown in different colors. (a) Google Jump rig with ODS overlaid showing rays for left and right eyes. (Figures taken from [Anderson et al., 2016])

Figure 3.16: Samsung Round 360 VR camera specifications and measurements (Figure taken from [Limonov et al., 2018]).

(a) (b)

Figure 3.17: Figures taken from [Aggarwal et al., 2016]. (a)Coffee filter camera by Aggarwal et al. (b)Using a camera to shoot the panoramic video reflected in the suggested mirror.

device. Similar to [Anderson et al., 2016], they use the ODS projection as an approximation for the stereoscopic 360 projection to solve the parallax problem and stitch videos for the left and right eyes. They optimize their code using parallel processing on GPU to make it faster.

Single camera and one mirror

Aggarwal et al. [Aggarwal et al., 2016] proposed the generation of a stereo 360° video using any single standard camera along with a special mirror they invented (see figure 3.17). The mirror aims to reflect all the rays needed to render an omnidirectional view for each eye. This is done through the petals shown in figure 3.17). They also provide a comparison between their method and a number of other state of the art solutions showing that their method offers a good compromise in quality and resolution with an affordable capture system.

Light-Field Scanner with Multiple Cameras

Another popular approach to obtain 3D panoramic videos is using LIDAR scanners that capture light field data at very high speed and for long dis-

Figure 3.18: The VR capture system designed by HypeVR and Velodyne (figure taken from [Juchmann, 2015])

tances. It can be associated with panoramic regular cameras that capture the 2D colour information of the scene. An example was a VR rig that was constructed by the American startup HypeVR in collaboration with Velodyne LiDAR in Silicon Valley [Juchmann, 2015]. The rig consisted of 14 Red Dragon cameras of 6K resolution each and covering a 360° scene associated with Velodyne's spinning LiDAR scanner (see figure 3.18). Their goal was to create a seamless 3D video by introducing a dense 3D point cloud extracted from the LIDAR sensor and map the point cloud to the 2D videos. With this high resolution and powerful cameras together with the LIDAR's long distance estimation and accuracy of the 3D points, the outcome should be expected to be highly promising. The drawback is that the system is very expensive and cumbersome.

3.3 Panoramic Video from Unstructured Camera Array

In this section, we present a paper published by researchers in Disney Zurich [Perazzi et al., 2015]. The main idea of this work was to establish a parallax error metric that is later used to optimally order pairs of panoramic frame views to minimize this error function. This method was implemented, tested and output was used for the proposed method of this thesis.

The choice of this algorithm to be the baseline of our experiments since it was similar to the initial parallax compensation idea we intended to implement. Also because of the availability of the datasets and the output of their method. And finally because the results were the most promising at that time. C++ language and OpenCV library were used to implement the method and the source code will be made available online.

Reference Projection

The first step of this algorithm is the creation a reference projection. By this term, the authors meant to do the necessary steps of image stitching once and for all. They proceed on a chosen representative frame with the baseline stitching steps: feature extraction, pairwise feature matching and homography-based camera estimation explained in details in chapter 2. These steps will never need to be recalculated, instead these registration data and camera parameters will be re-used for every other frame.

Motion Estimation in Overlap Region

As explained earlier, parallax is the apparent difference in position of an object in a scene taken from two different view points. Parallax errors cause very disturbing artifacts in panoramic videos such as misalignment and ghosting. Thus, a lot of research is dedicated to find a solution to this problem as we discussed earlier in this chapter. Perazzi et al. [Perazzi et al., 2015] suggest a solution that falls into the category of error minimization solutions. In order to calculate their parallax error function, they first calculate an optical flow between the overlapping regions of of images. This flow is later used to warp an image to another for parallax compensation.

We consider the case of a pair of images and later extend for any number of overlapping images. Given image Ii and image Ij with overlapping region

$\Omega = Ii \cap Ij$, the regions of overlap will be δIi and δIj. The motion field μ_{ij} is calculated using optical flow (in this paper Brox 2004 [Brox et al., 2004]).

Parallax Error Function

In order to address the problem of parallax errors globally, Perazzi et al. suggest an error function that will be calculated between each pair of views in the reference projection and will be used for optimal warp order. This step is done only once and used for the rest of the video frames.
The error function is calculated as follows:

- Given M views/cameras and K pairs matched, each pair is considered if the overlap exceeds 10% of the whole image.
- Optical flow is calculated between pairs in the overlapping regions, thus we calculate μ_{ij} for a given pair I_i and I_j in the overlapping regions δI_i and δI_j.
- A fused image I_{ij} is created by combining backward warped image I'_i and unwarped image I_j.
- The idea of the error calculation suggested by Perazzi et al. consists in trying to find the origin of each patch of the combined image δI_{ij}, by comparing the distance between patch p_{ij} and each of p_i and p_j of the images in the reference projection keeping the smallest distance.
- The total will be the average of error at each pixel location.

This process does not need to be done on the color images but instead the authors suggest using the image gradients since they are interested in structural errors and do not care about image intensities which can be handled in the blending step. Therefore, the exact steps will be:

1. The images δI_i, δI_j and δI_{ij} are transformed into gradients G_i, G_j and G_{ij} using Sobel operator.
2. A sliding window patch of 25*25 is used to compare patches from G_{ij} to G_i and G_j.
3. In order to be able to compare patches of G_i to $_{ij}$, the warp applied to $\delta I'_i$ has to be applied to each patch p_{ij}. This is done by calculating a

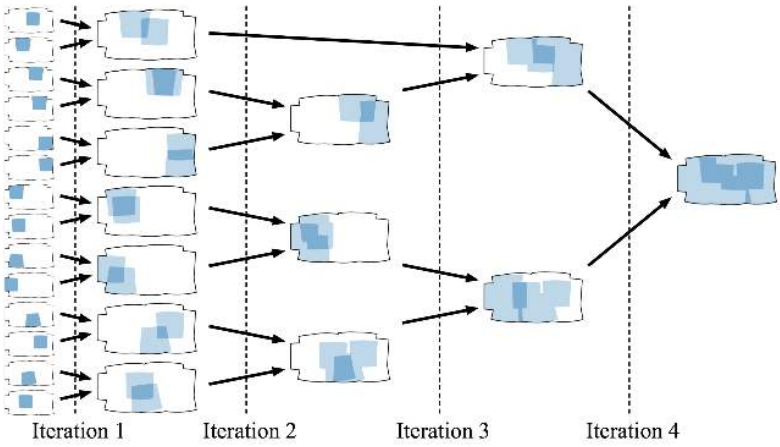


Figure 3.19: Diagram from [Perazzi et al., 2015] showing the iterative process of finding an optimal warp order.

homography for each patch of the image G_i and use this homography H to affine transform this patch. They suggest doing this to avoid depending on the per-pixel motion field which can be erroneous.

4. Distance will thus be calculated as follows: $d = min(|p_{ij} - p_i|^2, |p_{ij} - (H \circ p_j)|^2)$

5. To constitute the error map ϕ , the distance value of the patches including a given pixel $p*(x)$ is accumulated at this pixel location in the error map which is equal in size to image I_{ij}.

6. Finally the global parallax error of this pair is calculated as the average of values in ϕ:
$\Phi_{ij} = \sum_{x \epsilon G_{ij}} \phi(x)$.

Optimal Warp Order

After calculating the parallax error value for each pair, the pairs will be sorted by this error value such that the pairs with the least error will be warped first in an iterative approach. Each iteration will combine as many pairs as possible with the correct order blending them into pairs until no pairs remain. The pairs are now fragments of 2 and probably some single

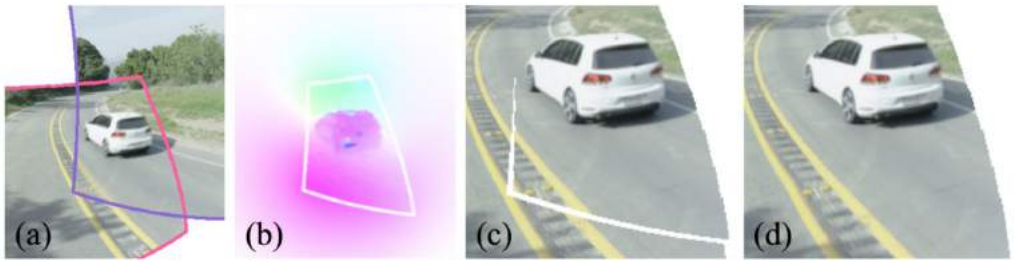


Figure 3.20: Warp extrapolation in [Perazzi et al., 2015]. (a) Two images overlapping with borders highlighted in different colors. (b) Calculated motion field in the region of overlap. (c) Gap resulting from merging the two views after warping one image to another in overlap region only. (d) Result after extrapolation took place.

images will remain if they do not match with anything or if the number of images is odd. The second iteration will do the same but with fragments, until all are combined within this iteration. The loop stops when all the images have become one fragment (see figure 3.19).

Warping for Parallax Compensation

The optimal warp order explained above is calculated only once in the reference projection. This is saved and used for every frame in the video. It is also important to note that optical flow, warping and blending calculated so far are only meant to obtain this optimal warp ordering but the original images remain intact. After an optimal order has been established, images are warped one by one using the accumulated flow field calculated at each step.

Globally Coherent Warping

So far, the warp function has been applied in the overlapping region only with the accumulated motion fields. This can cause a visible seam and can even cause a gap between the region of an overlap in an image and the rest of the image (see figure 3.20).

To overcome this, the authors propose to extrapolate the motion field u_{ij}

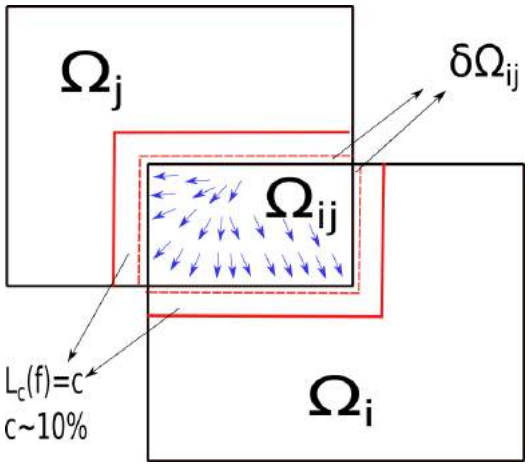


Figure 3.21: Motion field extrapolation of [Perazzi et al., 2015] explained.

by minimizing the Poisson equation with Dirichlet conditions shown below:

$$E(\tilde{u}_{ij}) = \int_{\bar{\Omega}_{ij}} |\nabla \tilde{u}_{ij}| dx \tag{3.1}$$

where $\bar{\Omega}_{ij} = \Omega_j - \Omega_{ij}$ is the non-overlapping region for image I_j and $\tilde{u}_{ij}$ is the unknown motion field to be extrapolated in the non-overlapping region of the image. Equation 3.1 is solved with the corresponding Euler-Lagrange equation $\Delta \tilde{u}_{ij} = 0$ with dirichlet conditions $\tilde{u}_{ij} = u_{ij}$ along the boundary $\delta\Omega_{ij}$. In addition, they add another set of boundary conditions to increase smoothness by setting $\tilde{u}_{ij} = 0$ along $L_c f = c$, where c is equal to 10% of the panorama resolution and L_c is a distance transform from the closest pixel in the region of overlap Ω_{ij}. Figure 3.21 clarifies the notations further.

A last step is done which involves aligning all video frames to the reference frame to avoid temporal instability. This is done using a global relaxation which imposes a high alignment weights for the pixels in the overlap regions through the video frames and gives less weight to the rest of the pixels which can move more freely. This is because the pixels in the overlap region have been subject to more processing and are more affected by parallax. Therefore, a relaxed map v_s is calculated as follows:

$$E(v_s) = \int_{\Omega} w(x)|v_s - v|^2 + |\nabla v_s|^2 dx \tag{3.2}$$

where v is a function mapping a pixel in a given frame to the corresponding pixel in the reference projection. In the case of multiple original pixels (in

the overlapping region), an average of all contributing pixels is used.
The energy function in equation 3.2 is solved using the corresponding Euler-Lagrange equation $wv_s - \lambda \Delta v_s$.

3.4 Performance Evaluation

The previous sections in this chapter has put the light on how challenging video stitching can be. Video stitching is a very active research area at the moment and optimal solutions are not reached. This inspired us to work on methods that can assess the quality of panoramic videos. We believe this can be very helpful in order to design better solutions and compare videos generated with different algorithms. The next chapters are dedicated to the proposed quality metrics and the experiments and results obtained during this PhD.

3.5 Summary

We presented an overview of the difficulties in stitching panoramic videos. These included mainly calibration methods for multiple cameras, parallax errors and temporal instability. We discussed state of the art methods that strived to minimize these issues. We also addressed the topic of panoramic video content for VR including panoramic stereo videos and their added challenges. We showed a number of stereo capture systems and their associated video stitching implementations. We finally provided a detailed explanation of Disney's solution to video stitching [Perazzi et al., 2015] which was implemented as a baseline for this thesis. We ended by a motivation for the thesis main contribution in quality assessment for panoramic videos.

Chapter 4

Objective Quality Assessment for Panoramic Videos

4.1 Overview of Quality Metrics

Digital videos are one of the most essential multimedia tools. They can suffer however from many artifacts due to compression, transmission, streaming issues and more. Synthesized videos are also becoming popular and they have more challenging problems to solve. Assessing those videos is important for video production for various reasons. First, it helps comparing different video processing algorithms as well as the possibility to optimize them by minimizing the error. Then, it can be beneficial to understand and analyze video contents and the source of distortions.

Since human observers are the end users for those videos, we would ideally like to have them rate the quality of those videos. Unfortunately, it is quite expensive and unpractical to assess every video-based application by human participants. Therefore, designing automatic quality metrics that mimic the human perception is the alternative solution.

Employing traditional quality metrics designed for 2D images and videos is not well suited to capture the geometric nature of panoramic video distortions. A performance evaluation has been conducted by Zhang et al.2017 [Zhang et al., 2017] on a number of objective quality metrics for assessing omnidirectional visual content. The evaluated algorithms included 4 traditional methods and 3 others designed particularly for muti-view content, all of which are PSNR-based. Using subjective experiments they were able to provide a com-

parison between human-based assessment and automated assessment. They concluded that the tested algorithms designed for omnidirectional scenes do not outperform traditional methods, which motivates research to obtain better quality metrics for panoramic videos.

We start by an introduction and a categorization of quality metrics supported by Vranjes et al. survey in 2013 [Vranješ et al., 2013].

Objective vs. subjective metrics Subjective quality assessment consists in human participants giving their own opinion of the quality of a media, here a video. Although expensive, it is still essential to validate any automated metric and to gather statistics and facts human perception. Objective quality assessment on the other hand involves an algorithm for calculating a distortion metric that gives an indication of the overall quality of a video in our case.

Full reference, reduced reference or no-reference metrics Objective quality metrics can fall into one of 3 categories. The first one called full reference involves the comparison of two videos, one is considered the original and is used as a reference to compare with the distorted one. The reduced reference extracts features from the original videos that are taken into account in the calculation of distortion of the processed video. The last category is the no-reference metric which only calculates the error on the processed video.

Data metrics vs. picture metrics Data metrics [Vranješ et al., 2013] is a category of methods where the comparison between reference and processed images is done directly on the data such as in mean square error (MSE) or peak signal-to-noise ratio (PSNR). Whereas, picture metrics obtain information about the video content and distortion types such as modeling the human vision perception.

4.2 Motion Estimation for Video Quality Assessment

Motion estimation is an important feature when considering videos. Motion-based objective quality metrics for videos have shown to be successful [Seshadrinathan et al., 2010]. An early research in 1993 by Webster et al. [Web-

ster et al., 1993] has proposed a spatio-temporal approach for videos where the spatial feature is calculated using a Sobel operator for edge detection whereas the temporal feature is extracted from frame difference between t and $t+1$. Standard deviation is used to detect a motion in the degraded video that highly deviates from the reference video.

A no-reference metric for omnidirectional videoconferencing was presented by Leorin et al. in 2005 [Leorin et al., 2005] and depends on extracting spatial and temporal features to signal distortions. The temporal feature is extracted from motion tracking based on the correlation between motion quantification and the perceived quality. An additional analysis is done on the seam edges between images where artifacts are likely to occur.

Although years apart, a similar idea has been adopted by k. et al. 2016 [K. and Channappayya, 2016]. The method calculates a temporal feature based on optical flow and a spatial feature based on MS-SSIM. Finally, a pooling method is applied to obtain a single score for the whole video. The temporal feature is calculated by first computing the optical flow on a frame-by-frame basis. Statistics are then extracted per patch and deviation of values between the reference and distorted videos is treated as an indication of a distortion. The method is similar to ours in the idea of using statistics from optical flow such as standard deviation used in both works whereas it is intended for single-view videos and thus does not have to deal with multiple inputs.

4.3 Quality Metrics for 3D Synthesized Views

3D novel-view images and videos are generated using depth image-based rendering (DIBR) approaches. This process is carried out through the interpolation of multiple views in the purpose of generating a novel view. We address this problem given its similarity with panoramic videos, which also generates a single video out of multiple videos after undergoing a series of geometric transformations.

Bosc et al. 2011 [Bosc et al., 2011] conducted a set of experiments on objective and subjective quality metrics for novel-view synthesis and proved that traditional quality metrics fail to capture distortions for this type of images/videos. They therefore called for future research to establish tailored quality metrics for 3D synthesized view assessment.

We investigated a number of related work that address quality metrics

for this type of problem. Conze et al. [Conze et al., 2012] propose a full reference quality metric which they call "VSQA: View Synthesis Quality Assessment" to capture geometric deformations that occur due to 3D warping in synthesized views. They base their work on masking effects that happen in the human visual system when looking at an image. Three visibility maps are calculated based on 3 features, contrast, texture and diversity of gradient orientations. These three maps are used as a weights for traditional quality metrics such as structural similarity index metric (SSIM) and peak signal-to-noise ratio (PSNR). Their experiments are focused on the SSIM which captures structural similarity between a reference and a distorted image. They use one of the two views to be interpolated as the reference and the other one warped towards it as the distorted. Their results seem to outperform the use of simple SSIM and we use their visibility maps in our experiments presented in the coming sections of this chapter.

In 2015, Battisti et al. [Battisti et al., 2015] propose another full reference metric called "3DSwIM: 3D Synthesized-view Image Metric". The method performs a block-based comparison between the reference and the synthesized images and calculates the metric using Haar wavelet transform to detect the statistical variations of both images. A skin detector is used beforehand in order to increase the weight of distortions present on humans, since it has been noticed that artifacts are more disturbing when perceived on humans. The metric is calculated on datasets generated by 7 DIBR algorithms and compared with other quality metrics. Results shows the metric performs better or worse depending on the algorithm used.

So far, the methods described are only concerned with synthesized images. A quality metric for synthesized videos were presented by Liu et al. in 2015 [Liu et al., 2015]. A full reference objective quality metric is proposed which focus on a particular type of artifact, which is temporal flickering. Temporal flicker is shown to be the most annoying type of error when it comes to synthesized videos from multiple views. Flickering in a video will be represented as a fluctuation in a pixel intensity between consecutive frames. Therefore, they define it using what they call the temporal gradient which will signal a flicker in case of a high change of the gradient magnitude between two frames at a given pixel. They also use a spatio-temporal structure assessment to detect flicker on foreground or background with camera motion.

All the previous studies validate and compare their works using subjective studies.

4.4 Quality Metrics for Panoramic Videos

Most published techniques for panoramic video assessment rely on user experiments where observers wear a HMD (head-mounted display) and salience data is recorded and analyzed [Xu et al., 2017, Zhang et al., 2017].

However, there still exist very few objective metrics proposed. For instance, Cheung et al. 2017 [Cheung et al., 2017] presented an interesting research on quality metric for a stitched panoramic image which uses optical flow calculation in overlapping regions between views along with a salience map that guides the calculation. This method is well suited for stitched images but does not include a temporal extension in the case of video. It also relies on the use of a central image as the reference which constraints the setup of the cameras. This type of metric however is very useful as a guidance to refining stitching algorithms, such as [Li et al., 2018] who proposed a human perception based seam-cut stitching algorithms.

4.5 Proposed Spatial Quality Assessment

In this section, we present our proposed solution for a spatial quality evaluation of a panoramic video frame prior to actual blending.

4.5.1 Suggested workflow

We choose to do our error calculation prior to blending for three main reasons: first, although blending strives to remove some artifacts, it is a blind method that can introduce new artifacts by removing parts of objects or mistakenly erasing something that is not actually an error. Second, once images have been blended into the final panorama, it is very difficult to recover the original images, which are as the name of the method implies, blended and mixed together in the overlapping areas, therefore post-processing to correct defects will also be difficult. Finally, to detect misalignment and discontinuities, it is essential to compare the structural dissimilarities between intersecting views, which is only available prior to blending. Thereupon, given a number of input views, we go through the stitching steps explained in chapter 2 without proceeding to the final step of blending. We examine the differences between pairs of views in two cases as shown in figure 4.1

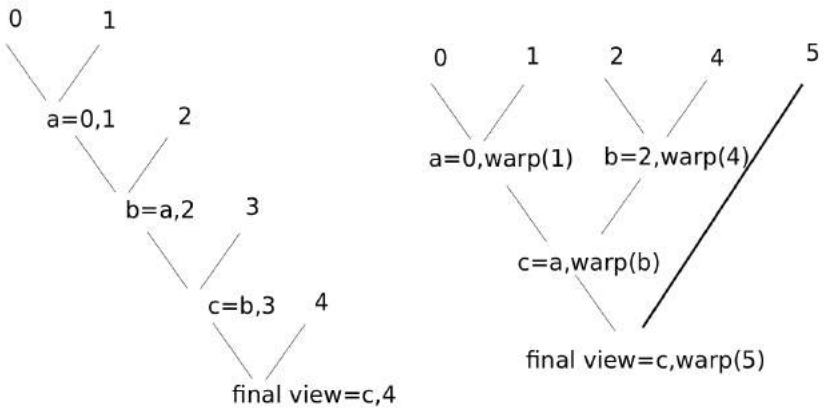


Figure 4.1: Different blending orders used in the current work. Left is progressive blending in the order of image appearance used in Hugin [PanoTools developers and contributors,]. Right is optimal warp order proposed by the authors of [Perazzi et al., 2015] who apply a parallax compensation.

1. Non-warped views in the overlapping regions in the order in which they appear in blending.

2. One unchanged view and the other warped towards it in the overlapping regions in an optimal order calculated as suggested in [Perazzi et al., 2015].

4.5.2 View-Synthesis Quality Assessment: VSQA

The core of our method is based on the VSQA quality metric [Conze et al., 2012], which was designed for DIBR/novel view synthesis, with a new goal, which is error prediction and identification in panoramas. Figure 4.2 explains the pipeline in a simplified manner. The VSQA metric is defined as follows:

$$VSQA(i,j) = \text{dist}(i,j).[W_t]^{\delta}.[W_o]^{\epsilon}.[W_c]^{\zeta}. \tag{4.1}$$

where *dist* is the chosen metric, in this case SSIM [Wang et al., 2004], calculated between a reference view and a synthesized view. This metric is weighted by 3 maps, each representing a type of local feature to which the human eye is most sensitive. Below is a list of these terms (please refer to the original paper [Conze et al., 2012] for more details):

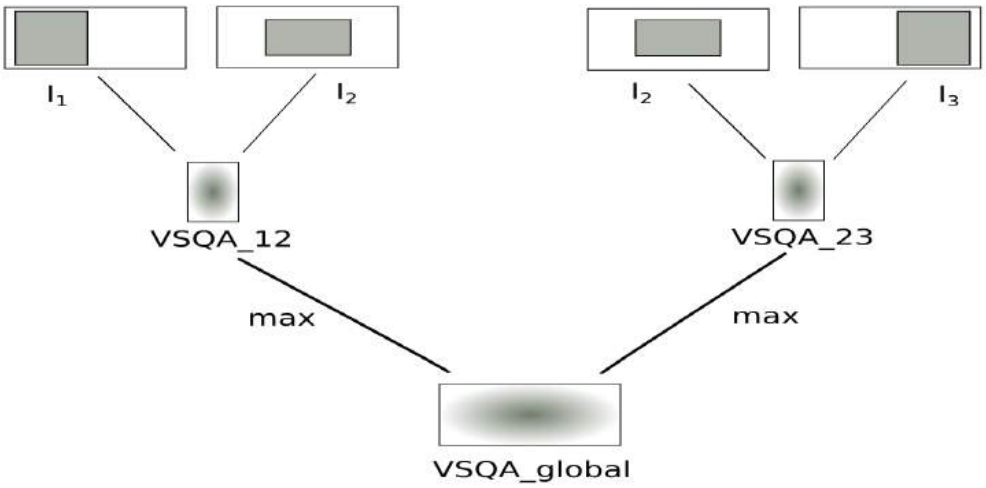


Figure 4.2: A simplified figure for the construction of our error detection on a panoramic video frame.

The texture-based visibility weighting map W_t which compares the gradient of a pixel with respect to its neighbors.

$$V_t(i,j) = \frac{1}{N^2} \sum_{l=i-\lfloor \frac{N}{2} \rfloor}^{i+\lfloor \frac{N}{2} \rfloor} \sum_{k=j-\lfloor \frac{N}{2} \rfloor}^{j+\lfloor \frac{N}{2} \rfloor} w_{l,k} \text{grad}[l,k], \tag{4.2}$$

$$W_t(i,j) = \frac{2V_t(i,j) - \min V_t}{\max V_t - \min V_t}. \tag{4.3}$$

The orientation-based visibility weighting map W_o which calculates the diversity of the gradient orientation of a pixel with respect to its neighbors.

$$\begin{aligned} V_o(i,j) = & min_q[\frac{1}{N^2} \sum_{l=i-\lfloor \frac{N}{2} \rfloor}^{i+\lfloor \frac{N}{2} \rfloor} \sum_{k=j-\lfloor \frac{N}{2} \rfloor}^{j+\lfloor \frac{N}{2} \rfloor} \\ & w_{l,k} \min[(\theta(l,k) - \theta_q)^2, (\theta(l,k) + \pi - \theta_q)^2]], \end{aligned} \tag{4.4}$$

$$W_o(i,j) = \frac{2V_o(i,j) - \min V_o}{\max V_o - \min V_o}. \tag{4.5}$$

The contrast-based visibility weighting map W_c which evaluates the contrast of a pixel with respect to its neighbors.

$$V_c(i,j) = \frac{1}{N^2} \sum_{l=i-\lfloor \frac{N}{2} \rfloor}^{i+\lfloor \frac{N}{2} \rfloor} \sum_{k=j-\lfloor \frac{N}{2} \rfloor}^{j+\lfloor \frac{N}{2} \rfloor} w_{l,k} |\mathrm{Lum}(l,k) - Lum(i,j)|, \tag{4.6}$$

$$W_c(i,j) = \frac{2V_c(i,j) - \max V_c}{\min V_c - \max V_c}. \tag{4.7}$$

In all of the 3 equations, N is the window size and $w_{l,k}$ is a Gaussian weight.

4.5.3 Global Map Creation

The VSQA map explained above is a similarity metric between two images, where one is the reference and the other synthesized or processed. In our case, we do not have a single original image and a processed output, however we have N input views and one final output, so we build our error map, by comparing each pair of images in the same order of their blending tree and creating one final composite map 4.2. Consider N views at a time t, after calculating pairwise matches $P_n(i,j)$, for each pair I_i and I_j, we calculate the region of overlap $I_i \cap I_j$ and we compute VSQA metric between the region of interest in each view δI_i and δI_j.

We finally calculate the equation 4.8 to generate a global map for the whole panorama:

$$VSQA_{global}(i,j) = max_{i,j} VSQA_{i,j}(\delta I_i, \delta I_j). \tag{4.8}$$

Where i,j represent pixel location.

We test another case where we choose one view to be warped towards the other and in that case the unchanged view is considered the reference. This will change 4.8 to:

$$VSQA_{global}(i,j) = max_{i,j} VSQA_{i,j}(\delta I_i, warp(\delta I_j)). \tag{4.9}$$

A Normalization is performed on the output map globally.

4.5.4 Blend Mask Visibility Map

The steps described above provide a global prediction of all possible areas where parallax errors can occur by comparing pairs of overlapping regions and identifying structural differences weighted by masks that enforce distortions in areas that are more salient with respect to human perception. However, as mentioned earlier, the blending step aims mainly to remove as many of these errors as possible, though it does not succeed in all the cases. The multi-scale blend described in [Burt and Adelson, 1983] usually uses a Voronoi mask that chooses the blending line to be irregular and therefore more difficult to notice a line between boundaries. But still there will be more probability to see errors around this line where one can imagine it as a pathway between both images, so we assume that the closer the pixels are to that boundary line, the more visible it is. Based on this assumption, we propose to create a weighting mask around this blending edge, which will give more weight to the pixels that fall onto this line and decreases gradually the more we go farther away.

Within the same iterations over pair-wise matches as described in the previous sub-section, for a pair of views I_i and I_j, we calculate the Voronoi seam cut which produces a mask for each view M_i and M_j that determine the cutting line between both views. We are also interested only in the region of intersection between the two images, so we use the sub-masks δM_i and δM_j. In order to create the desired mask which gives weight to the errors on the blending cut, we calculate a distance transform from that line for each of the latter sub-masks, we then calculate a common mask that will be applied to the resulting VSQA as the OR between δM_i and δM_j and we get a mask M_{blend} that we normalize between 0 and 1 as shown in 4.3. We multiply this mask to our VSQA computed at each step in order to enforce errors at the region where the transition between images takes place and attenuate errors farther away from this boundary as described in equation 4.10. We call this measure MVSQA.

$$MVSQA = M_{blend}.VSQA. \tag{4.10}$$

We generate the global MVSQA with the same process used to calculate the composite VSQA as described previously.

Figure 4.3: Example of the suggested mask created around the boundary of the blending line

4.6 Proposed Temporal Quality Assessment

The previous section explained our proposal for a spatial metric that assesses the quality of a single video frame. However, this is not enough for the assessment of a video, especially in panoramic videos since there might appear large artifacts when moving from one frame to another. Therefore, we propose another quality measurement using motion estimation [Nabil et al., 2018].

4.6.1 Suggested workflow

Unlike the spatial metric which was calculated prior to blending, here we use our final panoramic image as our processed image and each of the original input views as a reference. The core of this temporal metric is optical flow estimation. We calculate optical flow at time t with that of time $t+1$ for each of the input views. We do the same for the corresponding output panoramas. We then compare the difference of the end point of each of the views with that of the output panoramic frame. Figure 4.4 explains the approach in a simplified manner.

4.6.2 Quality assessment calculation using motion estimation

In a panoramic video, the final output at time t is a composite novel view from a number of input views that go through a number of geometric transformations from projection to a common surface up to the final blending stage.

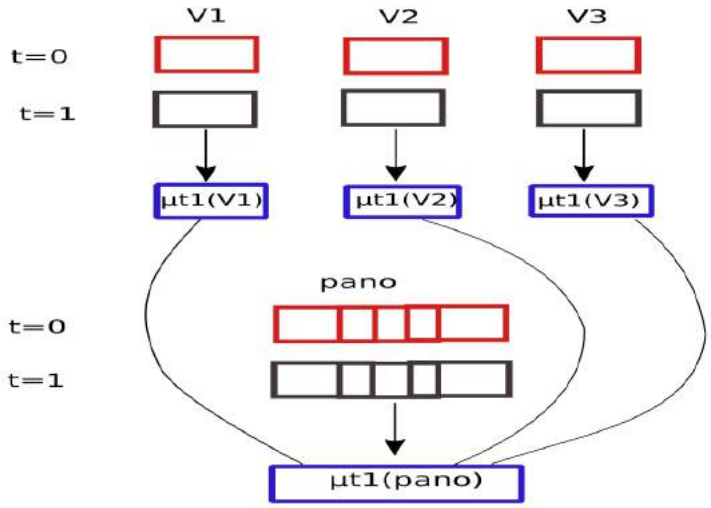


Figure 4.4: Workflow of the temporal assessment calculation.

Panoramic video stitching methods usually add additional transformations for parallax compensation [Perazzi et al., 2015, Lee et al., 2016].

Our method suggests using the original videos as a reference to the single output panorama. The idea is to compare the difference in motion between two given frames in the original videos and motion in the final panorama. However, this is not directly applicable since a given pixel x_{pano} in the final panorama can have one to N sources corresponding to N input videos with overlapping regions. To overcome this, we suggest to calculate the deviation of displacements of all source pixels $x_i, i \in N$ from the displacement of the panorama x_{pano} between times t and $t+1$. Thereupon, we calculate the optical flow between two frames at times t and $t+1$ of each input video and the final panorama. We calculate the standard deviation at each pixel x of the whole image to produce our distortion map M_d as follows:

$$M_d(x_t) = \sqrt{\frac{\sum_i^n (x_{i,t+1} - x_{pano,t+1})^2}{n}} \tag{4.11}$$

where $n \in [1, N]$ is the number of overlapping images at pixel x and $x_{t+1} = x_t + \mu(x_t)$ and μ is the motion field of a pixel between times t and $t+1$.

To obtain more accurate results for error identification, it is important to include human visual system notion of salience. Thus, we used three visibility maps suggested by [Conze et al., 2012] for novel view synthesis to extract a

salience map from our panoramic view. The weight maps represent a model of three features that are assumed to mask errors, which are contrast, texture and variation of texture orientation.

Given a panoramic frame I_{pano}, a salience map M_s is defined as:

$$M_s = W_c(I_{pano}).W_t(I_{pano}).W_o(I_{pano}) \tag{4.12}$$

where W_c, W_t and W_o correspond to contrast, texture and orientation weighting maps which are calculated according to equations 4.7, 4.3 and 4.5 respectively.

Building on the assumption that a human would gaze a region if it is distorted or if it salient, we propose to produce a distortion-salience map M_{ds} from M_d and M_s using a simple weighted sum. The parameter ω can be changed depending on the video content.

$$M_{ds} = \omega M_d + (1-\omega)M_s \tag{4.13}$$

In order to validate our objective quality metric, we conducted a study to assess humans perception to errors in panoramic videos. Details of the experiment are provided in the next section.

4.7 Summary

Objective quality metrics are an important tool for video quality monitoring. They also help in designing better algorithms for video processing. This chapter covered the main contributions carried out during this thesis. First, related work were presented, especially optical flow-based quality metrics and metrics for depth image-based rendering. Details of the methods proposed for panoramic videos were covered. The first method worked by comparing pairs of overlapping images and creating a global map that represents the distortion of the frame weighted by saliency features. Another assessment approach was suggested to improve the previous one by incorporating temporal features using optical flow. The comparison made was between the individual input videos as references and the final stitched video. A final map was created by combining the optical flow-based distortion map with a saliency map.

Chapter 5

Human-centered Evaluation for the Proposed Objective Quality Assessment

In the previous chapter, objective metrics were proposed to assess panoramic videos based on human perception. In order to validate these methods, a human-centered study was conducted to analyze the human's perception and its sensitivity to errors in panoramic videos.

We start by basics on human vision which help understand the proposed experiment based on eye-tracking. Subsequently, related work in eye-tracking and omnidirectional subjective quality metrics are presented. Finally, the proposed experiment's details are covered from design to analysis.

5.1 The Human Visual System

This section covers the anatomy of the human eye and the visual pathways from a physiological point of view followed by a discussion of eye movements and how they are controlled by the visual system. Visual attention is then introduced with classic experiments that laid foundations of visual attention modelling.

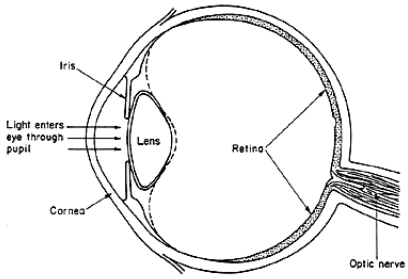


Figure 5.1: Simplified eye anatomy

5.1.1 The human eye

The human eye is a complex organ composed of different parts (see figure 5.1) that work together in order to form an image. The cornea is the curved transparent frontal surface of the eyeball allowing light refraction. The iris is the part containing the melanin[1] and together with the pupil, they are responsible to control the amount of light penetrating through the eye, similar to a camera's aperture. The lens, along with the cornea, refract the light and form a focused inverted image on the retina. Ciliary muscles control the curvature of the lens which in turn changes the focal length to allow the perception of objects at various distances from the eye, a mechanism known as accommodation. The retina itself is composed of several layers. The outermost layer is the layer of photo-receptors: rods and cones. Rods facilitate vision under low light conditions and cones are responsible for colour vision. Their distribution in the retina is not uniform. At the region of fovea[2], there is a high concentration of cones while outside the central fovea the number of cones gradually decrease and the rods start to appear. At the region of optic disc there are no photo receptor cells making it a physiological blind spot. The optic nerve starts here to transmit the signals to the brain which processes and interprets them.

[1] The substance that gives colour to eyes, skin and hair.

[2] A pit area of less than 1 square millimeter in the center of the retina responsible for high visual acuity.

5.1.2 Visual Pathways

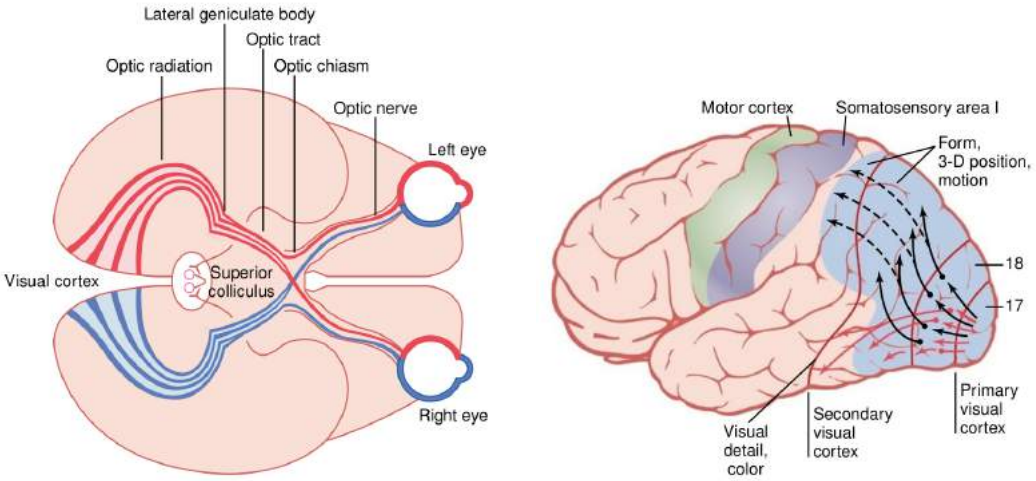
Figure 5.2 (a) shows the schematic representation of the visual pathway from the left and right retinas to the brain's visual cortex via optical nerves. Fibers coming from the retina's rods and cones travel through the optic nerve. The side of the retina closer to the nose, called nasal half of the optic nerve fibers, cross over to the opposite side at the optic chaism. The optic tract, which is a continuation of the optic nerve, has nerves from both eyes and relays information to the lateral geniculate nucleus which transmits the received impulses to the visual cortex via the optic radiation as well as to the superior colliculus. In addition, the lateral geniculate nucleus receives inhibitory control from the visual cortex and this regulates the extent of the visual signal from the optic tract and hence enables it to exert attention. The filtered signals from the lateral geniculate nucleus then pass to the visual cortex by the way of optic radiations.

The visual cortex is divided into a primary visual cortex and the secondary visual areas. The signals first reach the primary cortex responsible for pattern recognition and motion detection. Afterwards, the signals are transmitted through two major pathways to the secondary visual areas which analyze and interpret the visual information received. The black arrows shown in figure 5.2(b) are the first pathway that analyzes the shape and the 3D location of objects and whether they are moving. The second pathway represented by the red arrows is responsible for the analysis of visual detail such as colour details, contrast and texture as well as the interpretation of letters and characters [Hall and Guyton, 2011].

5.1.3 Eye movements

Our central fovea is limited a 2 degrees vision field, equivalent to a thumb nail at a shoulder distance. Foveal vision is what helps us accomplish reading and similar focusing tasks. The eye movements complement this task by enabling us to perceive a larger field of view and be able to grasp a lot of information from the world [Gegenfurtner, 2016].

One of the most important eye movements are those causing fixations. Voluntary movements allow the human eye to move to locate an object of interest while an involuntary movement causes the eyes to fixate. This involuntary fixation mechanism is controlled by the superior colliculus and it causes the eye to lock the image on the fovea.



(a) (b)

Figure 5.2: (a) Schematic visual pathways. (b) Visual cortex in the brain. Figures taken from [Hall and Guyton, 2011].

The superior colliculus is also involved in controlling other eye movements such as saccades and smooth motion pursuit. Saccades are successive fixations caused by a rapid movement of the pupil without the need to move the head. They are important for tasks such as scanning the surroundings or the lines of a book and relevant information is collected. Smooth pursuit is the movement that allows fixating on a moving object and is responsible for motion perception.

When perceiving objects at different distances, the eye uses vergence movements controlled by the superior colliculus to focus on these objects and the accommodation mechanism mentioned earlier allows it to refocus its lens in less than a second while achieving its best visual acuity.

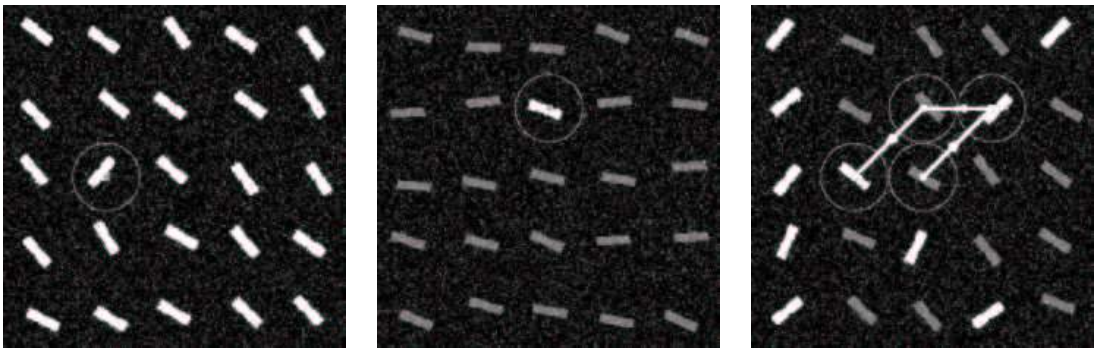
5.2 Visual Attention

Visual attention is a filtering process of the visual system allowing humans to focus on and select a subset of a scene perceived in their field of view [Borji and Itti, 2013]. Visual attention makes it easier for the brain to interpret a scene as it provides a small information and thus simplifying the task of scene understanding. It also provides feedback information that allows interpreting

the different visual attributes of a perceived object [Itti, 2003].

Visual attention can be bottom-up or top-down. The bottom-up attention, also known as image-based, is represented by involuntary eye movements driven by the stimulus. This pre-attentive perception can be demonstrated using techniques such as the visual search experiments published by Treisman and Gelade [Treisman and Gelade, 1980]. These experiments involved two sets of tasks, the first involves presenting to the observer a uniform array of stimuli where one element pops-out by being different in color or orientation whereas the second involves a more distracting array of stimuli that are less uniform and thus the target is more difficultly identified and requires further search(see figure 5.3). Results indicate that in the first experiment, a global scan of the scene involving pre-attentive visual processing is used by participants whereas in the second task, selective attention is used after the scene is closely observed until the observer could identify the target stimulus. As a consequence, the visual system is able to collect simple information from the scene before the attention can bind them together to interpret more complex structures. The experiments conducted by Treisman and Gelade [Treisman and Gelade, 1980] were at the base of many computational models afterwards.

Top-down visual attention is task-dependent which means they depend on the assigned task and not only the stimulus. The experiments conducted to measure this category of attention usually involves multiple salient stimuli within one scene. Results of these experiments show that humans can completely miss an important event happening within their field of view if they attend to another [Itti, 2003]. Using eye-tracking to register participants' eye movements in the classic Yarbus experiments [Yarbus, 1967], it was possible to understand the influence of a given task on the attention of observer. His experiments involved presenting a scene to observers and recording the scan-paths of their eyes while being given different tasks each time such as identifying the age of a person in the image, describing his/her clothes from memory or remembering other details in the image. Figure 5.4 shows an example of one of the images shown and the eye-tracking paths obtained by the same subject but with different tasks assigned. It can be observed from the registered scan-paths that the eye movements and eventually the visual attention greatly differ according to the requested task.



(a) (b) (c)

Figure 5.3: Experiments pioneered by Treisman et al. showing two set of visual search tasks. Figures (a) and (b) are examples of orientation and color pop-out respectively, both of which were easily identified by the observer, whereas figure (c) involves a conjunctive search where the target stimulus being different in more than one feature than others, in this example it is the only bright element with different orientation. The task shown in figure (c) was more challenging and resulted in multiple false positives prior to making the correct choice.

5.3 Subjective Quality Metrics

Subjective quality metrics for image and video are metrics obtained using a user study involving human participants who are asked to evaluate the quality of an image or a video. These metrics are considered the most reliable tool for assessing image and video quality since the end-user is most likely a human.

A survey has been conducted on objective and subjective quality assessment by Mohammadi et al. [Mohammadi et al., 2014] which categorizes the standard methods for conducting subjective quality metrics as follows:

- **Single stimulus categorical rating**, in which test images are displayed randomly for a given time on a screen and observers are asked to provide a score using a categorical scale from bad to excellent or equivalent.
- **Double stimulus categorical rating**, is the same as the previous method but with reference images and test images being displayed simultaneously.

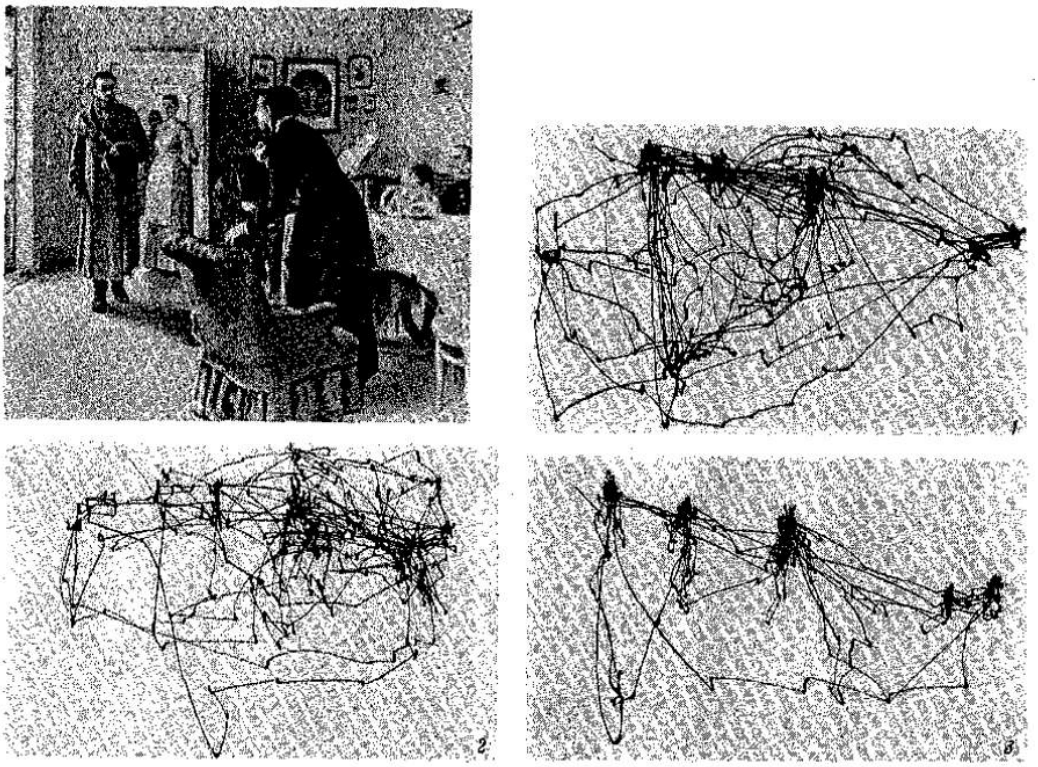


Figure 5.4: Results of eye-tracking from one participant at the Yarbus experiments 1967 [Yarbus, 1967]. In the first recording, the subject was left to observe the scene freely, whereas in the second, he/she was given instructions to give an estimation of the financial level of the family and in the third, the observer was required to estimate the age of the persons in the shown image. It is clear how the scan path of the eyes was different depending on the instructions given.

- **Ordering by force-choice pair-wise comparison** involves displaying two images where the participant is to choose one of them based on the highest quality. The task has no time limitation.
- **Pair-wise similarity judgments**, similar to the previous method but with an added task where the observer has to indicate the level of difference in quality between both images using a continuous scale.
- **Difference mean opinion score (DMOS)**, a score used by modern

quality assessment images using the following equation to obtain the difference between the reference and test images.

$$d_{p,I} = r_{p,I_{ref}} - r_{p,I_{test}} \tag{5.1}$$

where $r_{p,I_{ref}}$ corresponds to the score of participant p to reference image I_{ref} and $r_{p,I_{test}}$ is the score of participant p to test image I_{test}.
This metric is also used to compare the scores obtained from subjective quality assessment experiments with indices calculated by objective quality metrics.

- **Z-score**, is a score that normalizes the scores obtained by each observer to the quality of images. It is calculated according to the following equation:

$$z_{p,I} = \frac{d_{p,I} - \bar{d}_p}{\sigma_p} \tag{5.2}$$

where $\bar{d}_p$ and σ_p are the mean and variance of observer p for all images.

5.3.1 Eye Tracking for quality assessment

Given the importance of eye movements in human perception, eye-tracking has been widely used to collect information about human gaze in several applications [Krafka et al., 2016]. We focus here on experiments aiming to study the effect of visual attention on designing quality metrics.

Ninassi et al. [Ninassi et al., 2007] conducted an eye-tracking experiment examined the effect of where the human eye looks within an image on quality perception. Twenty participants were asked to observe a reference image and an impaired image degraded by compression for 8 seconds and rate each on a 5-scale between imperceptible and very annoying. Afterwards, two saliency maps were obtained using the fixation number and using fixation duration. Maps are then merged and smoothed to obtain a final saliency map. To assess its effect on objective metrics, distortion maps were calculated using two methods, simple absolute difference and structural similarity index(SSIM). A spatial weighting approach was applied to the distortion map using the human salience maps.

A similar study was published where two eye-tracking experiments were conducted by Liu and Heynderick [Liu and Heynderickx, 2011]. They also aimed to have a ground-truth of visual attention via eye-tracking. In one

experiment they asked 20 observers to look freely at 29 images while their gaze data is being captured. In the other, a different group of 20 participants was asked to score the quality of the images they saw on 5-scale from bad to excellent. With the gaze data obtained from eye-tracking, they constructed a salience map of the fixations. To assess the added value of the salience data, they calculated the mean opinion score with a number of well-known objective metrics including PSNR and SSIM. Afterwards, they weight the results of the objective metrics with the salience map obtained from each experiment.

Both studies [Ninassi et al., 2007, Liu and Heynderickx, 2011] show the importance of combining salience maps and modelling human visual attention with objective quality metrics for image or video.

5.3.2 Subjective Quality Evaluation for Omnidirectional Content

The quality of experience in virtual environments is a very important aspect that should not be neglected to avoid visual discomfort and fatigue. A number of experiments [Upenik et al., 2017, Rai et al., 2017] have been published for quality of experience in the case of omnidirectional images. While it is related to our work, we are concerned with quality assessment for panoramic videos which is more challenging. In the following, we review two methods for quality assessment in viewing panoramic videos.

Xu et al. [Xu et al., 2017] conducted a user experiment that included 40 participants. Each one was asked to put a *HTC Vive* headset and be seated on a swivel chair to allow the observer to turn freely. Head tracking data were collected to represent salience. A rating interface was displayed which allowed users to score a video without removing their virtual reality headset. The viewing direction data showed a preference of users to gaze at the center of the scene even though it might depend on the nature of the video. They also proposed variations on the existing subjective metric DMOS explained earlier, which reflect global and local quality scores for each video. They also proposed objective quality metrics which were validated by a comparison by the data obtained from the user experiment.

A study conducted by Schatz et al. [Schatz et al., 2017] focused on exploring the effect of stalling[3] when viewing an omnidirectional video using a

[3] A term referring to the the event of a video freezing in the middle of being played.

head-mounted display or a TV. Twenty-seven participants watched impaired videos and rated the overall quality in a range of 1 to 5 after each video. Furthermore, a questionnaire was filled by participants after viewing each video. The authors concluded that stalling can gravely affect the quality of experience when watching panoramic videos.

The main difference of the methods described in this section with respect to the method proposed in this thesis is the type of stimuli. While these methods tested immersive 360° videos within a virtual reality environment, our method used wide-angle panoramic videos as stimuli and aimed at studying the type of errors that are most perturbing rather than the global quality of experience. A detailed description of our method is presented next.

5.4 Proposed Approach

The objective visual quality metric presented in the previous chapter used a distortion map along with a salience map to model of the human visual system sensitivity to errors in panoramic videos. To determine its accuracy with respect to human perception, it was essential to conduct an empirical human-centered study. The main purpose of designing this experiment was to compare the objective metric with the subjective data provided by human participants. Our main experimental question was: Do humans recognize errors that are not identified by an algorithm and vice-versa?

5.4.1 Designing the experiment

To establish a protocol to the experiment, we used *Tracable Human Experiment Design Research (THEDRE)* [Mandran, 2017]. The planning step of the experiment resulted in the conclusion of studying decision-making by participants with respect to error perception through error annotations and to study their visual attention and what's most salient for them in a panoramic video setup through eye-tracking.

A user annotation interface

While the regular standardized methods discussed in section 2 are most commonly used, we observed that simply giving a score was not sufficient to achieve the goal of this experiment, instead a more precise comparison of the

Figure 5.5: Tobii Pro glasses 2.

errors identified by humans versus the algorithm was needed. Consequently, we decided using an interface of annotation within which the user can play a video, pause whenever they perceive an error and annotate the error. We designed a simple application to play the video and allow to draw circles in the regions where an error is perceived. Once they continue to view the video, the annotated is saved with the frame and the participant numbers. Annotations and labelling is widely used to create datasets corresponding to ground truth [Torralba et al., 2010] that can be used later for training a neural network for object recognition instance. Similarly, we needed to have a ground truth of errors recognition, thus we chose annotations for our experiment.

Eye tracking

In order to study visual attention in panoramic videos, we used eye-tracking to record gaze data of participants during the experiments. Eye-tracking was necessary to give us another perspective that is less subjective about the participants reactions to the video viewing. For this, we used *Tobii pro glasses 2* which is a light pair of glasses that can be worn and that lets the user watch freely. As seen in figure 5.5, the eye-tracking Tobii glasses have two side infrared cameras to capture the gaze from each eye. It also has a microphone to record the voice if needed, exchangeable nose pad to accommodate the participant and a removable protective lens depending on the environment in which it is used. Finally, a full HD wide angle camera is used to capture the gazed scene itself. This gives a reliable high quality gaze data.

Figure 5.6: Experimental setup: a participant sitting in front of a large screen, wearing Tobii eye tracking glasses and annotating errors with keyboard and mouse.

5.4.2 Protocol description

Our methodology involved two approaches: the study of visual attention on the objects of the panoramic scene and the study of the annotations, made by the participants to mark an error. We define an error as a region with high number of fixations and an annotation by at least one participant. The experimental setup consisted of a large screen where the panoramic videos were projected in a random order, linked to a keyboard and a mouse that allowed to perform annotations (see figure 5.6). Gaze data, related to visual attention, were collected with wearable eye-tracking *Tobii Pro Glasses 2* and processed with the gaze analysis software *Tobii Pro Lab*. The experiment took place in *INRIA Rhône Alpes* and *Amiqual4Home* buildings for 3 days, where 26 persons took the experiment with only 20 kept, whose eye-tracking data accuracy was 90% or above. The protocol for this empirical study is detailed below:

Goal of the experiment: There are two main goals to our experiment. The primary goal is to validate the proposed optical flow-based objective metric by comparing the detection of errors in panoramic videos by the automated method versus human identification to errors. The secondary one is to gain further knowledge about the sensitivity of the human visual system to the perception of errors as well as areas of salience when viewing panoramic

videos.

Experimental questions: In addition to the main experimental question, we more specifically wanted to answer the following questions:

- Does the proposed metric capture errors perceived by humans ?
- Are the same errors perceived by all participants?
- Which errors are most salient?

Participants: A total of 27 participants took the experiment, however we only kept the recordings of 20 whose captured gaze data was 90% or above of accuracy. The participants were 7 women and 13 men, with only 5 familiar with image and video stitching.

Duration of the experiment per participant: 20 minutes.

Task: Each participant watches a number of short video sequences (less than 30 seconds) within an interface that allows them to pause and annotate a perceived error. The participant wears eye-tracking glasses while watching for recording gaze information.

Metrics: The measurements we used to draw our statistics are:

- Number of errors annotated per video (from annotation interface).
- Total fixations count [Holmqvist et al., 2011] (from eye-tracker).
- Fixation duration [Holmqvist et al., 2011] (from eye-tracker).

Material: The setup of the experiment involved the following materials:

- A wide screen, a keyboard and a mouse were used for annotations.
- Tobii Glasses 2 Pro, Tobii Controller and Tobii Pro Lab software were used for the capture and analysis of eye data.

Stimulus: Four short panoramic videos suffering from stitching errors(ghosting, misalignment and deformation due to imperfect synchronization and/or parallax) were displayed in random order for each participant on a wide screen.

Filter: We used the Velocity-Threshold Identification (I-VT) Tobii filter for Attention [Salvucci and Goldberg, 2000]. It is a filter used to identify the eye movements using the velocity of eye shifts.

Analysis: To analyze the data, we used Tobii heat maps to obtain areas of high fixations by participants. We used Tobii's Areas of Interests(AOI) to generate statistics that answer our experimental questions. More details are provided in the next section.

Description of the experiment The steps of the conducted experiment which took place over 3 days in *INRIA*, were as follows:

1. We first explain to the participant the experiment's purpose, the description of the problem and what tasks are required.

2. Then, we give the participant an example to get familiar with how the annotation interface works and understand the task.

3. Afterwards, we calibrate the eye tracker with the participant's eyes by asking him/her to hold a card at a shoulder's length and to gaze at the black dot in the middle of the card.

4. Once the calibration is done, we start the eye tracking recording and launch the videos for the participant who continues to watch and annotate the errors.

5.4.3 Data analysis

The experiment's setup allowed the collection of two types of data, participants'gaze and frames annotations. Our analysis was conducted on both types of data jointly. For each video, we chose one or more keyframes representing the central view within a sequence of frames. Depending on the scene being taken from a fixed view-point or a moving camera, we chose one keyframe or more. To compare this with our objective metric, we calculated the temporal distortion on a sequence of frames whose center was a given

keyframe. Temporal pooling of a given sequence was done by a simple OR then we overlaid our final composite map on the keyframe.

Data analysis was done using *Tobii Pro Lab* software. For each keyframe, we defined annotations collected by the 20 participants using Tobii Areas of Interest (AOI) corresponding to at least one human annotation. On the same keyframe, we defined the error regions identified by our distortion map. Qualitative results were obtained by generating heat-maps from gaze data recordings using an I-VT Tobii attention filter as suggested by Salvucci and Goldberg [Salvucci and Goldberg, 2000]. Classical metrics such as total fixation count and total fixation duration [Holmqvist et al., 2011] were used to obtain descriptive statistics on the defined AOIs. Examples of AOIs are shown in figures 5.7 and 5.8.

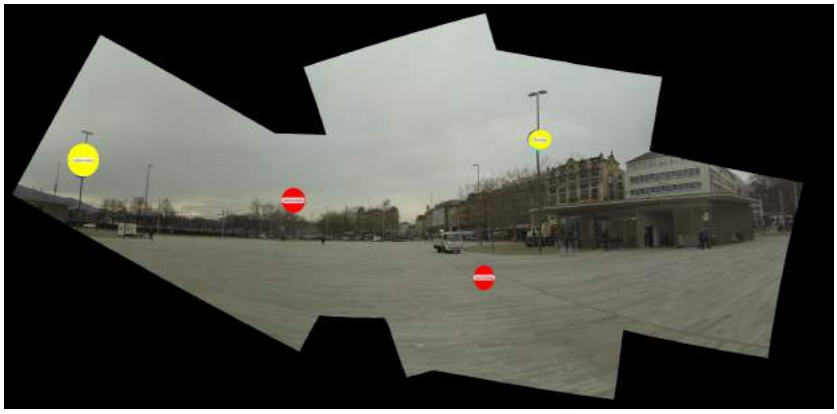


Figure 5.7: Areas of interest defined in regions of human annotations that agree with those detected by the algorithm in yellow and those that were only detected by the algorithm in red

.

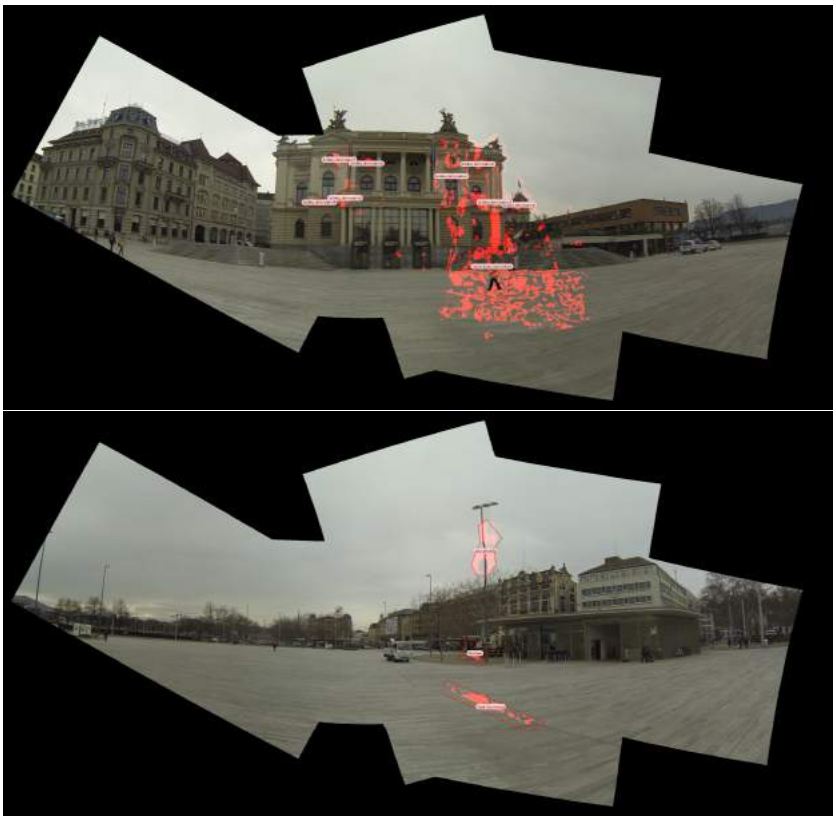


Figure 5.8: Areas of interest with labels describing the type of deformation
.

5.5 Summary

This chapter describes the conducted human-centered study that was used to validate our objective quality metric. It starts with a background on eye-movement and its effect on visual perception from a physiological and psychological point of views. It then discusses related work for subjective quality assessment based on eye tracking and experiments made for omni-directional videos. Finally, details of our experiment are provided from the design phase to the analysis. Results of the experiment are shown in the next chapter in correspondence with results of the proposed objective metrics.

Chapter 6

Experiments and Results

6.1 Creation of a Panoramic Video Dataset

This section describes the datasets used and acquired within this thesis to test the proposed methods. Three different datasets were used, the first one was acquired using a rig within our lab with only 3 cameras. It was used to capture various scenes that were used for initial testing but that suffered from synchronization issues and that were constrained by the setup of the rig and the field of views of the cameras. The second was created and shared by Disney researchers and it had 7 video sequences taken with a variety of rigs ranging from 5 to 14 cameras. Disney's dataset was the one we used the most since it did not suffer from synchronization problems and also helped testing the output of their method [Perazzi et al., 2015] which was an example of a successful stitching method in the state-of-the-art. Finally, we tested using the commercial camera Omni GoPro to take 360° panoramic videos.

6.1.1 A 3-camera rig

To create our first set of panoramic videos, we designed a camera-rig with the help of engineers from Amiqual4Home [Amiqual4Home, 2016].

Figure 6.1: 3-camera rig designed with the help of engineers in Amiqual4Home [Amiqual4Home, 2016]

The rig, shown in figure 6.1, consisted of a metal bar on which we added three accessories to fix the cameras that can be adjusted along the bar and another fixed accessory that allows placing the bar on a tripod. Two wooden handles were designed and cut using the laser cutter at *Atelier numerique Amiqual4Home* that allows carrying the handle in case we would like to shoot panoramic videos while moving. We used a manfrotto tripod and accessories that were available at INRIA. The cameras used were all Lumix Panasonic GH2 cameras with 20mm lens each. They were placed such that they have a sufficient overlap.

We used the rig to record some datasets within *INRIA* and outside. With a lot of trials, we decided to keep two datasets *Babyfoot*, which consisted of an indoor scene inside a babyfoot room in *INRIA* with multiple players and the rig was placed 1 to 2 meters away and *Snow* which was taken from the terrace of *INRIA*'s cafeteria for the view outside while there was falling snow and the rig moves a little to show the whole surroundings. The main drawback of this capture system is the lack of automatic synchronization between the cameras. To resolve this, we used manual synchronization by registering the sound of a hand clap then using the audio file associated with the video to synchronize the videos. However, the resulting video sequences had two main limitations. First, the number of cameras was relatively small, only three and the placement of cameras was horizontal, therefore the vertical field of view was limited. The second shortcoming was the lack of automatic synchronization which was not completely resolved manually, hence causing

temporal incoherence to the videos.

6.1.2 Disney dataset

The second dataset we used was shared by Disney researchers [Perazzi et al., 2015]. The dataset has 7 videos taken with their unstructured camera rigs mostly for faraway scenes with little movement. We mainly worked on 3 of these datasets: *Opera*, which was taken by a 5 camera-rig and the camera moves while shooting a street near the Opera house in Zurich. This dataset contains humans walking, buildings, camera movements and only minor non-synchronization between the cameras. The other two datasets are taken with 14 cameras, one is called *Street* which is a taken with a fixed rig for a street with some cars and pedestrians and having barely noticeable errors after stitching and the other is *Terrace* consisting of the inter area of some buildings taken from a terrace, the camera keeps moving and distortion is only visible towards the end of the video.

6.1.3 Omni GoPro

We have experienced a more advanced and professional capture system using Omni GoPro (shown in figure 6.2 that we borrowed from Kolor GoPro.

Figure 6.2: Taking panoramic videos with Omni GoPro at the Giza pyramids.

The camera-rig is composed of 6 Hero 4 GoPro cameras, placed within a 3D printed cube. The cameras are connected between them and are fully synchronized. To start shooting, you only need to turn on the master camera and the array of cameras will start and will be ready for shooting. The camera was used to capture both indoor and outdoor scenes, it was always necessary to place the camera on a tripod and use it to carry the rig around or to fix it on the ground. Videos recorded with Omni GoPro include 3 videos in Giza pyramids in Cairo, Egypt and four within Grenoble, France (see figures 6.3 and 6.4).

Figure 6.3: Panoramic video frame taken using Omni GoPro for Giza pyramids, Egypt and stitching using Hugin [PanoTools developers and contributors,]

Figure 6.4: Panoramic video frame taken using Omni GoPro for Notre Dame square in Grenoble France and stitching using Hugin [PanoTools developers and contributors,]

6.2 Results of Experiments with the Proposed Quality Metrics

As explained in chapter 4, two quality metrics were proposed within this thesis. Corresponding results of each are shown below along with validation from chapter 5.

6.2.1 Proposed method 1

The first method suggests a spatial approach that aims to predict errors prior to blending by comparing only regions of overlap. Each step creates a distortion map based on the view-synthesis quality assessment (VSQA) suggested by Conze et al. [Conze et al., 2012]. According to the method of stitching assessed, a warp is applied to one image towards the other or not. A global map is created by taking the maximum value at each pixel location. Finally, a blend mask corresponding to a Voronoi seam is used to weight errors. A comparison is done with the basic SSIM that shows that VSQA with the blend weighting mask perform better in precising the error location.

In the first proposed method corresponding to equations 4.8 and 4.9 were tested on two sets of outputs resulting from two stitching algorithms, basic stitching using Hugin open source software [PanoTools developers and contributors,] and video stitching algorithm by [Perazzi et al., 2015].

Figure 6.5: Example of the output from equation 4.9 on dataset from [Perazzi et al., 2015].

Figure 6.6: The output from equation 4.9 filtered by 4.10.

Figure 6.7: Example of the output from equation 4.8 on dataset taken by the 3-camera rig designed in Amiqual4Home [Amiqual4Home, 2016].

Figure 6.8: The output from equation 4.9 filtered by 4.10.

In order to test our method, we used the dataset *Opera*, a video sequence taken by a 5 GoPro camera-rig, provided by Perazzi [Perazzi et al., 2015] for their work on panoramic videos. We apply equation 4.9 to represent stitching methods incorporating parallax compensation. We also took our own panoramic video using a 3-camera rig designed by [Amiqual4Home, 2016] formed of 3 Panasonic GH2 cameras with 20mm lens each. Video frames were generated using the open source software Hugin [PanoTools developers and contributors,] for panorama creation, with graph-cut multi-band blending, for which we used equation 4.8. The results shown in 6.8 show a promising prediction for zones of potential errors not only spatially but across the whole sequence. Repeating the process for some key-frames in the video, can show which errors persist and which appear sporadically. It can also be noticed that the error seems concentrated in the right middle part of the panorama in the dataset *Opera* which contains four out of the five views overlapping. The suggested mask permitted to focus on errors around the blend mask and therefore reducing the number of false positives.

In order to obtain a metric index out of our distortion map, we calculated a score according to [Conze et al., 2012], which consists in counting the

number of remaining erroneous pixels after applying a threshold. We used the same method of spatial pooling to compare our results with basic SSIM. Table 6.1 shows the resulting scores in percent of remaining pixels for VSQA and MVSQA described in equations 4.8 and 4.10 as well as basic SSIM [Wang et al., 2004].

Metric / Score in %	**at t=84**	**at t=105**	**at t=385**
MVSQA	0.38	0.55	0.26
VSQA	0.56	1.04	0.29
SSIM	9.63	10.58	8.45

Table 6.1: Preliminary results of spatial pooling

The measures were applied to 3 chosen frames where we could see clear parallax errors. Figure 6.9 shows examples of parallax errors that appeared after stitching and their corresponding maps VSQA, MVSQA and SSIM. The results show that VSQA calculated in equation 4.9 filters more the errors than those calculated by SSIM. MVSQA calculated using 4.10 which gives more weight on the blending line between two images yields more precision and outperforms SSIM and VSQA and this is because errors on tend to fade the farther away from the blending seam cut.

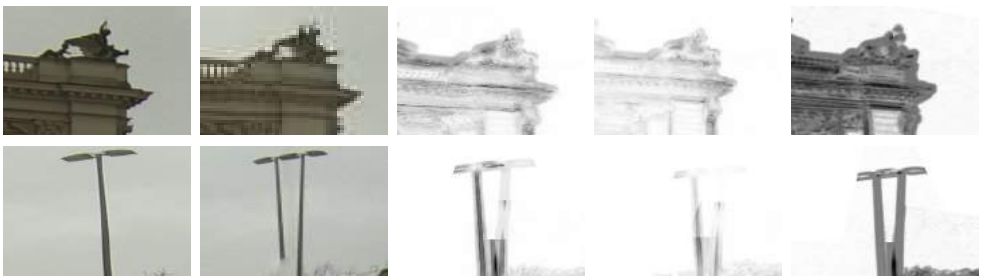


Figure 6.9: Zoom on error in Opera sequence. Top row is panorama at time t=105, the one below is t=385.Then from left to right, the figure shows the original view before stitching,then the image after being stitched. Then the error maps for VSQA, MVSQA and SSIM respectively are shown.

6.2.2 Proposed method 2

Although this method worked globally well, it lacked the temporal aspect as well as a global assessment of the output panorama with respect to the original input views.

Therefore, another method based on optical-flow between two consecutive frames in time was suggested which calculates the deviation of displacements at each pixel location between the input videos and the final video.

Figure 6.10 shows an example of the visualization of optical flow calculated between two frames in times t and $t+1$, with arrows of different colors showing the deviation between flow in the final panorama and those in the input videos. To emphasize the areas of high distortion, only areas of very high standard deviation value are kept. The resulting distortion map calculated using equation 4.11 is shown in figure 6.11.

(a) (b)

Figure 6.10: Example of calculated optical flow with a filter on high deviation between final panorama and input views (standard deviation >50 and >80).

Figure 6.11: Distortion map example on *Opera* dataset.

Afterwards, a salience map is calculated using equation 4.12. The map models three visibility features, texture, contrast and variation in texture orientation, responsible for masking errors. Examples of these feature maps are shown in figures 6.12, 6.14and B.4 corresponding to equations 4.3, 4.7and 4.5 respectively.

Results of the distortion map clearly identified zones of errors and outperforms the spatial metric proposed. The comparison of the deviation of optical flow in the final panoramic frame with respect to the input videos could successfully capture geometrical artifacts. Figure 6.11 shows the highest response on the area where a person's face gets completely deformed. Other distorted regions are given lower intensities according to their degree of deviation.

The combined map calculated using equation 2.2.5 is shown in the next section with correspondence to the results of the user experiment.

Figure 6.12: Texture visibility map calculated using equation 4.3 and used to model texture in the salience map. An area of high texture like leaves of a tree will most likely mask an error.

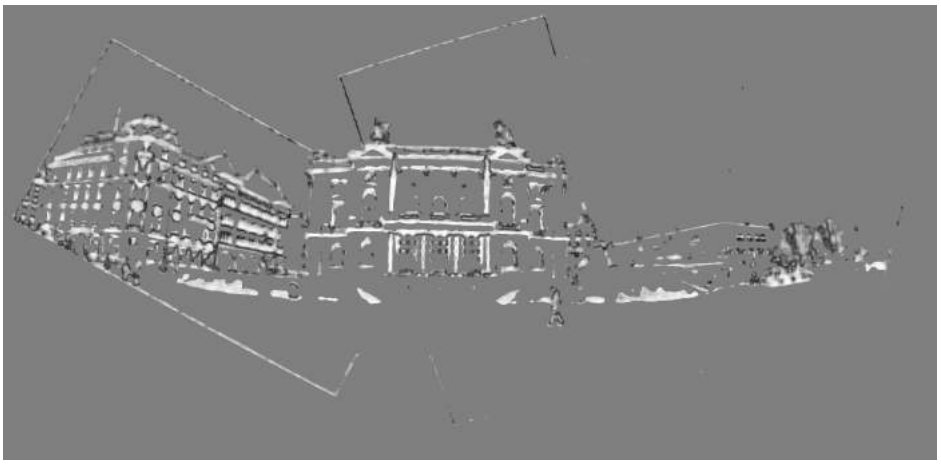


Figure 6.13: Orientation visibility map calculated using equation 4.5 and used to model variations in gradient orientation in the salience map. The orientation feature describes whether a textured area is uniform or not.The more regular the texture is the more visible a defect.

Figure 6.14: Contrast visibility map calculated using equation 4.7 and used to model contrast in the salience map. An error on a region with high contrast variation is more visible.

6.3 Method validation with human-based experiment

The experiment described in chapter 5 provided two types of data: gaze data from eye-tracking and frames annotated by participants. In order to compare these data with results of our optical flow-based metric, we used heat maps calculated using *Tobii Pro Lab* software to visualize fixations. Results are shown in figures 6.15. We could conclude that the highest fixation usually correspond to areas of high salience and/or a distorted region.

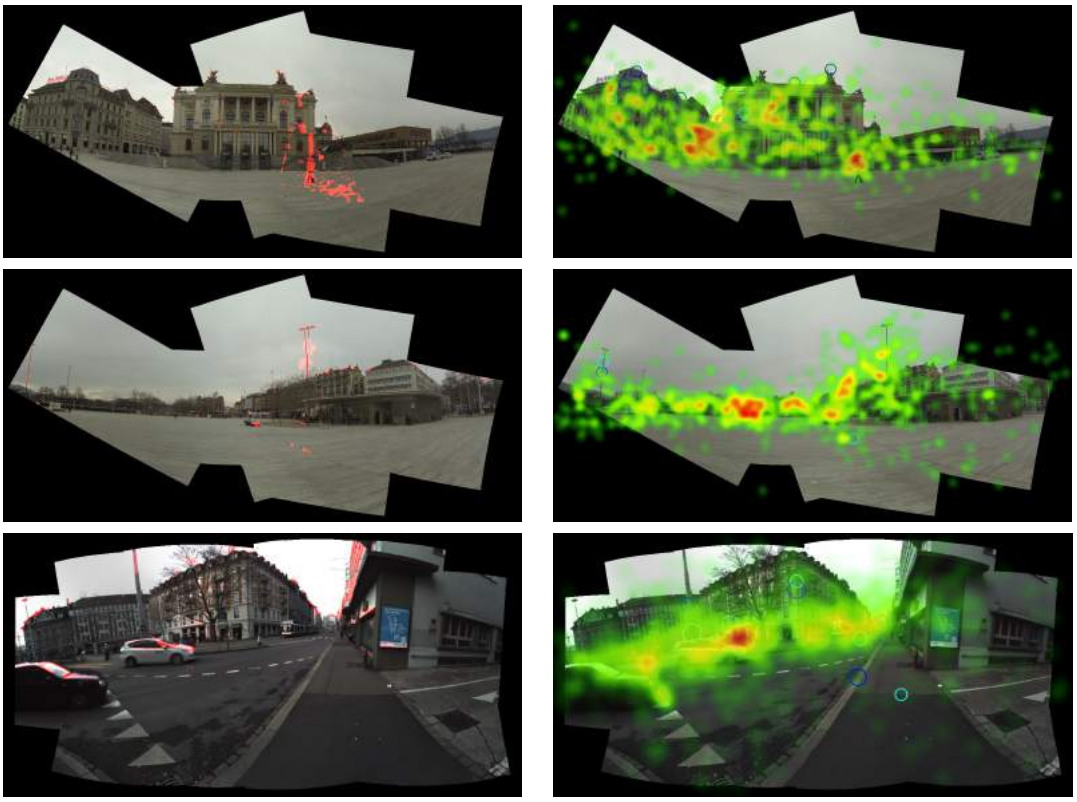


Figure 6.15: Left: calculated distortion-salience map using equation 2.2.5. Right: heatmaps of participants eye-tracking obtained using *Tobii Pro Lab*.

In addition, we used *Tobii Pro Lab* to calculate metrics in regions of the image where participants made annotations and regions where the algorithm signaled an error. This allowed us to draw statistics on the number of fixation in these regions. Figure 6.16 shows the total fixation count in regions annotated by humans as well as by the algorithm. It shows that people agreed with the result of the metric on the moving person being distorted while other regions have variable counts.

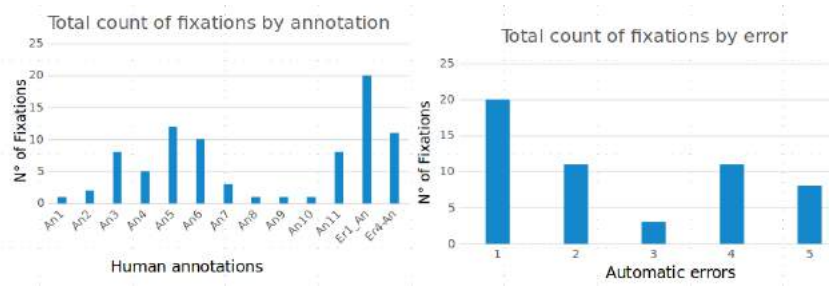


Figure 6.16: Total fixation count on regions annotated as errors by participants vs. errors detected by the algorithm.

Graph 6.17 shows the average number of participants who gazed a region annotated by at least one participant with respect to regions identified by the algorithm as error. The results on three different videos show different results. For the case of dataset *Street*, the errors identified by humans were nearly the same as those by the objective metric, which explains the close average of participants in both cases. Whereas, in video *Opera 1*, it seems that a higher number of participants noticed errors which agreed with what was identified by the algorithm.

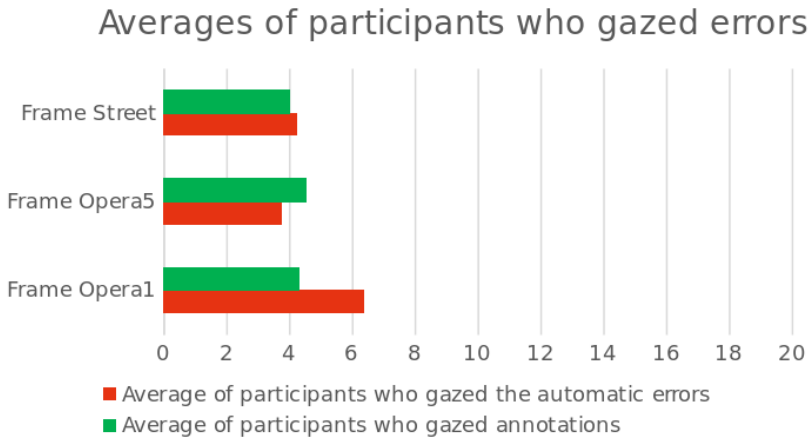


Figure 6.17: Average participants fixating on errors detected by algorithm vs. annotated by at least one participant.

Finally, we redefined our areas of interest to categorize errors that are most disturbing to humans as shown in figure 5.8. We extracted statistics

of several test videos at once to see what types of errors are more salient (figure 6.18. Results show that the human body deformation in the *Opera* sequence was the most perturbing one, while deformations in buildings in the background was less noticed. The ghosting in the other *Opera* sequence appears to be highly salient and we interpret this because of the high contrast and the regularity of the form of this object. Also, the absence of foreground in this sequence caused a redirection of the participants' attention to the static objects in the background. The emerging relatively fast car appearing in video sequence Street was slightly more visible to participants with respect to other types of defect.

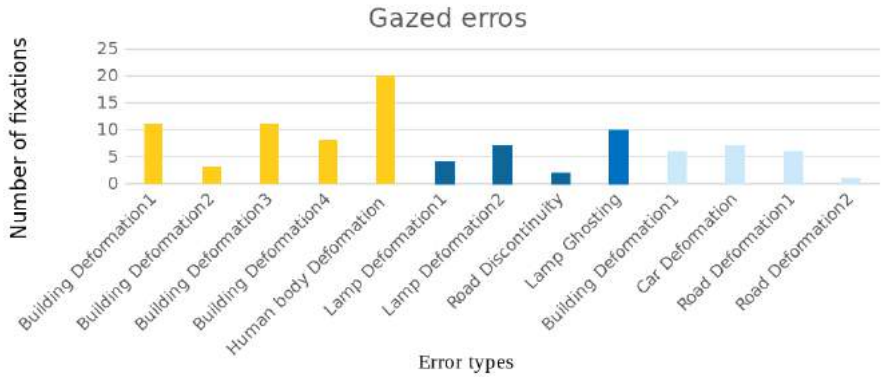


Figure 6.18: Number of fixations per error type to show the most disturbing types of errors.

6.4 Summary

In this chapter, results of the two proposed methods along with the human-based validation were shown. A discussion of the results is provided along with comparisons between the metric results and the human participants data. While the first proposed spatial metric gave more accurate results than the standard methods such as SSIM, it lacked the temporal aspect for assessing the video. The second metric incorporated a temporal feature using optical flow between consecutive frames and was combined with a saliency map. The second metric's results clearly outperformed the first one. The comparison of the second metric with the human participants gaze data and their annotations showed a good correlation. In addition, the statistics drawn

from the experiments allowed us to understand further about what drives human's vision when perceiving panoramic videos and which stitching errors are most disturbing. The next chapter presents the conclusion and limitations of the proposed approaches.

Chapter 7

Conclusion

7.1 Discussion

In this thesis, two methods are proposed to assess the quality of stitched panoramic videos. The first is a spatial metric based on an existing method from the state-of-the-art initially designed to evaluate the quality of novel-view synthesized images. The second includes a temporal feature using optical flow calculated between consecutive frames and a standard deviation of the motion fields from the input videos and the output panorama is used to create the distortion map. A salience map based on the same existing method used for the first metric is used and combined with the distortion map. The results of the second method clearly outperform those obtained by the first one, since it exploits the temporal feature.

To validate the second method further, a human-centered study was suggested based on error annotations and eye-tracking. The analysis resulted in a good correlation between the proposed method and what humans perceive as salient and/or mark as distorted. It also clarified some facts about human sensitivity to errors, mainly that humans are more likely to spot errors in salient regions especially in the foreground and tend not to notice the areas of distortions in the background. Statistics show that a human deformation would immediately catch the attention of most humans and that moving objects such as cars are more salient than static buildings and other background elements.

The objective metrics presented should be used to compare the quality of different stitching algorithms. They can also be used to optimize blending

methods by minimizing errors identified. The findings of the human-centered experiment can help create more accurate salience map that correspond better to human perception.

7.2 Limitations

Although the study conducted on humans showed that the suggested optical flow-based method was able to reflect a lot of areas of attention by humans, the study might be biased since participants were asked to annotate the errors they perceived rather than freely viewing the video. This is also the case with standard quality evaluation methods such as rating and scoring as was shown in the experiment done by Ninassi et al. [Ninassi et al., 2006]. A solution to that can be a study that consists of two groups of participants; one will be asked to freely view the videos and the other will view and annotate. This will permit to assess the effect of adding a task and how much it affects the results of gaze data. The analysis done on the collected gaze data can be improved by extracting metrics on eye saccades and motion pursuit which were theoretically explained in chapter 5. They should give richer insight and interpretation for the captured gaze since the stimulus used consisted of videos rather than still photos.

Another area of improvement is to combine methods for human and/or object detection with the calculated salience map which uses models contrast, texture and variation of texture orientation. This can correspond more to the findings of the experiment which show the high likelihood of fixations on foreground, especially humans and moving objects.

Finally, it will be interesting to expand the study to be tested on more videos with other structures and variations and especially 360° videos such as those taken within this work. A Tobii eye-tracking glasses for head-mounted displays is now available and can be tested to capture gaze data within a virtual reality environment.

Appendix A

Panoramic Image Projections

In order to represent the globe's sphere, cartographers use what is called "map projection" in order to transform the 3D spherical view to a 2D flat representation. Figure A.1 shows the way coordinate systems are represented before being projected. In the same way, when taking a panoramic video, the view is seen from the surrounding viewing circle of the cameras and to represent it on a flat surface, image re-projection is a necessity. Below is an overview of three commonly used projection surfaces for panoramas.

Equirectangular projection (also known as plat carré) is built by dividing a 2D rectangular surface into equal rectangles then map the latitudes and longitudes of the sphere to the grid. The main rectangle has almost its width double its height. This projection can show all the view of a 360° spherical surface, however it will suffer from high distortions near the north and south poles.

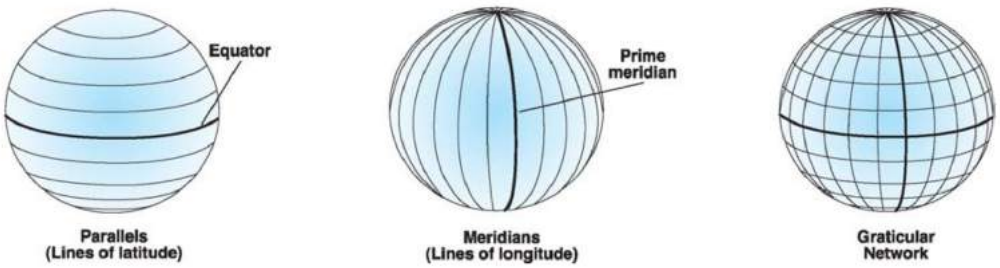


Figure A.1: Geographical coordinate systems and its terminology

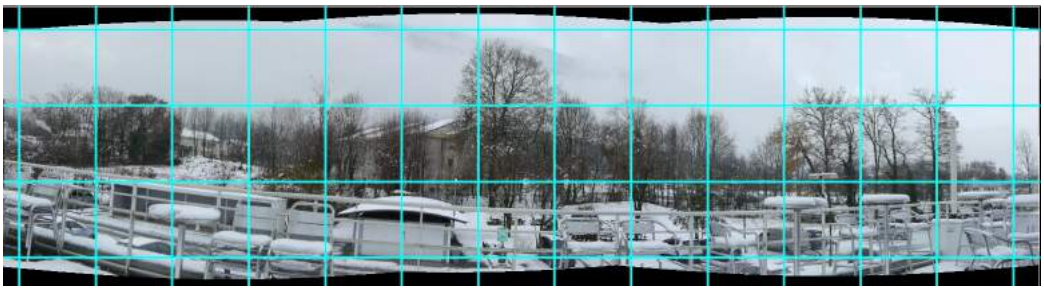


Figure A.2: Equirectangular projection from the interface of Hugin [PanoTools developers and contributors,] for a panoramic view taken by our 3-camera rig.

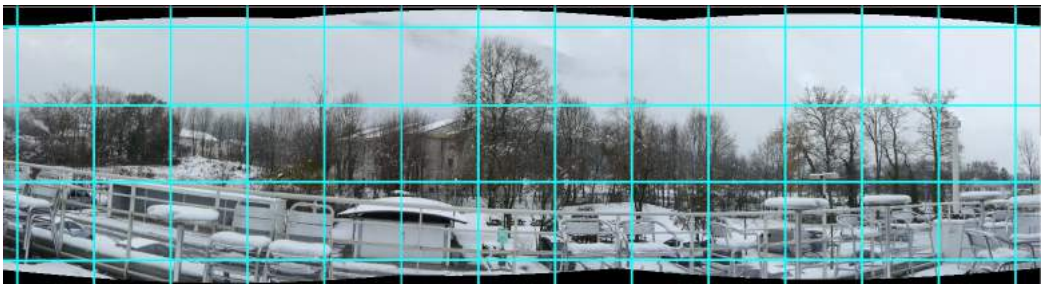


Figure A.3: Cylindrical projection from the interface of Hugin [PanoTools developers and contributors,] for a panoramic view taken by our 3-camera rig. It looks identical to equirectangular given the narrow vertical angle of view.

Cylindrical projection is similar to the equirectangular one, however the vertical lines are stretched to avoid distortions near the south and north, which is not suitable for the case of a wide vertical angle view and eventually not suited for a whole spherical 360° panorama.

Rectilinear projection, also known as flat or perspective projection, and it corresponds to the standard images we see and are familiar with. It works by mapping straight lines in 3D to straight lines in the flattened 2D surface. It is not suitable for images with an angle of view higher than 120°.

Fisheye projection aims to represent the 3D sphere such that the distance from the center of the 2D surface is proportional to the actual viewing angle.

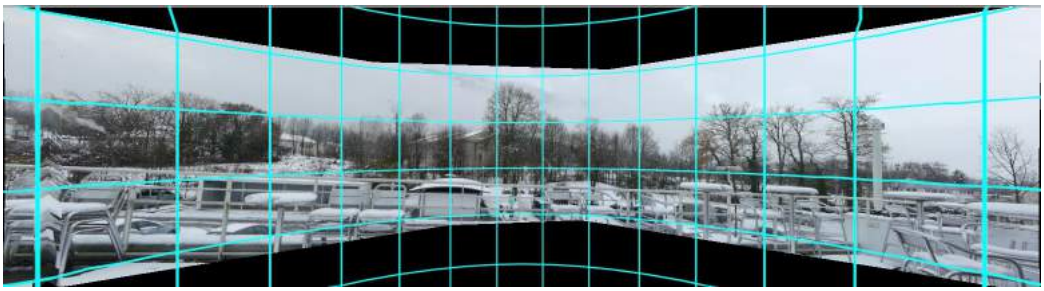


Figure A.4: Rectilinear projection from the interface of Hugin [PanoTools developers and contributors,] for a panoramic view taken by our 3-camera rig.

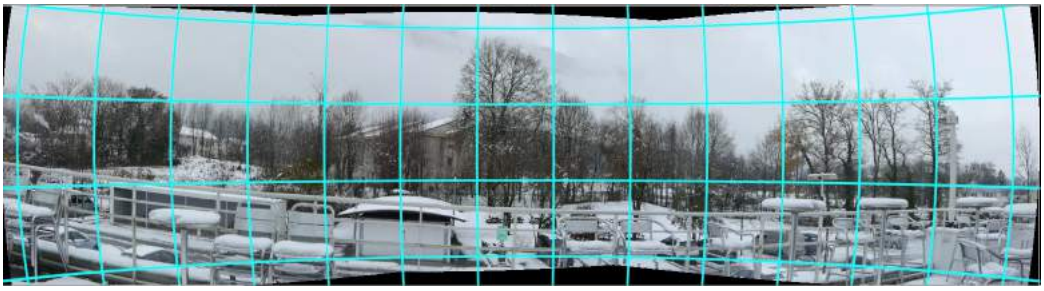


Figure A.5: Fisheye projection from the interface of Hugin [PanoTools developers and contributors,] for a panoramic view taken by our 3-camera rig.

This creates a grid that is more curved the farther away from the center (see figure A.5)and the resulting image is similar to the reflection of an image into a metallic sphere. It can be used for a wide angle panorama up to 180° but cannot be used for larger views.

Appendix B

Full Results

More examples of results from the proposed method 2 are shown in this appendix. Datasets used are *Opera*, *Street* and *Terrace* from Perazzi et al. [Perazzi et al., 2015] and dataset *Babyfoot* taken with our 3-camera rig.

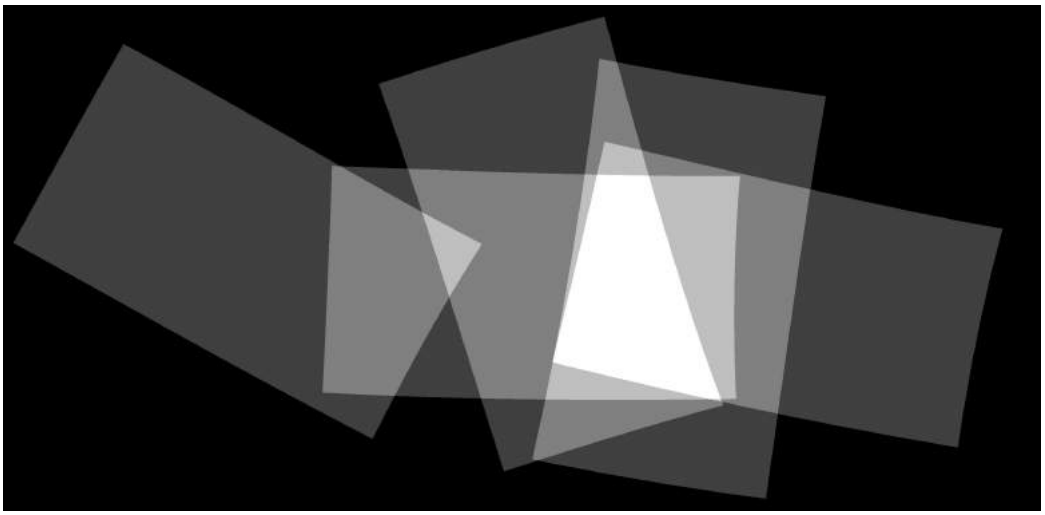
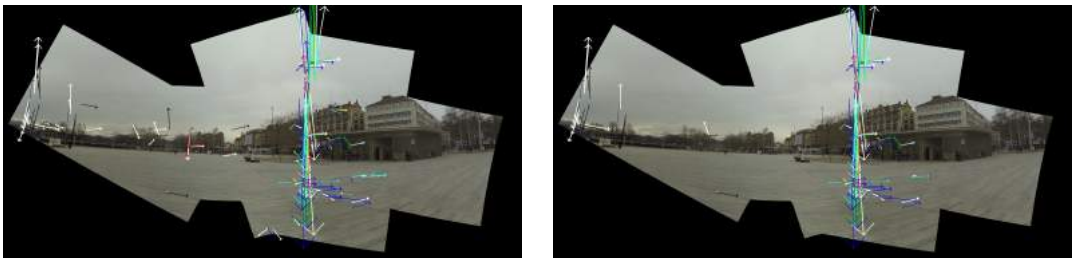


Figure B.1: Weighting map to ensure the contribution of each pixel in the image.



(a) (b)

Figure B.2: Optical flow with high deviation between final panorama and input views of a frame in *Opera* dataset (variance>50 and >80) .

Figure B.3: Distortion map on an example frame at $t = 385$ of *Opera* dataset.

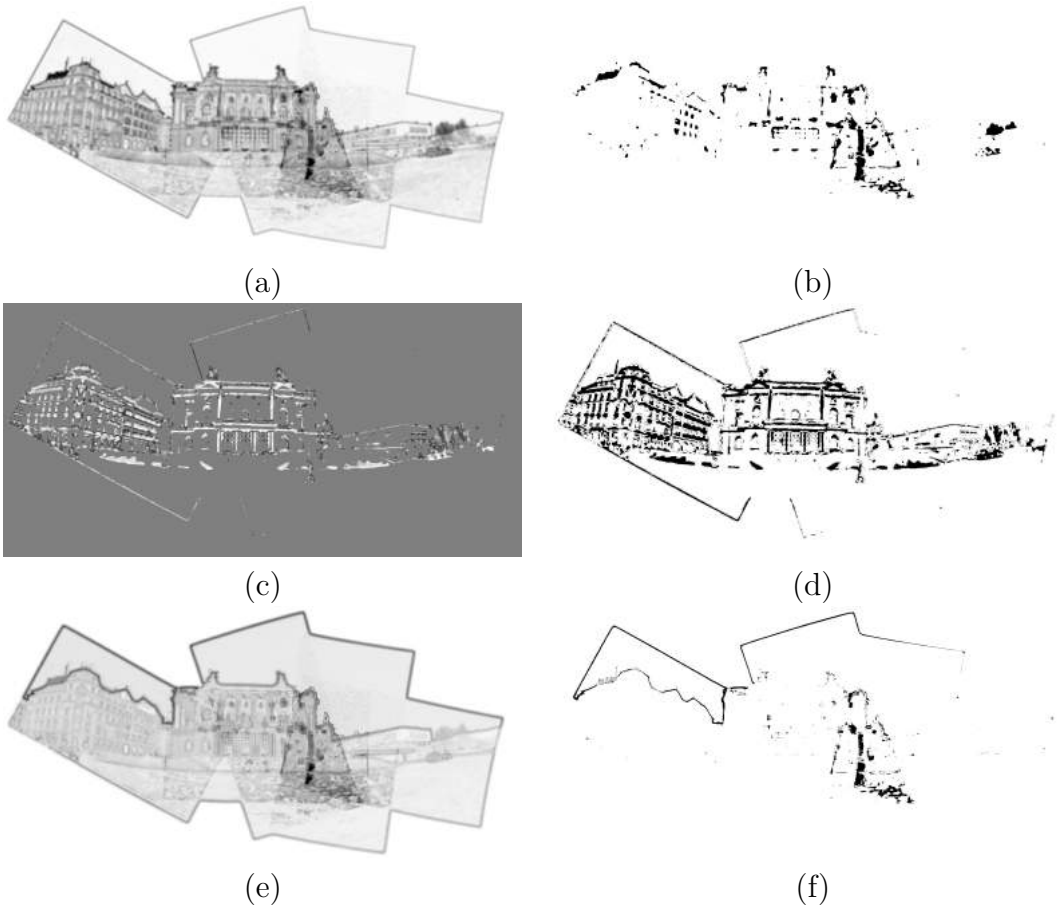
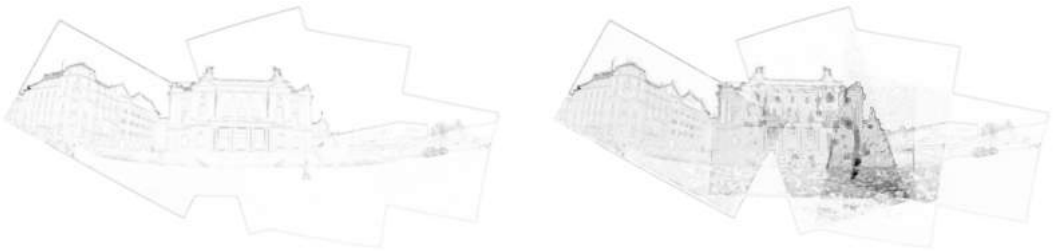


Figure B.4: Results of combining different salience maps [Conze et al., 2012] with our distortion map for frame 97. (a)Texture map+distortion map. (b) Thresholded (Texture map+distortion map). (c)Orientation map+distortion map. (d) Thresholded (Orientation map+distortion map). (e)Contrast map+distortion map. (f) Thresholded (Constrast map+distortion map).



(a) (b)

Figure B.5: (a)Salience map using equation 4.12. (b) Combined salience map+distortion map for frame 97 using equation 2.2.5 with equal weights.

Figure B.6: Texture visibility map for frame 385 as suggested by [Conze et al., 2012].

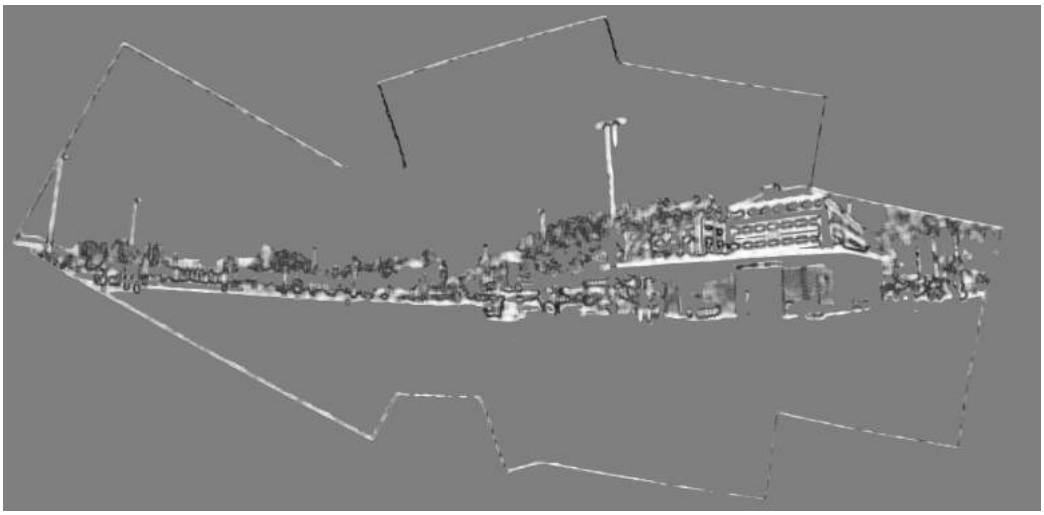


Figure B.7: Orientation visibility map for frame 385 as suggested by [Conze et al., 2012].

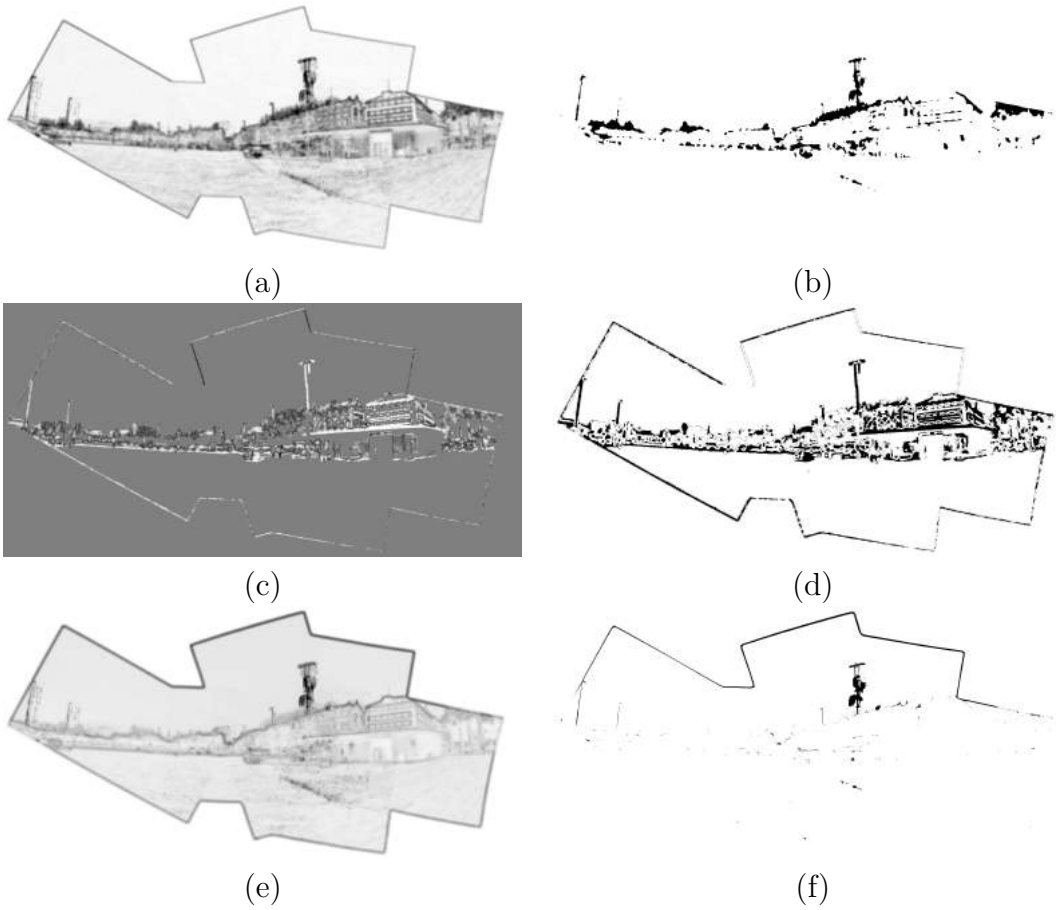
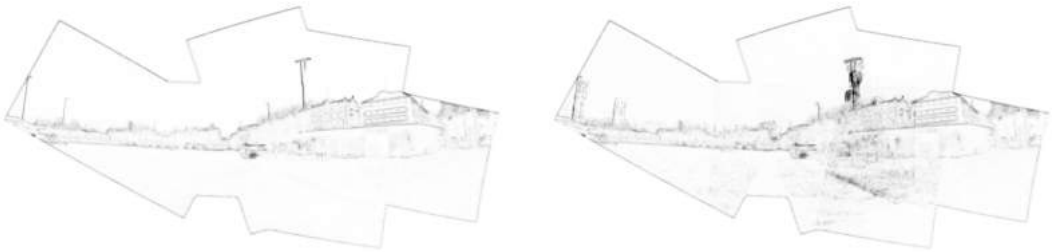


Figure B.8: Results of combining different salience maps [Conze et al., 2012] with our distortion map for frame 385. (a)Texture map+distortion map. (b) Thresholded (Texture map+distortion map). (c)Orientation map+distortion map. (d) Thresholded (Orientation map+distortion map). (e)Contrast map+distortion map. (f) Thresholded (Constrast map+distortion map).



(a) (b)

Figure B.9: (a)Salience map using equation 4.12. (b) Combined salience map+distortion map using equation 2.2.5 with equal weights for frame 385.

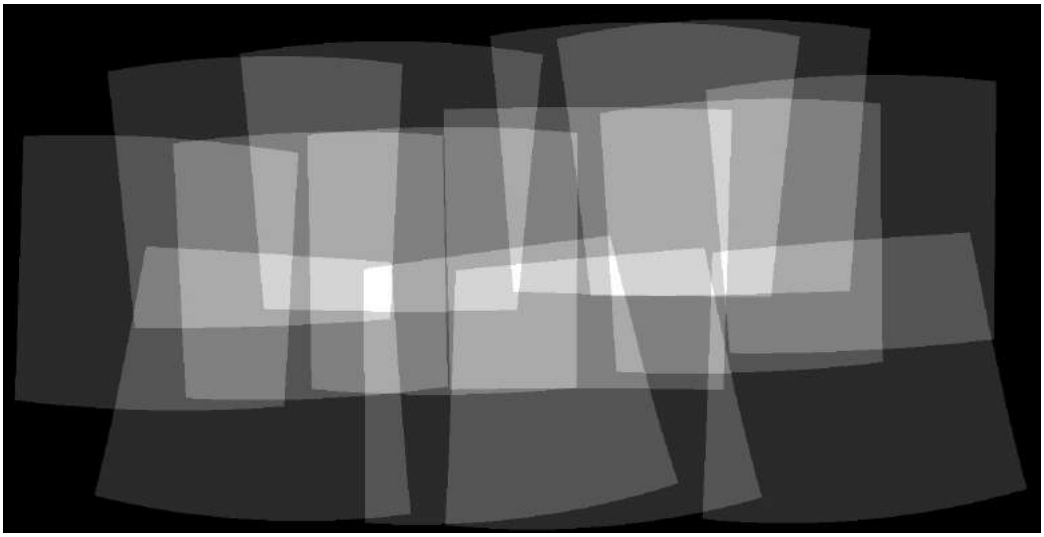


Figure B.10: Overlapping map used to calculate the the standard deviation (Since each pixel has 1 to N sources depending on the overlap).

Figure B.11: Visualization of motion fields in regions with high deviation between output view and input views.

Figure B.12: Distortion map calculated using standard deviation of displacement vectors.

Figure B.13: Salience map calculated on a frame of dataset *Street*.

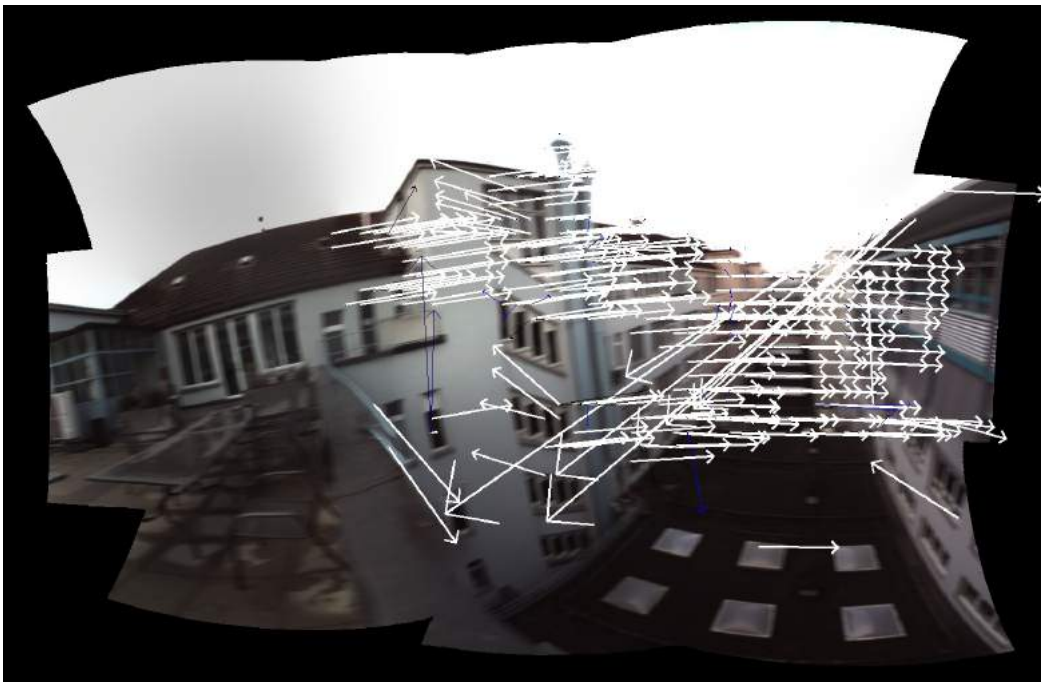


Figure B.14: A highly distorted frame from dataset *Terrace* shows the huge variance in flow fields in distorted areas.

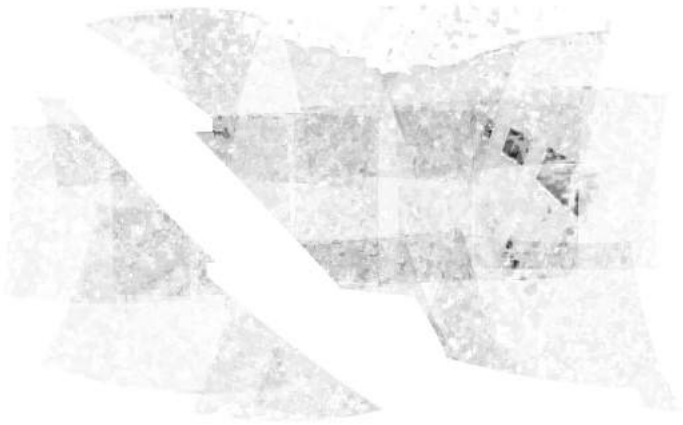


Figure B.15: Corresponding distortion map showing a lot of error zones.

Figure B.16: Combined map with weight of 0.8 to salience and 0.2 to distortion.

Figure B.17: Texture map on a frame of dataset *Babyfoot*

Figure B.18: Contrast map on a frame of dataset *Babyfoot*

Figure B.19: Variation of gradient orientation map on a frame of dataset *Babyfoot*

Figure B.20: Global salience map on a frame of dataset *Babyfoot*

Bibliography

[Aggarwal et al., 2016] Aggarwal, R., Vohra, A., and Namboodiri, A. M. (2016). Panoramic stereo videos with a single camera. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3755–3763.

[Amiqual4Home, 2016] Amiqual4Home (2016). Innovation factory. [Online].

[Anderson et al., 2016] Anderson, R., Gallup, D., Barron, J. T., Kontkanen, J., Snavely, N., Hernández, C., Agarwal, S., and Seitz, S. M. (2016). Jump: Virtual reality video. *ACM Trans. Graph.*, 35(6):198:1–198:13.

[Battisti et al., 2015] Battisti, F., Bosc, E., Carli, M., Le Callet, P., and Perugia, S. (2015). Objective image quality assessment of 3D synthesized views. *Image Commun.*, 30(C):78–88.

[Bay et al., 2008] Bay, H., Ess, A., Tuytelaars, T., and Gool, L. V. (2008). Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346 – 359. Similarity Matching in Computer Vision and Multimedia.

[Borji and Itti, 2013] Borji, A. and Itti, L. (2013). State-of-the-art in visual attention modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):185–207.

[Bosc et al., 2011] Bosc, E., Pepion, R., Callet, P. L., Koppel, M., Ndjiki-Nya, P., Pressigout, M., and Morin, L. (2011). Towards a new quality metric for 3-d synthesized view assessment. *IEEE Journal of Selected Topics in Signal Processing*, 5(7):1332–1343.

[Briggs, 2016] Briggs, F. (2016). Surround 360 is now open source. *https://code.fb.com/video-engineering/*.

[Brown and Lowe, 2003] Brown, M. and Lowe, D. G. (2003). Recognising panoramas. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV '03, pages 1218–, Washington, DC, USA. IEEE Computer Society.

[Brown and Lowe, 2007] Brown, M. and Lowe, D. G. (2007). Automatic panoramic image stitching using invariant features. *International Journal of Computer Vision*, 74(1):59–73.

[Brox et al., 2004] Brox, T., Bruhn, A., Papenberg, N., and Weickert, J. (2004). High accuracy optical flow estimation based on a theory for warping. pages 25–36.

[Burt and Adelson, 1983] Burt, P. J. and Adelson, E. H. (1983). A multiresolution spline with application to image mosaics. *ACM Trans. Graph.*, 2(4):217–236.

[Cabral, 2016] Cabral, B. K. (2016). Introducing facebook surround 360: An open, high-quality 3d-360 video capture system.

[Cheung et al., 2017] Cheung, G., Yang, L., Tan, Z., and Huang, Z. (2017). A content-aware metric for stitched panoramic image quality assessment. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 2487–2494.

[CMU, 2005] CMU (2005). Image pyramids and blending - computational photography lecture notes.

[Conze et al., 2012] Conze, P., Robert, P., and Morin, L. (2012). Objective View Synthesis Quality Assessment. In SPIE, editor, *Stereoscopic Displays and Applications*, volume 8288 of *Proc SPIE*, pages 8288–56, San Francisco, United States.

[El-Saban et al., 2010] El-Saban, M., Izz, M., and Kaheel, A. (2010). Fast stitching of videos captured from freely moving devices by exploiting temporal redundancy. In *2010 IEEE International Conference on Image Processing*, pages 1193–1196.

[Fischler and Bolles, 1981] Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395.

[Forsyth and Ponce, 2011] Forsyth, D. and Ponce, J. (2011). *Computer Vision: A Modern Approach. (Second edition).* Prentice Hall.

[Gegenfurtner, 2016] Gegenfurtner, K. R. (2016). The interaction between vision and eye movements. *Perception*, 45(12):1333–1357.

[Guo et al., 2016] Guo, H., Liu, S., He, T., Zhu, S., Zeng, B., and Gabbouj, M. (2016). Joint video stitching and stabilization from moving cameras. *IEEE Transactions on Image Processing*, 25(11):5491–5503.

[Hall and Guyton, 2011] Hall, J. and Guyton, A. (2011). *Guyton and Hall Textbook of Medical Physiology.* ClinicalKey 2012. Saunders/Elsevier.

[Hartley and Zisserman, 2003] Hartley, R. and Zisserman, A. (2003). *Multiple View Geometry in Computer Vision.* Cambridge University Press, New York, NY, USA, 2 edition.

[Heikkila and Silven, 1997] Heikkila, J. and Silven, O. (1997). A four-step camera calibration procedure with implicit image correction. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1106–1112.

[Ho et al., 2017] Ho, T., Schizas, I. D., Rao, K. R., and Budagavi, M. (2017). 360-degree video stitching for dual-fisheye lens cameras based on rigid moving least squares. *CoRR*, abs/1708.05922.

[Holmqvist et al., 2011] Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Jarodzka, H., and van de Weijer, J. (2011). Eye tracking: A comprehensive guide to methods and measures.

[Hu et al., 2017] Hu, H.-N., Lin, Y.-C., Liu, M.-Y., Cheng, H.-T., Chang, Y.-J., and Sun, M. (2017). Deep 360 pilot: Learning a deep agent for piloting through 360 sports video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

[Itti, 2003] Itti, L. (2003). Visual attention. In Arbib, M. A., editor, *The Handbook of Brain Theory and Neural Networks*, pages 1196–1201. MIT Press.

[Jiang and Gu, 2015] Jiang, W. and Gu, J. (2015). Video stitching with spatial-temporal content-preserving warping. In *2015 IEEE Conference*

on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 42–48.

[Juchmann, 2015] Juchmann, W. (2015). As lidar goes to hollywood, hypevr and velodyne are de ep in 3d. *LIDAR News Magazine*.

[K. and Channappayya, 2016] K., M. and Channappayya, S. S. (2016). An optical flow-based full reference video quality assessment algorithm. *IEEE Transactions on Image Processing*, 25(6):2480–2492.

[Kim et al., 2017] Kim, B. S., Choi, K. A., Park, W. J., Kim, S. W., and Ko, S. J. (2017). Content-preserving video stitching method for multi-camera systems. *IEEE Transactions on Consumer Electronics*, 63(2):109–116.

[Krafka et al., 2016] Krafka, K., Khosla, A., Kellnhofer, P., Kannan, H., Bhandarkar, S., Matusik, W., and Torralba, A. (2016). Eye tracking for everyone. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[L. Crowley, 1981] L. Crowley, J. (1981). *A representation of visual information*. PhD thesis, The robotics institute, Carnegie-Mellon University.

[Lee et al., 2016] Lee, J., Kim, B., Kim, K., Kim, Y., and Noh, J. (2016). Rich360: Optimized spherical representation from structured panoramic camera arrays. *ACM Trans. Graph.*, 35(4):63:1–63:11.

[Lee et al., 2017] Lee, W.-T., Chen, H.-I., Chen, M.-S., Shen, I.-C., and Chen, B.-Y. (2017). High-resolution 360 video foveated stitching for real-time vr. *Comput. Graph. Forum*, 36:115–123.

[Leorin et al., 2005] Leorin, S., Lucchese, L., and Cutler, R. G. (2005). Quality assessment of panorama video for videoconferencing applications. In *2005 IEEE 7th Workshop on Multimedia Signal Processing*, pages 1–4.

[Les, 2015] Les, S. (2015). Basics of stitching software for 360º video.

[Levenberg, 1944] Levenberg, K. (1944). A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics*, 2(2):164–168.

[Li et al., 2015] Li, J., Xu, W., Zhang, J., Zhang, M., Wang, Z., and Li, X. (2015). Efficient video stitching based on fast structure deformation. *IEEE Transactions on Cybernetics*, 45(12):2707–2719.

[Li et al., 2018] Li, N., Liao, T., and Wang, C. (2018). Perception-based seam cutting for image stitching. *Signal, Image and Video Processing*, 12(5):967–974.

[Limonov et al., 2018] Limonov, A., Yu, X., Juan, L., Lei, C., and Jian, Y. (2018). Stereoscopic realtime 360-degree video stitching. In *2018 IEEE International Conference on Consumer Electronics (ICCE)*, pages 1–6.

[Lin et al., 2016] Lin, K., Liu, S., Cheong, L.-F., and Zeng, B. (2016). Seamless video stitching from hand-held camera inputs. *Computer Graphics Forum*, 35(2):479–487.

[Liu and Heynderickx, 2011] Liu, H. and Heynderickx, I. (2011). Visual attention in objective image quality assessment: Based on eye-tracking data. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(7):971–982.

[Liu et al., 2015] Liu, X., Zhang, Y., Hu, S., Kwong, S., Kuo, C. C. J., and Peng, Q. (2015). Subjective and objective video quality assessment of 3d synthesized views with texture/depth compression distortion. *IEEE Transactions on Image Processing*, 24(12):4847–4861.

[Lowe, 1999] Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157 vol.2.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110.

[Luhmann, 2004] Luhmann, T. (2004). A historical review on panorama photogrammetry.

[MacQuarrie and Steed, 2017] MacQuarrie, A. and Steed, A. (2017). Cinematic virtual reality: Evaluating the effect of display type on the viewing experience for panoramic video. In *2017 IEEE Virtual Reality (VR)*, pages 45–54.

[Mandran, 2017] Mandran, N. (2017). *THEDRE : method of design research in computer science human centered : Traceable Human Experiment Design Research*. Theses, Université Grenoble Alpes.

[Mchugh, 2005] Mchugh, S. (2005). Cameras vs. human eye.

[Mohammadi et al., 2014] Mohammadi, P., Ebrahimi-Moghadam, A., and Shirani, S. (2014). Subjective and objective quality assessment of image: A survey. *CoRR*, abs/1406.7799.

[Nabil et al., 2018] Nabil, S., Balzarini, R., Devernay, F., and Crowley, J. L. (2018). Designing objective quality metrics for panoramic videos based on human perception. In *IMVIP 2018 - Irish Machine Vision and Image Processing Conference*, pages 1–4, Belfast, United Kingdom.

[Nguyen and Lhuillier, 2016] Nguyen, T.-T. and Lhuillier, M. (2016). Adding synchronization and rolling shutter in multi-camera bundle adjustment. In Richard C. Wilson, E. R. H. and Smith, W. A. P., editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 62.1–62.11. BMVA Press.

[Ninassi et al., 2007] Ninassi, A., Meur, O. L., Callet, P. L., and Barba, D. (2007). Does where you gaze on an image affect your perception of quality? applying visual attention to image quality metric. In *2007 IEEE International Conference on Image Processing*, volume 2, pages II – 169–II – 172.

[Ninassi et al., 2006] Ninassi, A., Meur, O. L., Callet, P. L., Barba, D., and Tirel, A. (2006). Task impact on the visual attention in subjective image quality assessment. In *2006 14th European Signal Processing Conference*, pages 1–5.

[PanoTools developers and contributors,] PanoTools developers and contributors. Hugin: Open source panorama stitcher.

[Perazzi et al., 2015] Perazzi, F., Sorkine-Hornung, A., Zimmer, H., Kaufmann, P., Wang, O., Watson, S., and Gross, M. H. (2015). Panoramic video from unstructured camera arrays. *Comput Graph Forum*, 34(2).

[Pérez et al., 2003] Pérez, P., Gangnet, M., and Blake, A. (2003). Poisson image editing. *ACM Trans. Graph.*, 22(3):313–318.

[Rai et al., 2017] Rai, Y., Callet, P. L., and Guillotel, P. (2017). Which saliency weighting for omni directional image quality assessment? In *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6.

[Salvucci and Goldberg, 2000] Salvucci, D. D. and Goldberg, J. H. (2000). Identifying fixations and saccades in eye-tracking protocols. In *Proceedings of the 2000 Symposium on Eye Tracking Research & Applications*, ETRA '00, pages 71–78, New York, NY, USA. ACM.

[Schatz et al., 2017] Schatz, R., Sackl, A., Timmerer, C., and Gardlo, B. (2017). Towards subjective quality of experience assessment for omnidirectional video streaming. In *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6.

[Seshadrinathan et al., 2010] Seshadrinathan, K., Soundararajan, R., Bovik, A. C., and Cormack, L. K. (2010). Study of subjective and objective quality assessment of video. *IEEE Transactions on Image Processing*, 19(6):1427–1441.

[Su et al., 2018] Su, J., Cheng, H., Yang, L., and Luo, A. (2018). Robust spatial–temporal bayesian view synthesis for video stitching with occlusion handling. *Machine Vision and Applications*, 29(2):219–232.

[Su et al., 2016] Su, Y., Jayaraman, D., and Grauman, K. (2016). Pano2vid: Automatic cinematography for watching 360 videos. *CoRR*, abs/1612.02335.

[Szeliski, 2004] Szeliski, R. (2004). Image alignment and stitching: A tutorial. Technical report.

[Szeliski, 2010] Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Springer-Verlag, Berlin, Heidelberg, 1st edition.

[Torralba et al., 2010] Torralba, A., Russell, B. C., and Yuen, J. (2010). Labelme: Online image annotation and applications. *Proceedings of the IEEE*, 98(8):1467–1484.

[Treisman and Gelade, 1980] Treisman, A. M. and Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97 – 136.

[Tsai, 1987] Tsai, R. (1987). A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal on Robotics and Automation*, 3(4):323–344.

[Upenik et al., 2017] Upenik, E., Rerabek, M., and Ebrahimi, T. (2017). On the performance of objective metrics for omnidirectional visual content. In *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6.

[Vranješ et al., 2013] Vranješ, M., Rimac-Drlje, S., and Grgić, K. (2013). Review of objective video quality metrics and performance comparison using different databases. *Signal Processing: Image Communication*, 28(1):1 – 19.

[Wang et al., 2014] Wang, O., Schroers, C., Zimmer, H., Gross, M., and Sorkine-Hornung, A. (2014). Videosnapping: Interactive synchronization of multiple videos. *ACM Trans. Graph.*, 33(4):77:1–77:10.

[Wang et al., 2004] Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612.

[Webster et al., 1993] Webster, A. A., Jones, C. T., Pinson, M. H., Voran, S. D., and Wolf, S. (1993). Objective video quality assessment system based on human perception. In Allebach, J. P. and Rogowitz, B. E., editors, *Human Vision, Visual Processing, and Digital Display IV*, volume 1913 of , pages 15–26.

[Wikipedia Contributors,] Wikipedia Contributors. Discovery of linear perspective by Bruneleschi.

[Wikipedia contributors, 2018] Wikipedia contributors (2018). Panoramic photography — Wikipedia, the free encyclopedia. [Online; accessed 12-August-2018].

[Xu et al., 2017] Xu, M., Li, C., Wang, Z., and Chen, Z. (2017). Visual Quality Assessment of Panoramic Video. *ArXiv e-prints*.

[Yarbus, 1967] Yarbus, A. L. (1967). *Eye Movements and Vision*. Plenum. New York.

[Zhang et al., 2017] Zhang, B., Zhao, J., Yang, S., Zhang, Y., Wang, J., and Fei, Z. (2017). Subjective and objective quality assessment of panoramic videos in virtual reality environments. *2017 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, 00:163–168.

[Zhang, 2000] Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334.

[Zheng et al., 2008] Zheng, M., Chen, X., and Guo, L. (2008). Stitching video from webcams. In Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Porikli, F., Peters, J., Klosowski, J., Arns, L., Chun, Y. K., Rhyne, T.-M., and Monroe, L., editors, *Advances in Visual Computing*, pages 420–429, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Zoric et al., 2013] Zoric, G., Barkhuus, L., Engström, A., and Önnevall, E. (2013). Panoramic video: Design challenges and implications for content interaction. In *Proceedings of the 11th European Conference on Interactive TV and Video*, EuroITV '13, pages 153–162, New York, NY, USA. ACM.