TÉCNICO LISBOA

Plenoptic Face Reconstruction

Gonçalo António dos Santos Cruz e Carreira Pedro

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisor

Professor José António da Cruz Pinto Gaspar
Engineer Nuno Miguel Barroso Monteiro

Examination Committee

Chairperson: Professor João Fernando Cardoso Silva Sequeira
Supervisor: Professor José António da Cruz Pinto Gaspar
Member of the Committee: Professor Nuno Miguel Mendonça da Silva Gonçalves

October 2021

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Agradecimentos

Em primeiro lugar gostaria de expressar a minha gratidão aos meus orientadores, Professor José Gaspar e Engenheiro Nuno Monteiro, que me estimularam o pensamento, ajudaram a trazer o meu trabalho a um patamar superiror, cuja experiência e orientação foram imprescindíveis. Isto, ao mesmo tempo que me apoiaram e permitiram discussões científicas enriquecedoras. Tenho ainda a agradecer ao meu colega Miguel Rodrigues, que trabalhou lado a lado comigo no final desta etapa e me facultou dados reais para efetuar experiências.

Aos meus pais, que sempre me apoiaram durante esta jornada, um grande obrigado! Por tudo, pela educação que me deram, todo o apoio, coragem, e cujo sacrifício permitiu que eu aqui chegasse de cabeça erguida. Aos meus tios e tias, um muito obrigado por acompanharem esta jornada de perto e sempre com entusiasmo.

Um grande agradecimento é também devido aos meus amigos. À Raquel, Ana, Maria, Mariana, Sérgio, Nuno e Mateus: um gigante obrigado por me darem sempre alento, por me ajudarem a manter a sanidade (menção honrosa ao Horizonte neste ponto), por acreditarem em mim, pela coragem, pela paciência e pelo apoio incondicional não só durante esta fase da minha vida, mas sempre! Àqueles que me acompanharam nos anos que precederam este trabalho, David, Francisco, Filipe, Pedro, Samuel, Paulo e tantos outros que participaram nesta jornada comigo, obrigado. Pelas refeições em conjunto, pelas conversas sobre tudo e mais alguma coisa, pelas pausas no trabalho, toda a diversão e camaradagem que nos permitiu completar esta etapa juntos.

Por fim, agradeço a todos aqueles que não se encontram acima mencionados mas que me apoiaram, acreditaram em mim e me encorajaram a tornar-me uma melhor versão de mim próprio.

A todos e a cada um, o meu muito obrigado!

Resumo

O uso de informação biométrica é uma necessidade emergente na nossa sociedade, com a biometria facial a fazer parte da escolha natural em processos de autenticação. Modelos bidimensionais de faces permitem ataques de apresentação facial, efetuados recorrendo a fotografias ou ecrãs. Para elevar os níveis de precisão, fidelidade e segurança em autenticação por biometria facial é necessário investigar modelos tridimensionais. Este trabalho investiga a aplicação de câmaras plenópticas à reconstrução 3D de faces, com foco em zonas de gradiente baixo. As câmaras plenópticas captam diferentes pontos de vista simultaneamente, permitindo extrair informação 3D de cenas (reconstruir) através da redundância de informação.

As metodologias de reconstrução atuais, baseadas em *edge-points*, mostram dificuldades em zonas de baixo gradiente. Nesta tese, é conduzido um estudo preliminar para aferir se zonas de baixo gradiente possuem informação para reconstrução, recorrendo a light field *shearing* em dados sintéticos. Concluiu-se que, dividindo zonas de baixo gradiente (não nulo) em segmentos suficientemente grandes, é possível recuperar informação para reconstrução. É proposto um método para reconstrução facial, que visa complementar os métodos *edge-based*, recorrendo à aplicação de light field *shearing* a segmentos delimitados por, mas não contendo, gradientes elevados.

As experiências efetuadas utilizando o método proposto, em dados sintéticos e reais, mostram uma reconstrução consistente em zonas de baixo gradiente. O método proposto para reconstrução facial fornece informação capaz de complementar a reconstrução efetuada por métodos *edge-based*. A performance de dois métodos de reconstrução alternativos, conhecidos na literatura, foi analizada no contexto de reconstrução de faces tendo o método proposto fornecido resultados comparativamente promissores.

Palavras chave: Imagiologia Plenóptica, Arranjo de Câmaras, Reconstrução 3D, Confiança na Reconstrução, Reconstrução em Zona de Baixo Gradiente

Abstract

In our evolving society the use of biometric information is an increasingly pressing need, with facial biometrics having a large expected role in authentication applications. Two dimensional face models allow presentation attacks, based on photographs or displays. Research in the three dimensional face models is necessary to achieve higher levels of accuracy and reliability resulting in improved security. This work aims to investigate the use of plenoptic cameras in 3D face reconstruction, focusing on the reconstruction of low gradient areas. Plenoptic cameras capture a scene from different viewpoints, and store the information in a single image sensor. That information is sufficient to acquire 3D information from scenes, i.e., perform reconstruction.

Current face reconstruction methodologies based on edge-points reveal difficulties within low gradient areas. In this thesis, a preliminary study, conducted on synthetic data, assesses whether low gradient areas encompass relevant reconstruction information, considering light field shearing. The study shows that segmenting low (non-zero) gradient areas into large enough patches allows finding sufficient registration information. The application of light-field shearing on patches bordered by, but not containing, high gradients, is the basis proposed for a face reconstruction method which can complement edge-based reconstruction methodologies.

Experiments on synthetic and real data show consistent depth estimation in low gradient areas using the proposed method, being capable of providing additional information to edges-based reconstruction. Two well known reconstruction methodologies have been analyzed for their performance in the context of face applications, and the proposed reconstruction method has been found to provide promising comparison results.

Keywords: Plenoptic Imaging, Camera Array, 3D Reconstruction, Reconstruction Confidence, Low Gradient Depth Estimation.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Three-dimensional face models are widely used for biometric systems, face verification, facial expression recognition, 3D visualization, and so on. However, reconstructed models of the face generated from the optical setups used are quite noisy, due to the lack of texture and thin structures present in the face.

Plenoptic cameras [36, 39] are capable of imaging a scene from different perspectives. The information of the different perspectives is stored on a single image sensor which allows the acquisition of dynamic scenes and perform 3D reconstruction easily [35].

This thesis is developed in the framework of the research project proposal "Plenoptic Face Imaging and Biometrics in Identity Documents for Security Applications" (*PlenoFace*). *PlenoFace* focuses on strong authentication combined with robust ID-docs, with a scope ranging from hardware selection, ID-doc manufacturing, and the authentication itself. For more information regarding the project and application cases please refer to annex A. The research motivation of this thesis is exploring the use of light field imagery, acquired with plenoptic cameras [36, 39], to perform 3D reconstruction which may aid face detection and recognition processes.

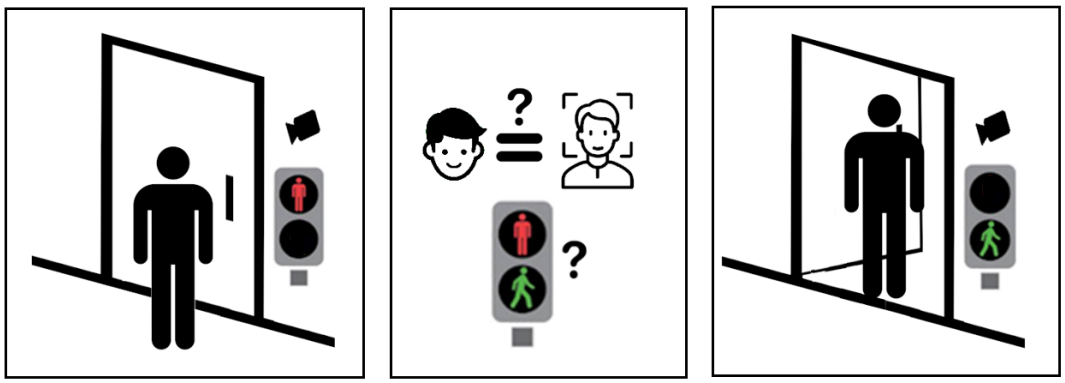
1.1 Face Reconstruction in Security Applications

An application of face reconstruction is using facial imaging to access a secured place, as presented in the storyboard of Fig. 1.1. A person, who has a picture of his face previously stored in a local access-control database, arrives at the entrance of a secured location. A picture of the face is automatically captured, as seen in Fig. 1.1(a).

The captured picture is compared to the images stored in the access-control database. This enables the validation of the person's ID and verification of permissions to access the area Fig. 1.1(b). If it is a match, and the access is granted, the person can enter the premises, Fig. 1.1(c). For further insight on possible applications of this research please refer to annex A.

Three-dimensional face models are widely used in biometric systems. The quality of the 3D reconstruction is crucial for these systems, and reconstructed face models are noisy, due to the lack of texture and the thin structures present in the face.

Presentation Attack Detection (PAD) is currently recognized as a need for biometric systems [42, 41]. Plenoptic (light field) cameras enjoy similar advantages to the conventional RGB cameras nowadays, such as being point and shoot, portable and having low cost. Therefore, sensors such as plenoptic cameras can help developing new PAD



(a) (b) (c)

Figure 1.1: Storyboard: using face recognition to access a secured place. In (a) a person arrives at the entrance of a secured location and has their picture taken. In (b) their ID and permissions are verified in the access control database. In (c) access is granted only to authorized personnel.

solutions [46, 47].

Furthermore, the ubiquity of smartphones makes them the device of choice for strong authentication. In [33] Mildenhall et al. propose a light field acquisition setup based on images acquired by smartphones and view synthesis. This work, further adds to the argument by opening the door for the general public to acquire light field imagery at a low cost.

1.2 Reconstruction in Facial Biometrics

Facial structure is a widely accepted biometric in security applications. Facial biometry systems originated in the military in the 1960s and started as manual systems. In the 1990s this area of research was dominated by solutions mapping the input to a lower-dimensional space, like eigenfaces determined by principal component analysis (PCA) [49], independent component analysis [5] or kernel PCA [2], using conventional cameras.

Later, model based solutions were developed to overcome sensitivity to scale, pose and facial expression. These solutions derive geometrical features [54], whereas multiple scales, orientations, and frequency bands are addressed by descriptors such as: local shape map [56], local binary patterns or local phase quantization [3].

Recent research was instigated by the development of deep neural networks, namely convolutional neural networks (CNN), yielding promising results. For example, the *GaussianFace* algorithm (2014) developed by the University of Hong Kong [29], surpasses humans' identification score. Facebook's *DeepFace* [37] achieved an almost like human identification score. Google's *FaceNet* [44] achieved a new record of identification score of over 99.5%. Other methods arose, such as recovery of a 3D model from a single image [55], resorting to auto-encoder chains. However, sophisticated forgery techniques also leveraged this technology, such as deepfake image/video generation [53].

Application to light fields had significantly less research efforts. Feng et al. proposed *FaceLFNets* [18], taking advantage of the 3D information extracted from light field imagery and the learning capabilities of CNNs, allowed the model to recover horizontal and vertical 3D facial curves. Using a curve by curve reconstruction approach, this method generalizes well for new data, thus, requiring few training samples. Also, it presented a reduction of reconstruction errors by over 20% comparing to the state of the art. Furthermore, the use of a model-free approach

provides additional robustness to changes in pose, facial expressions, ethnicity and illumination.

Alperovich et al. proposed a Deep Encoder-Decoder network [4] to extract light field intrinsics in an unsupervised manner. Namely, it performs disparity estimation, diffuse/specular component separation and light field reconstruction. Since it performs the separation between specular and diffuse components it is able to estimate depth in highly specular scenes [4].

In the line of handling specularity in light fields Johannsen et al. proposed a sparse coding [26] approach to detect if a light field possesses multiple layers. To perform specularity separation a Gaussian is fitted to the sparse coding coefficients. This helps to determine if specularity is present, and uses the appropriate two-layer model if so, which then enables disparity estimation to be performed.

Regarding general light field methods, Ferreira et al. [21, 20, 19] also developed work in the field of reconstruction, focusing on the depth estimation aspect. Proposing a fully automated method for depth estimation from a single light field image, with results comparable to the state of the art achieved in significantly less time, already considering a trade-off between computation time and accuracy. The method uses different focal length lenses, so it is functional even without the calibration data of the micro lens array, and enables all in focus renders with the dense depth map generated.

Concerning light field applied to faces, Dihl et al. [15] proposed a content-aware filtering methodology for 2.5D meshes of faces, recovered from light field images or images acquired with sensors like Kinect. This method preserves intrinsic features resorting to exemplar-based neighborhood matching. First, using previous knowledge of the model's geometry to improve matching, and then determining which regions have the same intrinsic geometric similarities to be compared. Thus, it is able to remove noise of a 2.5D face.

1.3 Problem Formulation

The European Union currently faces a challenge to protect its citizens' freedom and security without compromising their privacy nor limit their freedom. Directives imposing strong (multi-factor) authentication are already in place, including biometric information.

The use of biometric information to provide stronger authentication is becoming increasingly important for human activities such as banking or internet based businesses. Nowadays, face biometry is already used to unlock smartphones and computers. However, new methodologies need to be investigated, such as 3D face models, in order to raise security to the highest level possible.

The objective of this thesis is to help provide facial biometric information to help authentication processes, specifically, in the area of 3D face reconstruction. The ultimate goal is to have a solution that captures an image and is able to produce a three-dimensional model of the face that can be used by recognition and/or authentication processes.

A way to provide depth information, from a single image, is resorting to plenoptic setups. The images captured by these setups can be used to retrieve three-dimensional information, and consequently, reconstruct the 3D structure of a face.

1.4 Thesis Structure

Chapter 1 introduces the problem to approach in the thesis, in particular presents a short discussion on the state of the art on facial biometrics and reconstruction. Chapter 2 introduces face modeling, then presents background and related work on modeling plenoptic cameras and reconstruction methodologies. Chapter 3 introduces the shearing operation, then presents a series of conceptual experiments that culminate in a depth reconstruction method for low gradient areas. Chapter 4 expands the method presented in the previous chapter to be applicable to face reconstruction, resulting in the proposed method for face reconstruction. Chapter 5 details the experiments performed with faces, from the creation of the synthetic face models to the results attained with several reconstruction methodologies. Chapter 6 summarizes the developed work and highlights the main achievements. Moreover, this chapter proposes further work to extend the activities described in this document.

Chapter 2

Background on Plenoptic Imaging and Facial Biometrics

As mentioned in Chapter 1, plenoptic cameras enable single shot capture of sufficient information to retrieve 3D structure of the world. To comprehend how light fields yield this information it is necessary to model the plenoptic camera. Furthermore, an introduction to the process of camera calibration is required to understand how the recovery of world information from the captured imagery is performed. Afterward, we can study reconstruction methodologies that enable the recovery of metric information. The contents on this chapter follow this order of needs, starting with the plenoptic camera model, the calibration process and background on reconstruction methodologies.

However, before proceeding into the camera model we describe the fundamentals of face modeling which enable the creation of synthetic data required for the experimentation phase.

2.1 Face Modeling

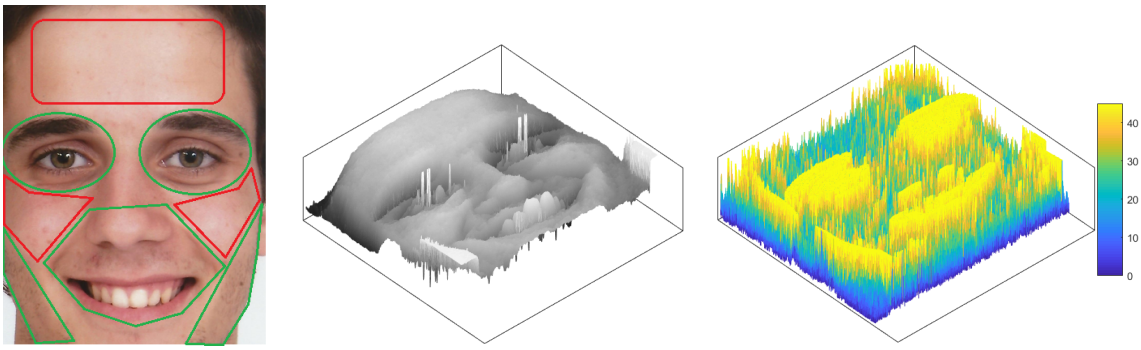
The confinement imposed due to the Covid-19 pandemic prevented the acquisition of real datasets. Therefore, a choice was made to generate synthetic face datasets, which requires modeling faces. To generate human face models a reconstruction process is required to have them looking as real as possible. Modeling of 3D faces is a popular area in Computer Graphics and Computer Vision. It requires a translation from two dimensional space (picture) to the three dimensional space (face model). Several techniques have been created for this purpose, using one or multiple cameras, 3D scanners, and combinations of sophisticated software and hardware. This section covers the fundamentals of the methodology applied in section 5.1.

Face Models

The human brain has a predisposition toward facial recognition. Humans use the face as the main distinguishing feature between people due to its discernible features, such as eye and hair color, shape of sensory organs and wrinkles. Infants are biologically programmed to recognize subtle differences in anthropomorphic facial features, developing these skills to nearly adult levels before any other types of images [17].

The discernible features are generally related to high gradients in the image, such regions are highlighted in green Fig. 2.1 (a). A detail of gradient information is provided through a mesh plot in Fig. 2.1 (c) where we can observe high gradients (in yellow) associated with the eyes, nostrils, mouth or expression wrinkles.

However, faces are also comprised of smooth areas, with low gradients, which are highlighted in red in Fig. 2.1 (a) and correspond to the green/blue regions in Fig. 2.1 (c). Nonetheless, as seen in Fig. 2.1 (b) these regions still have high intensity level, so we assume that the captured rays still possess reliable information.



(a) Face region highlight (b) Gray level image (c) Gradient of the image

Figure 2.1: Types of gradient regions found in the face. In (a) a highlight of face regions, with red outlining low gradient regions and green the high gradient regions. In (b) a mesh plot of the gray scale image that reveals pixel intensity. In (c) a mesh plot of gradient information found in the face.

Model Generation

The problem of face modeling, can be divided in three steps: data acquisition; 3D registration followed by 3D model deformation; and texture generation to cover the 3D model [16].

The data acquisition is the process of capturing the reality through photographs. The required resolution depends on the number of feature points in the projection pattern to be used, more points imply higher resolution required but yield better results.

Then, features are extracted with the help of a mesh, connecting a dot pattern, that is projected in the subject's face. Feature points are meant to be easy of detect regardless of the perspective, for instance: inner and outer corners of eyes; upper and lower connections of the ears to the face or corners of the mouth.

The registration of the feature points is then performed, resulting in 3D coordinates for the feature points. The calculation is done according to a standard 3D face model, fixing the height to the model and leaving width and depth free for adjustment. The deformation is then performed, adjusting the position of the standard model's feature points to match the registered feature points.

The final step is texture generation, which requires the reference images, a texture map and the deformed face mesh. The process consists on gathering color values from the original images and collect them in the appropriate locations of the texture image, now transformed to fit the texture map.

Filtering of Meshes of Faces

Alternatively, 2.5D cameras can be used to obtain a face model. However, these meshes are very noisy leading to mistakes on person recognition, pose detection and facial expression recovery.

Assuming prior knowledge about the structure of faces, one can match local features because of intrinsic geometric similarities despite the difference in the macro features. Dihl et al. [15] propose using an exemplar-based neighborhood matching to filter the intrinsic features to be preserved, while simultaneously the geometric nature of the model helps to improve matching. Also, facial feature points are used to define regions in which all points have intrinsic geometric similarities [15].

The neighborhood of each point on the model is compared to the neighborhoods at the exemplars, and its position replaced by the position of the respective best match. This search is done using a Nearest Neighbor method implemented by Kd-tree with PCA [23].

Then, an improvement of the neighborhood matching is performed to cope with macro-characteristics that are not in the exemplars. In such case an adjustment is made to comprise these similarities. The search is constrained corresponding regions, therefore there must be a subdivision of regions, and this process shall meet three conditions:

1. Existence of feature points, at correspondent places, for all faces;

2. The union of regions must cover the whole face;

3. The area of each region must be inversely proportional to the expected noise;

The process to meet these criteria begins with a definition of facial features (condition 1) [50], addition of further points generated by geometric operations on the existing feature points (condition 2) and a deformation of the models according to feature points. Now it is possible to compare each model to a filtered version, create an error map, calculate the expected error of the region, and find the smallest size for the larger region such that all sets of regions cover the respective model (condition 3).

2.2 Plenoptic Imaging

Conventional cameras are inspired on the behavior of human eyes, comprising a sensor and a main lens. Plenoptic cameras are also inspired in a biological design, the compound eye found in some insects and crustaceans, such as bumblebees and mantis shrimps.

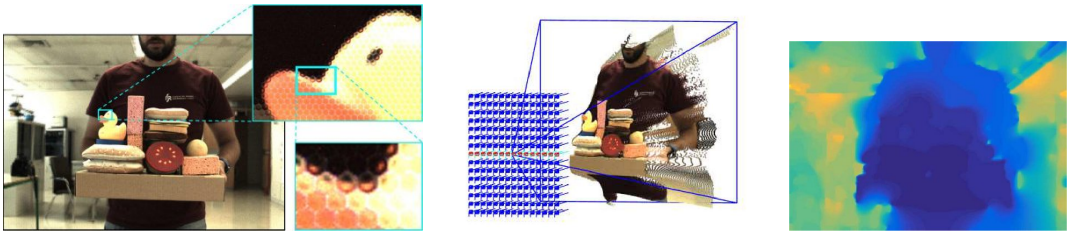
The most common plenoptic cameras are built based on an array of microlenses in front of a main lens, to create an artificial compound eye (see Fig. 2.2(a)). The main lens is designed to be well approximated by a thin lens model. Each microlens is modeled as a pinhole that captures a slightly different perspective from its neighboring lenses (Fig. 2.2(b)), thus enabling depth estimation of the scene (Fig. 2.2(c)).

The definition of plenoptic function arises from the question “What can potentially be seen?” [1]. The plenoptic function represents the light intensity spectrum, for all directions at every 3D point, along time. It is characterized by seven parameters:

- (x, y, z) spacial coordinates or viewing position;

- (az, el) angular coordinates (azimuth and elevation) or ray parametrization;

- (λ, t) wavelength or color, and time instant.



(a) (b) (c)

Figure 2.2: Camera array and light field scene reconstruction. In (a) raw light field image and zoom of the microlenses. In (b) camera array, showing the central camera field of view, with distances between projection centers multiplied 50 times and obtained 3D reconstruction. In (c) reconstructed depth map. Extracted from [34].

Plenoptic cameras [43] acquire light field images, which are 4D samples of the 7D plenoptic function. These images are obtained considering: one time instant, constant radiance along the rays and still monochromatic imaging; thus removing $t,\ z,\ \lambda$ respectively. Light field images represent the light intensity in multiple directions for each point in a plane, as opposed to pinhole cameras, where different light rays reflected by each point in the object space are captured in a single pixel location in the image space [14].

Though usually represented by the 4D sample, light fields can also be viewed as a collection of 2D viewpoint images, with projection centers slightly deviated from each other. Thus, standard plenoptic cameras (SPC) can be seen as a camera array [34], where each camera corresponds to a viewpoint that captures a different image. An example of camera array is presented in Fig. 2.2(b) alongside a 3D reconstruction of the captured scene. Notice in Fig. 2.2(b) the distance between the projection center of each camera, represented in blue, is augmented 50 times to be clear in the image.

Because of their approximately continuous baseline, SPCs allow the computation of disparities as gradients of Epipolar Plane Images (EPIs). Using the redundancy of information in light fields opens the possibility of single image depth estimation [10, 22] provided the gradients on the image are not zero.

The restriction on spacing between viewpoints limits the field of view and, consequently, the 3D reconstruction. However, in the proposed application this should not present any issues since the face can be well captured from frontal perspectives.

2.2.1 Back-projection Model

Light fields are typically defined in the so called object space. In the object space the light field is parametrized using two parallel planes, Ω and Γ, perpendicular to z and with the reference frame of the camera at their center (see Fig. 2.3). This results in a pair of coordinates (s, t), defined in parametrization plane Γ, that identify the point where the light ray intersects the plane. And another pair (u, v) which expresses a direction, given by the intersection of the light ray with the plane Ω at unitary distance from Γ.

The notation for this light field is, as proposed by Dansereau et al. [14], the following:

$$L_{obj} : (s,\ t,\ u,\ v) \in \mathbb{R}^4 \rightarrow I \tag{2.1}$$

where I can be in $\mathbb{R}$ or $\mathbb{R}^3$ depending on whether it is a gray scale image or an RGB image. The relation between

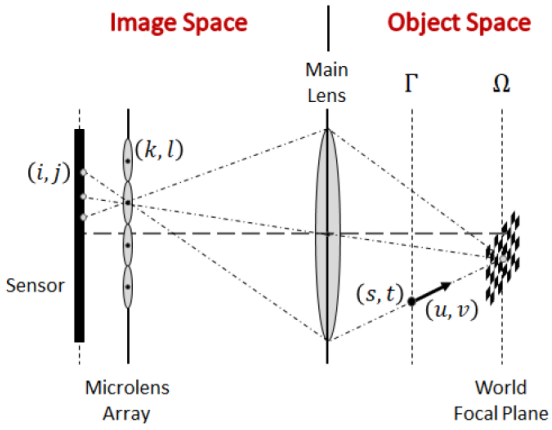


Figure 2.3: Geometry of a standard plenoptic camera, with parametrization spaces marked. Extracted from [34].

a world point $[x,\ y,\ z]^T$ and the light field is given by:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} s \\ t \\ 0 \end{bmatrix} + \lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \lambda \in \mathbb{R} \quad . \tag{2.2}$$

Light fields can be also represented in the so called image space (see Fig. 2.3). This parametrization is intrinsically related to the camera. The light field in image space is parametrized by two pairs of coordinates: the pair (k, l) which represents the selection of a microlens and the pair (i, j) which indexes the pixel underneath the selected microlens.

Since the capture occurs in the image space and the metric information is present in the object space, there is a need for conversion between both spaces. The transformation of light field coordinates is a back-projection, as proposed by Dansereau et al. in [14]. Here we will follow the notation of Marto et al.[30], for simplicity, which is:

$$\Psi = \mathbf{H}\Phi \Leftrightarrow \begin{bmatrix} s \\ t \\ u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} h_{si} & 0 & h_{sk} & 0 & h_s \\ 0 & h_{tj} & 0 & h_{tl} & h_t \\ h_{ui} & 0 & h_{sk} & 0 & h_u \\ 0 & h_{vj} & 0 & h_{vj} & h_v \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i \\ j \\ k \\ l \\ 1 \end{bmatrix} \tag{2.3}$$

where the 5 by 5 matrix is the intrinsic parameters matrix $\mathbf{H}$, Ψ denotes the light field in the object space and Φ denotes the light field in the image space.

2.2.2 Camera Calibration

Camera calibration enables the estimation of the parameters in Eq. 2.3. In order to match the equation to a real plenoptic camera, one needs to transform the raw sensor data into a light field. The process of transforming raw data to a light field in the image space is known as decoding.

Dansereau et al. [14] proposed a plenoptic camera model with a 5×5 intrinsic matrix, mapping 4D light fields,

and a method for decoding a camera's 2D microlens array images into 4D light fields without prior knowledge of its physical parameters.

Decoding

Decoding is the process of converting raw 2D microlens array images into 4D light fields. Linear demosaicing applied directly to raw 2D lenselet images yields undesired effects on pixels near lenselet edges. Therefore pixels near the frontiers of the fields of view of microlenses are ignored.

Regarding the microlens array, its exact placement is unknown with respect to the sensor (CMOS), the spacing of the microlenses images is a non-integer multiple of pixel pitch and the microlenses grid is hexagonally packed. To locate microlenses' centers, an image taken through a white diffuser is employed. The existance of vignetting at each microlens implies that the brightest spot approximates the microlens center.

The first step of decoding is to demosaic the raw microlens array image. Then, vignetting is corrected based on the white image. Afterwards the image is aligned, by resampling, rotating and scaling so that all lenselet centers fall on pixel centers.

Then, the light field is broken into identically sized, overlapping rectangles centered on the lenselet images. Followed by an interpolation in k to compensate for the hexagonal grid effects.

Lastly, correction for the rectangular pixels in (i, j) is applied, through 1D interpolation along i and masking of pixels that fall out of the hexagonal lenselet image. This process creates a virtual light field camera with its own parameters, and results in an "aligned" light field.

Projection through the lenselets

In section 2.2.1 the pinhole and thin lens model was presented assuming prior knowledge of the microlens associated with each pixel, however a pixel is not necessarily associated with its nearest microlens.

Correction of the physical camera parameters is required to cope with the virtual camera created previously. Due to the construction of plenoptic cameras, projection through the microlens array is well-approximated by a single scaling factor, similarly scaling and hexagonal sampling can be modeled as scaling factors. The lens distortion is also considered, even though that in a simple model of directionally dependent radial distortion.

Calibration and Rectification

The plenoptic camera gathers enough information to perform calibration from unstructured and unknown environments. However, a more conventional approach of camera calibration is used to start, resorting to calibration patterns with known feature locations (such as checkerboard patterns).

A single feature will appear in the imaging plane multiple times due to the very construction of plenoptic cameras. A meaningful way of finding the closest distance between each observation and the set of expected features is required. One way to do so is to generate a projected ray from each observation and use the point to ray distance as measurement (ray re-projection error). In order to get feature locations conventional methods are applied to the decoded light field. It is concluded that of the 12 non-zero terms in the model presented by Dansereau [14], 2 are redundant with camera pose, h_s and h_t, leaving 10 free intrinsic parameters.

In the initialization process the light field is considered on NxN array of viewpoint (2D) images, simply called viewpoints. The viewpoints are passed through a conventional camera calibration which yields a pose estimation

per image. Intrinsic parameters are also estimated in this process. The initial pose and intrinsic estimations formed are then used as starting point for an optimization process.

Once the camera calibration is found a rectification process can be performed. This process aims at the reversal of lens distortion and yielding square pixels in the light field imagery. The rectifications consists of an interpolation from the decoded light field such that it approximates a distortion-free light field.

2.2.3 Light Field Reconstruction

After the processing of the acquired images, when a light field is obtained, reconstruction can be considered. This section is dedicated to detailing the general approach to light field reconstruction.

When we have a camera array, like the one in Fig. 2.4(a), we can select a line of viewpoints (for instance the one depicted in red). From those viewpoints the two at the edges are unusable, since the images tend to be too dark.

Taking the viewpoint images and stacking them together results in a 4D image volume, which is a hypercube, as represented in the bottom half of Fig. 2.4(b). The top half was sliced off, with exception of one image, and the difference from the first to the last image is visible. This horizontal slice of the hypercube, highlighted in Fig. 2.4(b) and zoomed in on Fig. 2.4(c), is an example of an Epipolar Plane Image (EPI).

EPIs show the effect of parallax [8] which is the difference in the apparent position of an object regarding a background when viewed from different positions. It is visible that the lines corresponding to different points have different slopes, specifically, the cube that is closer has greater slope than the checker pattern that is in the back. This variation in the pixel position of a world feature relative to the variation in the camera considered is called disparity.

Taking advantage of the fact that a single feature in a scene has multiple projections, one for each viewpoint, Marto el al. [30] states that it is possible to use $\mathbf{H}$ to find a constraint that a collection of rays, corresponding to the same feature, must follow. The relation between space and pixel indexes is given by

$$\begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{H}_{ij}^{st} \begin{bmatrix} i \\ j \end{bmatrix} + \mathbf{H}_{kl}^{st} \begin{bmatrix} k \\ l \end{bmatrix} + \mathbf{h}_{st} + z \left(\mathbf{H}_{ij}^{uv} \begin{bmatrix} i \\ j \end{bmatrix} + \mathbf{H}_{kl}^{uv} \begin{bmatrix} k \\ l \end{bmatrix} + \mathbf{h}_{uv} \right) \tag{2.4}$$

where $\mathbf{H}_{(\cdot)}^{(\cdot)}$ denotes a sub-matrix of $\mathbf{H}$. More precisely, the intrinsic matrix $\mathbf{H}$ is partitioned in four 2×2 diagonal sub-matrices and two 2×1 vectors:

$$\begin{aligned} \mathbf{H}_{ij}^{st} &= \begin{bmatrix} h_{si} & 0 \\ 0 & h_{tj} \end{bmatrix}, \ \mathbf{H}_{kl}^{st} = \begin{bmatrix} h_{sk} & 0 \\ 0 & h_{tl} \end{bmatrix}, \ \mathbf{h}_{st} = \begin{bmatrix} h_s \\ h_t \end{bmatrix}, \\ \mathbf{H}_{ij}^{uv} &= \begin{bmatrix} h_{ui} & 0 \\ 0 & h_{vj} \end{bmatrix}, \ \mathbf{H}_{kl}^{uv} = \begin{bmatrix} h_{uk} & 0 \\ 0 & h_{vl} \end{bmatrix}, \ \mathbf{h}_{uv} = \begin{bmatrix} h_u \\ h_v \end{bmatrix}. \end{aligned} \tag{2.5}$$

Note that in Eq. 2.4 previous knowledge of the z coordinate is implied. Assuming a constant position for a feature and taking its derivative, we get a relation between the depth and the disparity:

$$z = -\frac{h_{si} + h_{sk}\frac{\partial k}{\partial i}}{h_{ui} + h_{uk}\frac{\partial k}{\partial i}} \quad \vee \quad z = -\frac{h_{tj} + h_{tl}\frac{\partial l}{\partial j}}{h_{vj} + h_{vl}\frac{\partial l}{\partial j}} \tag{2.6}$$

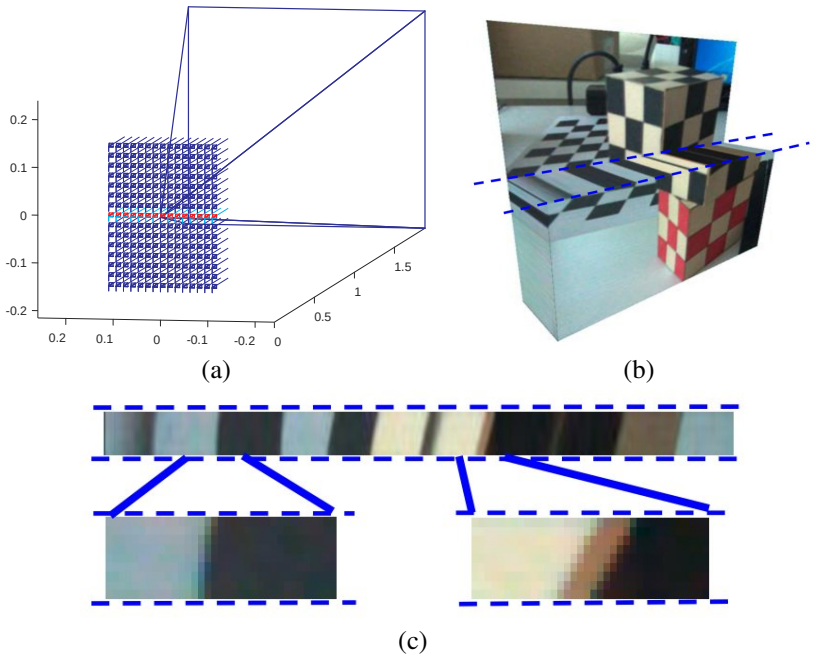


Figure 2.4: Structure from EPI reconstruction. In (a) a camera array, showing view field of central camera, distances between projection centers augmented 50 times. In (b) a stack of viewpoint images, all sliced at the middle line except the last image. In (c) the Epipolar Plane Image (EPI) highlighted in (b), detailing a background feature (on the left) and a foreground feature (on the right). Extracted from [30].

where the disparity is represented by the gradients $\frac{\partial k}{\partial i}$ and $\frac{\partial l}{\partial j}$.

Thus, we have a method to determine $[x\ y\ z]^T$, by estimating the disparity, computing the depth z, and finally use z on 2.4 to determine x and y.

2.3 Affine Light Fields

Affine light fields are a first order approximation of light field images, obtained by constraining the image gradients not to be null, and yielding a representation that contains depth information [31]. Affine light fields are categorized as being locally or globally affine, depending on the extent of the affine definition. Globally affine light fields require a very specific setup and environment, being more of a theoretical example with very little real scenarios associated [40].

The model presented in Eq. 2.3 can be simplified to a more familiar form under some simple assumptions [31]. The first assumption is that the parameters referring to the horizontal and vertical coordinates are equal. In fact the hexagonal microlens array structure is re-sampled during the decoding process, and this can be done in a square lattice. The second assumption is that the parametrization plane Π is moved along z such that the viewpoints' center of projection is contained on Π.

Consequently, $h_{sk} = h_{tl} = 0$ with the translation along z being compensated in the extrinsic parameters. Furthermore, $h_{ui} = h_{vj}$ can be set to zero since they describe a shift in the principal point of each viewpoint image which can easily be removed. Finally, the elements of the last column are dependent parameters, constrained by Ψ_{center} mapping to Φ_{center}. The combination of this assumptions results in:

$$\mathbf{H} = \begin{bmatrix} b & 0 & 0 & 0 & s_0 \\ 0 & b & 0 & 0 & t_o \\ 0 & 0 & f^{-1} & 0 & -c_x/f \\ 0 & 0 & 0 & f^{-1} & -c_y/f \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.7}$$

where b denotes the baseline (distance between adjacent cameras) and f denotes the focal length. More details on the intrinsics matrix applied to the camera array can be found in [30].

2.3.1 Setup

In general constant light fields do not provide depth information, however [31] proved that affine light fields do provide depth information directly from the affine parameters. The most straightforward scene producing a smooth (globally) affine light field consists in a fronto-parallel plane Π, with normal $\mathbf{n} = (0, 0, 1)$, colored with a constant gradient (see Fig. 2.5) [31].

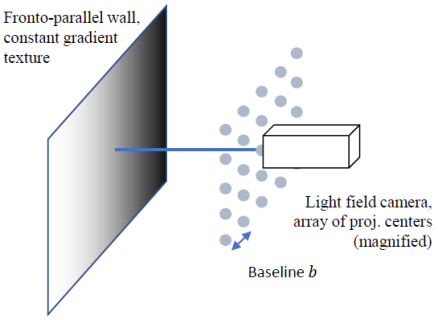


Figure 2.5: Fronto-parallel plane colored with a gradient in the setup to acquire a globally affine light field. Extracted from [31].

The color of a given point $p \in \Pi$ is given by $c(\mathbf{p}) = \mathbf{p} \cdot \mathbf{g} + c_0$, where $\mathbf{g}$ is the color gradient vector. To determine the color sampled by a ray Ψ one has to find where it intersects the plane Π using the back projection model:

$$\mathbf{p} = [s\ t\ 0]^T + \lambda[u\ v\ 1]^T \tag{2.8}$$

where λ denotes the extension of the ray before hitting Π, i.e., the depth z of the plane ($z = \lambda$). Therefore an expression for z is:

$$z = \frac{r - (s, t, 0) \cdot \mathbf{n}}{(u, v, 1) \cdot \mathbf{n}}. \tag{2.9}$$

The expression in Eq. 2.9 combined with the camera parametrization yields the affine light field [40], defined as follows:

Definition 2.3.1. *(Locally affine light field) A light field is denoted locally affine at $\Phi = (i, j, k, l)$ if it is affine with respect to all variables*

$$L(i, j, k, l) = l_0 + [a_i\ a_j\ a_k\ a_l] \cdot [i\ j\ k\ l]^T \tag{2.10}$$

within a neighborhood of (i, j, k, l).

There are now familiar terms since: $a_i = bg_x$, $a_j = bg_y$, $a_k = zg_x/f$, $a_l = zg_y/f$ and l_0 collects the constant terms.

2.3.2 Depth Computation

Locally affine light fields provide directly depth information, being the minimal order case that enables depth reconstruction [31].

A common way of obtaining a locally affine light field is through a first order Taylor series expansion, which allows us to consider many real world light fields [40], since we only require these characteristics in a patch. The expansion results in Eq. 2.10 [40], where the gradient of L, $\nabla L = [a_i\ a_j\ a_k\ a_l]^T$, contains the depth z in the (k, l) derivatives.

The remaining unknown parameters are g_x and g_y present in both (i, k) and (j, l) respectively. These can be canceled out using a quotient, resulting in the following depth estimates:

$$z = bf\frac{a_k}{a_i} \quad and/or \quad z = bf\frac{a_l}{a_j}. \tag{2.11}$$

We now possess a formula to calculate z as a function of: the gradient of the light field ∇L, the baseline b and the focal length f. Comparing Eq. 2.11 with stereo reconstruction, we find that a_k/a_i and a_l/a_j play the role of disparities.

In order to use Eq. 2.11 to extract depth in a real scene, an estimation of $a_{(\cdot)}$ is required. To do so a locally affine approximation has to be performed. Two approaches are proposed in [31] to perform this approximation. Estimating the gradients in the EPIs with Sobel operators as in [13]. Or using structure tensors, which involves estimates in the four components of the light field, plus low pass filtering in the four dimensions to mitigate high frequency noise enhanced by the derivatives, as in [52]. The latter was chosen in [31].

Regarding application purposes, the depth estimation requires the gradient of the light field, specifically the gradients on viewpoint images, to be non zero. Thus, for constant areas (with zero gradient) this method will fail.

2.4 Reconstruction Methodologies

In this section we dive deeper in the reconstruction methods. We start with the introduction of a Naïve depth estimation algorithm. Then, we review the workings of the state of the art methods for reconstruction. Lastly, we present an in depth analysis of two methods: the spinning parallelogram operator and the superpixel segmentation which can be used for depth estimation.

2.4.1 Gradient Based Depth Reconstruction

In [30] Marto et al. proposed a method to perform depth reconstruction. The method receives a light field in the image space as input and produces a depth map as output.

In order to produce the input for the method, the raw images have to be decoded and the calibration of the camera performed beforehand, following the procedure described in 2.2.2.

The first step consists of disparity estimation using epipolar plane images (EPIs). To do so, the gradient of the image, $\frac{\partial k}{\partial i}$ or $\frac{\partial l}{\partial j}$, is extracted from the EPI. The direction orthogonal to this, is, by definition, the direction of least change in the image, which, in most cases, corresponds to the direction where all pixels belong to the same real world feature.

Edges in the EPIs correspond to the areas of the image with large gradients, allowing useful epipolar gradient information to be extracted. To calculate the gradients in the EPI structure tensors are used. These require gradients of the image along i and k, I_i and I_k respectively, which can be calculated using a Sobel operator. Thus, defining the local structure tensor, S_0, for each pixel as:

$$S_0 = \begin{bmatrix} I_i^2 & I_i I_k \\ I_i I_k & I_k^2 \end{bmatrix} \quad . \tag{2.12}$$

The goal is to estimate the structure tensor, $S(k, l)$, for every viewpoint pixel (k, l), yielding a 2D array of structure tensors indexed by viewpoint pixel. Averaging S_0, associated with every EPI, along i reduces the dimension to a 1D array of structure tensors, S_e. Now, S_e is calculated for every possible EPI, vertical and horizontal, and on all color channels. The vectors $S_e^{(\cdot)}$ will then be added, on the location they refer to:

$$S(k, l) = \sum_i \sum_j S_e^{jl}(k) + S_e^{ik}(l) \tag{2.13}$$

resulting in $S(k, l)$, with a structure tensor for every viewpoint pixel index. Since this method will give preference to stronger gradients, areas with noise dominated gradient will not disturb the final results.

The structure tensor provides more than the gradient direction across the EPI. From the difference of its eigenvalues it is possible to extract a confidence measure on how accurate is the disparity estimation in any given location [6].

In a second step, after the initial disparity estimation, low confidence estimates are disregarded, noise is handled and regularization is performed to obtain data pertaining the whole viewpoint area.

The regularization consists in finding the values b that minimize a function of the form $E(a, b) + \lambda J(b)$, where λ is the regularization parameter. As defined by Chambolle in [11]:

$$E(a, b) = ||b - a||^2/2 \tag{2.14}$$

and it measures the difference between the original image a, and b. The total variation of b is given by:

$$J(b) = ||\nabla b||. \tag{2.15}$$

The third and final step is to obtain the depth map from the regularized dense disparity map. It requires the calculation of $[x\ y\ z]^T$ for all (k, l) and for a fixed (i, j), as detailed in 2.2.3.

2.4.2 Reconstruction Fundaments

In 2017 Johannsen et al. published a taxonomy and evaluation of light field reconstruction methods [25]. The goal was to compare strengths and weaknesses of different approaches, and to correlate which components lead to good results in a given region. The assumption of a Lambertian scene is the cornerstone for all disparity estimation

algorithms discussed. In such a scene, by definition, all rays originated from a 3D point should share the same radiance.

Light fields have multiple possible representations, but the main representations are: subaperture views (viewpoint images), EPIs, Surface Cameras (SCams) and Focal Stacks. Mutli-view stereo methods use the subaperture views, relying on patch comparison to find the best correspondence among the images for a set of disparities [25]. Methods that rely on EPIs are also frequent, since 3D points are projected onto lines in the EPIs the depth estimation is reduced to orientation analysis of said lines. Angular patches or Scams sample the radiance for all corresponding projections of a scene point at the respective depth [57] and can be leveraged to analyze occlusions. The focal stack is composed by a set of refocused images, each at a given depth z, which is no more than integrating over the angular patches at that depth [25].

Depth estimation can be achieved using any of the proposed representations. Generally, light field depth estimation algorithms follow a common pipeline which consists in three main steps:

1. Information selection;

2. First reconstruction;

3. Refinement of the initial estimations.

The first step comprises the choice of the light field representation(s) to be used as well as the selection of the views to be considered, e.g. a crosshair or the full set of views. The result of this first step is a cost volume.

The second step performs disparity estimation via global optimization of the cost volume. Common methods are Markov Random Fields (MRF) and graph cut approaches or variations of these with regularization.

The refinement stage aims at filling in the missing information of the initial estimation. Usually consists in local filtering (weighted median or bilateral filters) or global regularization of the disparity map.

In the next section one of the algorithms presented in [25] is further analyzed. This algorithm is the spinning parallelogram operator [58] which uses the EPI representation in a crosshair of views, yielding two epipolar plane images to analyze. A cost volume is built for a discrete set of disparities by comparing the regions on either side of the epipolar line. The initial estimation is obtained via winner takes it all strategy and in the refinement step a guided filter is applied on the cost volume.

2.4.3 Spinning Parallelogram Operator

The Spinning Parallelogram Operator (SPO), proposed by Zhang et al. in [58], addresses the problems caused by occlusion and noise in light field depth estimation. Using a crosshair of views the method is used to locate lines in the Epipolar Plane Images (EPI) and to estimate their orientation. Furthermore, local and global confidence measures are calculated to handle occlusion, followed by filter-based in-painting to cover texture-less regions.

The mechanism behind this method splits the EPI into two regions - slightly to the left and right of the line in question - and computes histograms for both. Through histogram comparison a cost function over different disparities is attained for each EPI. The distance between the distributions of color is measured using the χ^2 difference of the histograms:

$$\chi^2(g_\theta, h_\theta) = \sum_i \frac{(g_\theta(i) - h_\theta(i))^2}{g_\theta(i) + h_\theta(i)} \tag{2.16}$$

where $g_\theta(i)$ and $h_\theta(i)$ are the histograms of the separated parts. The derivative of a Gaussian weighs the contribution of each pixel based on the distance from the point to the center line.

After the calculation of the histogram distance, in the horizontal and vertical EPIs, a confidence metric is defined. Taking the difference between the maximum and average scores, results in low confidence for ambiguous and occlusion zones.

The cost volumes for both EPIs are then combined according to the confidence metric, through a weighted summation:

$$d_{u,v}(x,y,\theta) = c_{y,v^*}(x,u^*)d_{y,v^*}(x,u^*,\theta) + c_{x,u^*}(y,v^*)d_{x,v^*}(y,v^*,\theta) \tag{2.17}$$

where $c_{.,.}$ denotes the confidence metric in the specified EPI.

The resulting cost volume is then regularized for each individual depth label, with the correct information being propagated to similar regions with low texture using a filter-based method.

Lastly, a disparity map is generated using a winner-takes-all strategy. The line's orientation is determined through the maximization of the distance between the histograms of pixel intensity. Large differences between the histograms indicate the presence of an edge dividing the regions. Thus, the maximum orientation response is taken:

$$\Theta_{y,v}(x,u) = \arg\max_{\theta}\ d_{y,v}(x,u,\theta) \tag{2.18}$$

where $d_{y,v}(x,u,\theta)$ is the histogram distance measured by the SPO on the EPI $I_{y,v}(x,u)$ (analogous for the vertical EPI).

For synthetic images this process approximates global results, resorting to an edge preserving smoothing filter - the guided filter. However, in real images further optimization is required, therefore a weighted median filter, graph cuts and iterative refinement are used.

This methodology outperforms the traditional feature matching techniques in the presence of ambiguity or occlusions because it retains the correct depth information. Three factors contribute to this: occlusions create joints in the EPI, where the correct line is intersected by the occluding line; the distance between histograms remains at a local maximum; and finally, uses the surrounding area instead of a single point. Specifically, the SPO sets local and global confidence and chooses information for individual views during the depth scores calculation to deal with occlusions. Furthermore, the method is also very robust to noise and artifacts [58].

Using θ to define the direction of the, the corresponding local depth estimations can be obtained by:

$$z = f\frac{\Delta u}{\Delta v} = \frac{f}{tan\theta} \tag{2.19}$$

according to [51].

2.4.4 Light Field Superpixel Segmentation

A formal definition of light field superpixel (LFSP) is a set of all light rays radiated from a proximate, continuous and similar 3D surface [59]. Let R be a proximate, similar and continuous 3D surface and $L(u,v,x,y)$ the recorded light field, the LFSP $s_R(u,v,x,y)$ is defined as:

$$s_R(u,v,x,y) = \bigcup_{i=1}^{|R|} L(uP_i, vP_i, xP_i, yP_i) \tag{2.20}$$

where P_i denotes the i-th point of the surface R and $|\cdot|$ denotes the number of elements in the set [59].

The application of superpixel segmentation in light fields aims to simplify their processing by grouping similar pixels among all views in a consistent manner [28]. Ideally superpixels would be accurate, compact, efficient and view consistent, however, a compromise has to be made. In [28] the priority lies with accurate and view-consistent superpixel segmentation while explicitly handling occlusion and implicitly computing per view disparities.

The method starts with a robust detection of lines in the central EPIs (horizontal and vertical), using directional filters and line fitting to handle occlusion cases. Then, it enforces view consistency in an occlusion-aware way through explicit line estimation and bipartite graph matching [28], which pairs the lines into regions using depth ordering. This angular segmentation in the EPIs is clustered in the last step, where the estimated disparity is used to regularize the process. In the end, the unlabeled pixels remaining are labeled via a propagation step resulting in a view-consistent superpixel segmentation.

The light field superpixel segmentation method implicitly produces a per view disparity map, therefore it is possible to recover the depth information using the relation between depth and disparity presented in Eq. 2.6.

Chapter 3

Patch Based Reconstruction

Up until now we have defined a model for the plenoptic camera, which enables us to get light field imagery. Furthermore, we studied how to recover metric information from light fields and analyzed two methods for that purpose.

This chapter proposes a reconstruction methodology that complements the weaknesses of edge based reconstruction algorithms. Specifically, a method to retrieve depth information from low gradient areas is proposed.

We first introduce the concept of light field shearing, which is the fundamental operation for the proposed method. Then, through a series of experiments we refine the idea of reconstruction using patches of locally planar surfaces. In the end, we summarize the findings of the experience with the proposal of a depth reconstruction algorithm.

3.1 Light Field Shearing

The process of shearing a light field is equivalent to changing the world plane in focus, as shown by Ren Ng in [43]. As a consequence of this operation, all objects in that plane have zero disparity. Thus, the objects appear in the same position in all viewpoint images.

This process can also be referred to as refocusing, as done by Tao et al. [48], but Ren Ng refers to refocusing as a method that produces a conventional refocused image. However, in Ren Ng [43] there is a need for integration in i and j to produce a 2D refocused image. Nonetheless, in both cases the shearing operation is performed in the same way.

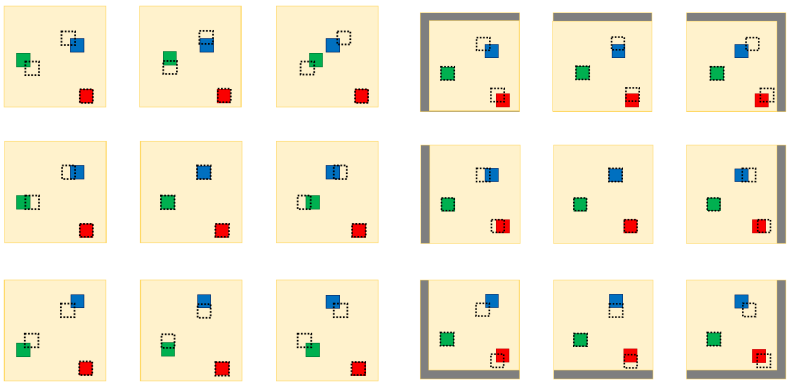
In practice, this process consists in a translation of each viewpoint's position by an amount α proportional to its distance to the central viewpoint, as shown in equation 3.1.

$$L_\alpha(i,\ j,\ k_\alpha,\ l_\alpha) = L(i,\ j,\ k_\alpha + \alpha(i - i_{center}),\ l_\alpha + \alpha(j - j_{center})) \tag{3.1}$$

For features at a given disparity to become in focus, their disparity after shearing must be zero. Thus, the amount α by which the light field is sheared corresponds to the features' disparity.

The translation of the viewpoint images, inwards or outwards, causes the appearance of undefined regions in the edges of the viewpoint images (see Fig. 3.1). This happens because we are translating the viewpoint to an area that was not captured in the original light field.

An example is presented in Fig. 3.1, where we see a red feature in focus in the original light field (Fig. 3.1(a)). The squares in dashed line denote the features' positions in the central viewpoint. The green feature, in a plane further away from the camera than the red feature, will be focused. In Fig. 3.1(b) we see the green feature in focus, i.e. in the same position in all viewpoints. Gray borders appear on the outer side of the viewpoint images of the sheared light field. Conversely, if we were to focus the blue feature the borders would appear towards the center side of the viewpoint images, since it is contained in a plane closer to the camera.



(a) Original light field (b) Sheared light field

Figure 3.1: Viewpoints of a light field, containing 3 features, before and after shearing. The dashed squares show the positions on the central viewpoint. In (a) we have the original light field with a red square in focus, a green square with positive disparity and a blue square with negative disparity. In (b) we have a sheared light field focusing on the green feature's plane which is farther than the red feature's plane.

Having a sheared light field, given by Eq. 3.1, one can obtain the refocused image, at disparity α, by averaging the sum of all the viewpoints of the sheared light field:

$$I_\alpha(k,\, l) = \frac{1}{N} \sum_i \sum_j L_\alpha(i,\, j,\, k_\alpha,\, l_\alpha) \tag{3.2}$$

where N is the number of viewpoints composing the array.

3.2 Light Field of a Locally Planar Surface

In this section we conduct a study to assess if reliable depth estimation can be achieved on smooth light field areas. The proposed approach uses the shearing operation on a local scale rather than relying on edge points. This study considers only light fields of planar Lambertian objects illuminated by a single point light source. The choice of a point light source is related to the need to obtain a smooth texture that still has a small level of gradient in it. The experiments start with the simplest case and progress into increasingly complex cases to allow a natural follow-up.

3.2.1 Lambertian Surface

The application of the shearing operation to planar Lambertian surfaces is the cornerstone for the proposed experiments. In such case, two light fields are considered: (i) the light field of a locally planar Lambertian surface $L(\cdot)$,

and (ii) a virtual light field $L_0(\cdot)$, representing a textured plane at the focused distance, which has the same central viewpoint as $L(\cdot)$.

Since the virtual object is at the focused plane, all viewpoints of $L_0(\cdot)$ are equal. Thus, shearing $L_0(\cdot)$, at a local (i, j), according to the real depth of $L(\cdot)$ will make the two light fields locally equal (see Definition 3.2.1).

Definition 3.2.1 (Light field of a (virtual) planar object placed at the focused plane (LFFP))**.** The light field of a planar object placed at the focused plane is the central viewpoint replicated at all viewpoints. In other words, it is an imaging texture assumed to be at a constant depth which is the focused one.

The locally planar hypothesis, specifically the consideration of an area (neighborhood) around a point is relevant in the creation of a reconstruction algorithm. Under the assumption that local texture provides enough information for depth estimation, considering a local area for the shearing handles occlusion issues (which appear at depth discontinuities). Depth reconstruction is achieved through the registration of the local texture.

While this property is true for all light fields of Lambertian locally planar surfaces, for some light fields it is not as interesting. For instance, in locally affine light fields the property is verified but it does not provide a reconstruction algorithm. For this reason we choose a punctual light source, as shown in Fig. 3.2, as a light source at infinity would make an affine light field.

It is important to note that the created virtual object, defining $L_0(\cdot)$, is a convenience for creating an intuitive property and demonstrating it. In practice, a reconstruction algorithm can be built directly on top of $L(\cdot)$.

The experiment's procedure is now described. First, the shearing operation, as described in 3.1, is applied to change the focused depth plane. A focused object should appear in the same position in every viewpoint due to its zero disparity (see Definition 3.2.1). Therefore, in the light field of a focused fronto-parallel plane we should see that all viewpoint images are equal.

Then, the viewpoint images of the sheared light field are compared among themselves to verify this hypothesis. To do so, we define an average similarity error in Eq. 3.3 by comparing each viewpoint image against the central viewpoint image.

Let L be a light field with $N \times N$ viewpoints, each with size $W \times L$, the average similarity error, C_S, is computed as:

$$C_S = \frac{1}{N^2 - 1} \sum_{k=1}^{N} \sum_{l=1}^{N} \sum_{i=1}^{W} \sum_{j=1}^{L} \big(L(i, j, c, c) - L(i, j, k, l)\big)^2 \tag{3.3}$$

where c is the index of the central viewpoint.

To assess the effect of using the error metrics as a means for depth estimation, a deeper study on their calculation is performed. The impact of the size and positioning of the area considered in the calculation, i.e. the considered window, is studied. Specifically, the size is studied using a centered window with variable size and the position using a grid by comparing each grid unit individually.

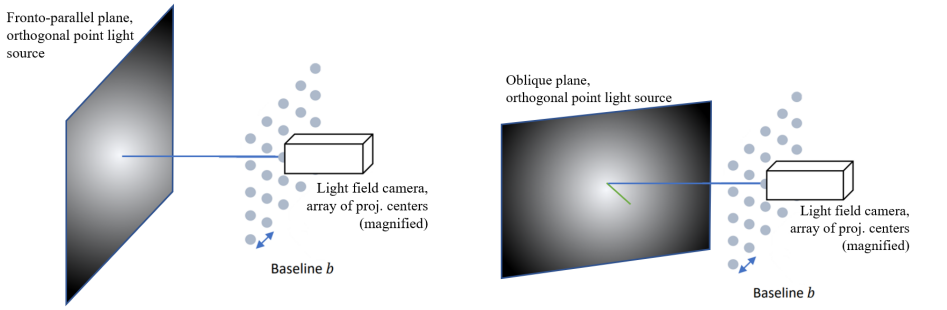
Lastly, the extension to locally planar patches is done resorting to an inclined plane, where the same study is performed. It is relevant to note that the local plane can have a general orientation, except an orientation orthogonal to the imaging plane. The shearing yielded by the exact registration provides information on the local plane's central depth and slant (sloping direction).

3.2.2 Shearing Study Setup

This experiment uses synthetic data created using Virtual Reality Markup Language (VRML) [12]. A scene containing a textured plane serves as basis for the two experiment setups. The plane's texture emulates point light source illumination, which possesses small but non zero gradient information, as shown in Fig. 3.2.

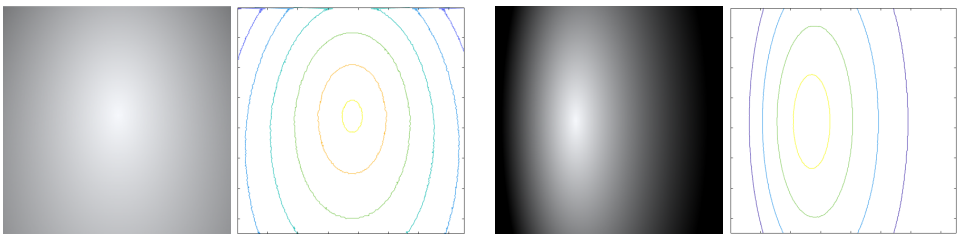
Given the presented synthetic VRML scene, Matlab's Virtual Reality toolbox is used to acquire the light fields. For the first acquisition we place a camera such that the plane is fronto-parallel as shown in Fig. 3.2 (a) and at a distance of 0.3 units. Then, the camera is moved around the vertical axis such that the plane's right side is closer to the camera, as shown in Fig. 3.2 (b).

For each of the presented setups, an array of images is captured and arranged into a light field. An example of the central viewpoint image for both light fields is shown in Figs. 3.2 (c) and (d). To further grasp the idea of a smooth image that still possesses gradient information, we show alongside each viewpoint image the corresponding level curves.



(a) First setup

(b) Second setup



(c) Central viewpoint and level curves in LF from (a)

(d) Central viewpoint and level curves in LF from (b)

Figure 3.2: Proposed experiment setups: (a) Fronto-parallel plane with texture emulating point light source illumination; (b) Oblique plane with the same texture (the plane's normal is highlighted in green). Central viewpoint images from the acquired light fields and respective level curves: (c) for the first setup of a fronto-parallel plane (d) for the second setup of an oblique plane.

The light fields are captured in a grid of 11 by 11 viewpoints, forming an array, and the distances between projection centers are equal in both directions, horizontal and vertical. The intrinsic matrix estimated using J. Y. Bouguet's calibration toolbox [9] is presented in Eq. 3.4.

$$\mathbf{H} = \begin{bmatrix} 2.795e\text{-}4 & 0 & 0 & 0 & -0.0014 \\ 0 & 2.876e\text{-}4 & 0 & 0 & -0.0015 \\ -0.0010 & 0 & 0.0018 & 0 & -0.3492 \\ 0 & -0.0010 & 0 & 0.0018 & -0.3479 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

After the light field acquisition, we apply shearing multiple times, at different disparities, to the original light field. This results in a series of sheared light fields refocused at different depths.

Since we expect all viewpoints to be equal when the fronto-parallel plane is in focus, have to compare them and search for the most similar. The method chosen to measure similarity is the difference between the central viewpoint image and every other viewpoint image. Thus, we calculate the Sum of Squared Differences (SSD) and the Sum of Absolute Differences (SAD), both of which normalized to yield a per pixel value.

There are N viewpoints meaning that there will be $N - 1$ measurements of each metric, so the average of the error is taken to represent each shearing, as defined in Eq. 3.3 for the SSD. The average error is expected to have a minimum around the depth in which the plane is placed, according to Definition 3.2.1. An additional metric can be extracted from the SSD by taking the square root, yielding a measurement for the Average Brightness Error (ABE).

3.2.3 Full Light Field Shearing

We start by using the light field of the first proposed setup (see Fig. 3.2 (a)). A set of ten shearings, equally spaced in a depth ranging from 0.1 to 1 units, is applied to the full light field.

The three error metrics, described in the end of section 3.2.2, are calculated for each of the sheared light fields. The resulting error values, depicted in Fig. 3.3, reveal that every metric displays a local minimum on the shearing depth of 0.3 units.

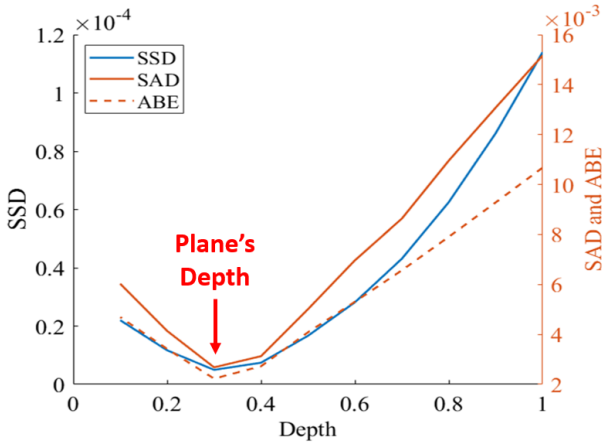


Figure 3.3: Plot of the error metrics against depth used for each shearing. On the left axis (in blue) we have the sum of squared differences (SSD). On the right axis (in orange) we have the sum of absolute differences (SAD) in full line and the average brightness error (ABE) in dashed line.

We know that the smallest error corresponds to the highest similarity. By definition, we can claim that the

Lambertian plane is placed around the depth of 0.3 units, which we know to be true (see section 3.2.2). Therefore, we confirmed that it is possible to estimate depth of globally planar surfaces through maximization of viewpoint similarity.

To establish a reference for comparison, the error metrics were also calculated for the original light field. We found that the original light filed displays a lower viewpoint similarity than the sheared light field, which is expected when the fronto-parallel plane is out of focus.

3.2.4 Local Window Shearing

After achieving a correct depth estimate using the full light field, we investigate if a portion of the image is enough to produce reliable results. To do so, another experiment is performed on the same setup. This time, an increasingly larger border of the images is discarded, resulting in a smaller centered window of the viewpoint image used when calculating the error metrics.

We start the experiment with a border of 4 pixel, which increases additional 10 pixel per iteration, until a 10 by 10 pixel window remains. The error is expected to hold in the same range up until a relatively small window, due to the synthetic nature of the data. Nonetheless, quantization errors still exist in the process of encoding of the brightness.

The 3D plot shown in Fig. 3.4 refers to the SSD value against the first five shearing depths and all the window sizes tested. Along the depth dimension we observe a valley, where the minimum values correspond to the depth 0.3 units in which the plane is located. The local minimum in this plot, marked by the red dot, coincides with shearing depth of 0.3 units and the largest window size of 370 px.

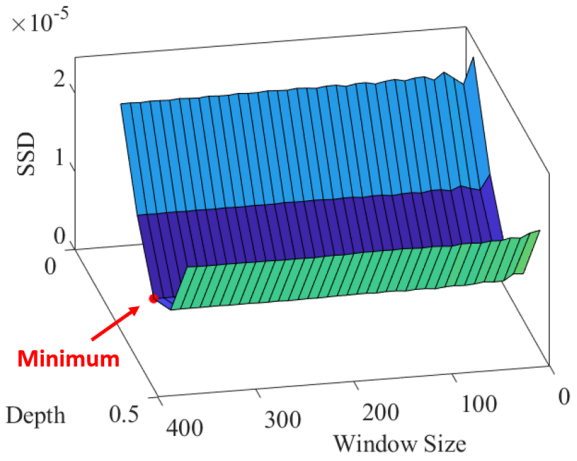


Figure 3.4: Plot of the sum of squared differences for the sheared light field against the shearing depth and the size of a centered window (the red dot denotes the minimum value of 4.7×10^{-6} units).

It is also noticeable that the error value tends to increase when the window becomes too small, regardless of the shearing depth. Hence, we sliced the graphic in Fig. 3.4 at the shearing depth of 0.3 units, obtaining a plot that relates the error value and the window size.

The slice is represented by the blue line in Fig. 3.5, additional error metrics, SAD and ABE, are plotted in orange against the right vertical axis. This figure reveals that the error stays within a limited range (highlighted in

red) until a window size of 80 px, increasing significantly for smaller window sizes. Furthermore, the SAD and ABE lines reveal the same pattern but increasing from the 60 px mark instead of 80 px.

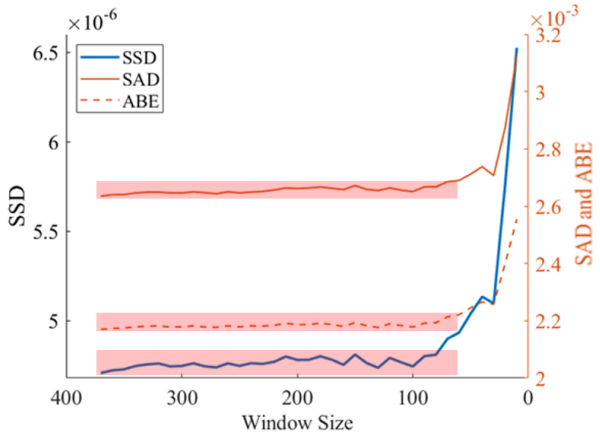


Figure 3.5: Error metrics plotted against the centered window sizes, calculated for the sheared light field that places the plane in focus. On the main axis is the SSD value (sliced from Fig. 3.4) while on the secondary axis are the SAD (full line) and ABE (dashed line). For all metrics an error envelope is highlighted in red.

3.2.5 Locally Planar Surfaces

We have proven that the viewpoints are similar for a focused fronto-parallel plane, i.e. a globally planar surface. The experiment is now extended to study the effect in locally planar patches, namely the method's performance and the impact of patch comparison. For this experiment we use the second setup, where the plane retains the texture but the camera pose rotates around the vertical axis (see Fig. 3.2 (b)). This setup results in a foreshortening effect, meaning that since the right side of the plane is closer to the camera, it occupies a larger area of the image than the left side.

The shearing procedure is the same as before, however the whole plane is no longer expected to be in focus. Nonetheless, the global shearing corresponding to the lowest depth should yield the smallest error, since objects closer to the camera occupy larger areas of the image.

After the application of successive shearings, for different disparity values, we observe that the smallest depth minimizes the error. We select this sheared light field to calculate the error metrics in a centered window, as done in the previous setup.

Since the right side of the image is mostly in focus it contributes with small errors, conversely the left side yields high errors since it is out of focus. Error starts by increasing due to the exclusion of a larger area in focus than the one out of focus, which is due to the foreshortening effect. Then, it decreases because the we discard more of high error areas than small error ones, with the minimum being attained at the best proportion of close to focus/out of focus areas. For small windows it increases again because the center is not in focus, hence viewpoint image similarity is lower. The result, depicted in Fig. 3.6, reveals that the local SSD minimum occurs for a window size of 150 px.

Furthermore, the average SSD value obtained is 1.487×10^{-4}, with a minimum value of 1.299×10^{-4}. The closeness between these values indicates that a centered window is not ideal for this setup.

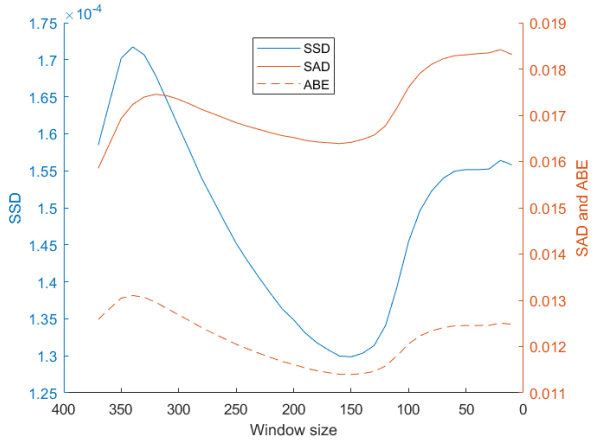


Figure 3.6: Plot of the error metrics for the sheared light field against the size of a centered window.

Given the limitations of the use of a centered window, a new iteration of this experiment is performed changing only the window used for comparison. Instead of comparing a centered window, the viewpoint images were divided in a grid and the corresponding patches were compared.

The grid patches are compared one by one against their corresponding counterparts in other viewpoint images, using exactly the same pixel coordinates. It is expected that areas closer to the camera present a significantly lower SSD value, however areas further away from the camera should present higher SSD value.

In a first attempt we use a 3 by 3 grid of 100 px patches. This results in an average SSD value of 1.605×10^{-4}, and a minimum SSD value of 7.929×10^{-5}. Notice the difference between the average and minimum values regarding the order of magnitude. We can already conclude that the position in which the error is calculated influences the result. In this case, the minimum corresponds to the region closer to the camera, which is in focus.

Then, we iteratively refine the grids up to 30 by 30 squares. The average SSD value remains close to the one obtained with the 3 by 3 grid, however, the minimum SSD value decreases by 39%.

When compared against the centered window results we see a decrease in the minimum SSD error, influenced by the regions closer to the camera. However, the average value is slightly higher due to the consideration areas further away from the camera. These areas generate higher error due to the higher depth variation within the patches. The perspective change between viewpoints also influences the error, since it is not accounted for in the calculations.

The Lambertian assumption combined with a texture of some gradient is sufficient for our purposes. Thus, we drop the single point light source hypothesis. Despite being superfluous this hypothesis provided a solid basis for the creation of the required texture.

3.3 Depth Reconstruction Algorithm

In this section we propose a naïve algorithm for depth estimation in smooth areas of the light field. The algorithm is to be used as a complement to an edge based reconstruction method, for instance gradient based method presented in section 2.4.1, filling the gaps of the low gradient regions.

The algorithm, summarized in Fig. 3.7, searches for the disparity that locally maximizes the viewpoint similar-

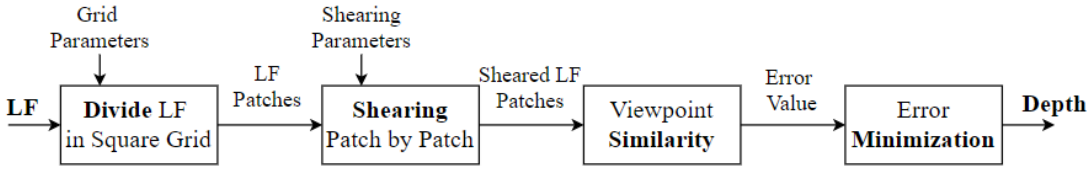


Figure 3.7: Block diagram of the shearing based depth estimation.

ity, which should correspond to the real depth value.

The first step consists in the division of the input light field in patches. It is assumed that all points in a patch have approximately the same depth value. The division follows a grid scheme and is performed in a centered window, meaning that a border of the light field can be discarded. Thus, this step has two tuning parameters: the size of the border to ignore and number of squares of the grid. This process results in a series of light field patches that serve as input for the next step.

The second step comprises the application of shearing to the light field patches. Each patch is sheared at multiple disparities resulting in several sheared light fields, one for each disparity. The two main shearing parameters are: the number of shearings per patch and the disparity range in which the shearings are performed.

The third step is the determination of the viewpoint similarity for the light field patches. In this step we assess if the shearing placed the patch in focus, or how close it is from being in focus. Since for a patch in focus the viewpoints should be equal, we calculated the sum of squared differences for each sheared patch according to Eq. 3.3.

The final step is depth assignment, performed via minimization of the average similarity error. We start by searching for the disparity that yields the smallest error value in a given patch. Then, we assign the corresponding depth to the whole patch yielding a depth map.

The assumption of regular patches having similar depth values can quickly fall in real scenarios. In such cases the first step should be adjusted to perform a more adequate segmentation, namely with a free shape instead of squares. Note that said segmentation should yield locally planar patches, or as close to it as possible. Furthermore, the independence between patches should be conditioned in a regularization step to prevent outliers.

Chapter 4

Face Reconstruction

In this chapter, we analyze three reconstruction methods previously discussed: the square patch based depth reconstruction (proposed in section 3.3), the spinning parallelogram operator (SPO) [58] (introduced in section 2.4.3) and the light field superpixel segmentation (LFSP) [59] (introduced in section 2.4.4).

Then, we propose a method that aims to complement depth estimation from edge based approaches, as [32], with depth estimation on low gradient areas. Our proposal, summarized in Fig. 4.1, detects low confidence areas and performs reconstruction on those areas to complement the information of the first reconstruction.

The novelty from the method in section 3.3 is a more accurate approach to patch segmentation, based on level sets of the reconstruction confidence. This metric gets lower as one strays from edge points, however useful information can still be extracted from these regions. The remainder of this chapter details the calculation of the reconstruction confidence from the structure tensor followed by the detection and segmentation methods.

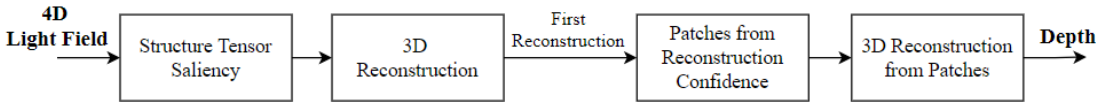


Figure 4.1: Simplified block diagram of the proposed approach for face reconstruction.

4.1 Reconstruction Methods

A depth estimation algorithm is proposed in section 3.3, which focuses on the smooth regions of the image. The idea is that depth can be estimated correctly in smooth regions resorting to the shearing of local regional patches, i.e., a local shearing.

In this section we highlight the strengths and weaknesses of the method proposed in 3.3 and compare it against the spinning parallelogram operator [58] and the superpixel segmentation [59]. Furthermore, we introduce a more refined approach to patch segmentation which will be further explored in the remaining of the chapter. These methods are summarized in a simplified block diagram presented in Fig. 4.2.

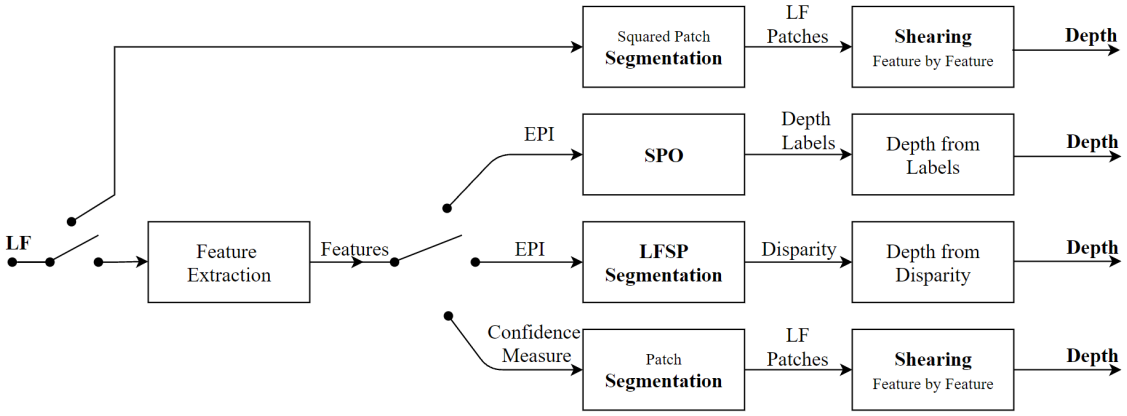


Figure 4.2: Simplified block diagram for the methods used in the experiment: square patch based depth reconstruction; spinning parallelogram operator [58], superpixel segmentation [59], level sets on reconstruction confidence.

Square Patch Based Reconstruction

Under the assumption that a human face can have approximately local planar patches, we consider the application of the algorithm proposed in section 3.3. On the top path of the block diagram in Fig. 4.2 we summarize its behavior. The method splits the light field in square patches, which are then individually sheared until the maximum viewpoint similarity is attained, i.e., the patch is in focus. Knowing the disparity that places the patch in focus we can assign the corresponding depth to the patch.

The application of this method to faces consists in a naïve approach, fundamentally designed to prove that it is possible to estimate depth, where edge based methods struggle, through local shearing. Furthermore, it bears the advantage of being simple to understand and apply.

The assumption of constant depth in grid squares is unrealistic, however it suffices for extraction of a dominant depth. Moreover, the error is not expected to be zero since patches in a real scene will not all be exactly planar and viewpoint shift not accounted for in the photo-similarity calculation process.

Spinning Parallelogram Operator

The second path, or first path after feature extraction, of the block diagram in Fig. 4.2 depicts the working of the spinning parallelogram operator (SPO) [58], which operates on epipolar plane images. The SPO estimates the orientation of epipolar lines by comparing the regions on either sides of the lines. A cost volume is built for a discrete set of disparities, which is then regularized for each depth label individually. Since this method outputs a depth labeling for each pixel, an additional step is required: the conversion from depth labels to metric depth values.

In [25] Johannsen et al. performed a thorough evaluation of the SPO, we summarize the key takeaways regarding strengths and weaknesses in the next paragraphs.

For strengths, this method has proven very robust due to the comparison of small regions with weighting. Despite not explicitly modeled, the method still handles occlusion boundaries robustly because of the maximization of histogram distance. Furthermore, it performs very well in fine structure thinning and fattening, and has a strong performance in general discontinuities. Conversely, in the matters of surface reconstruction it performs below

average. Specifically, it struggles in areas of low texture (small gradients) despite having the best tradeoff on discontinuities and fine structures. Nonetheless, the SPO is very robust to noise, artifacts and occlusion [58].

Light Field Superpixel Segmentation

The third path of the block diagram presented in Fig. 4.2 presents the usage of superpixel segmentation (LFSP) [59] for depth estimation. The LFSP is defined in the 4D space and aims at light field segmentation, with the intent to simplify their processing by grouping similar pixels among all views. The method detects lines in the central EPIs (horizontal and vertical), using directional filters and line fitting to handle occlusion cases. Then, it pairs the lines into regions using depth ordering, thus enforcing view consistency in an occlusion-aware way. This angular segmentation in the EPIs is clustered in the last step, where the estimated disparity is used to regularize the process. Lastly, unlabeled pixels are labeled via a propagation step. This method can be used for depth estimation since it implicitly creates a disparity map.

This method outperforms the original LFSP proposal on view consistency and boundary accuracy. However, since not every pixel in the light field is view consistent issues remain. Specifically, the clustering step does not explicitly handle occlusion, with only a mild influence by a high disparity weight in the regularization. Furthermore, when pixels are occluded from both sets of views the labels are propagated without occlusion awareness nor spatial smoothing. If the background and foreground share similar textures, it is also difficult to segment them well using existing cues.

Level Sets on Reconstruction Confidence

The path in the bottom of the block diagram in Fig. 4.2 depicts a more accurate approach improving on the method proposed in section 3.3. The difference lies in the patch segmentation step, which rather than blindly splitting the light field in squares it adapts the size and shape of the patches. To do so, the method relies on the reconstruction confidence obtained via structure tensor, thus enabling the detection of regions with lower confidence and focusing on those areas. This method will be further explained in the remaining of this chapter.

4.2 Level Sets on Reconstruction Confidence

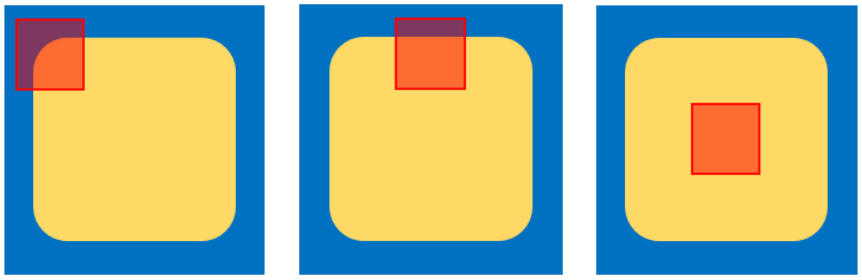
In the previous section we described the reconstruction methods presented so far and proposed an improvement on the algorithm presented in 3.3. This section further develops the introduced method by detailing its inner workings. We start by describing the confidence measure and its computation, followed by how we use that information to segment the light field and how reconstruction is approached.

4.2.1 Confidence Measure

In Section 2.4.1 we describe how the structure tensor can be used to estimate the slopes of lines in the epipolar plane images and present a method to compute it. However, it is possible to extract more than gradient information from the structure tensor. Specifically, a confidence metric for the reconstruction is provided through the eigenvalues of the structure tensor, λ_{max} and λ_{min}.

In a structure tensor, $S(k, l)$, the eigenvector corresponding to the greatest eigenvalue, λ_{max}, indicates the dominant gradient direction. Therefore, the more pronounced a gradient is, the greater its maximum eigenvalue.

Orthogonally we have the eigenvector corresponding to the smallest eigenvalue, λ_{min}, thus the more uniform the gradient directions the smaller the value of λ_{min}.



(a) Corner point
large λ_{max} and λ_{min} values

(b) Edge point
large λ_{max} and $\lambda_{min} \approx 0$

(c) Uniform point
$\lambda_{max} \approx 0$ and $\lambda_{min} \approx 0$

Figure 4.3: Examples of gradient variation in a point highlighted in red: (a) a corner point which has gradient changing in multiple directions; (b) an edge point which has gradient changing along one direction only; (c) uniform region which has a gradient close to zero.

To exemplify, corner points (see Fig. 4.3(a)) which possess high gradient in multiple directions, and are associated with strong texture changes, yield high values for both eigenvalues. Regions where a single gradient direction is dominant, meaning they are composed of edge points (see Fig. 4.3(b)), result in a large λ_{max} and a small λ_{min}. Following the same logic, uniform regions (see Fig. 4.3(c)) cause both eigenvalues to be close to zero due to the low texture that characterizes them.

Since the structure tensor is considered in the epipolar plane images, the focus is to provide a confidence level for the detected edges. Thus, the difference between the eigenvalues should suffice as a confidence metric for any given point. Specifically, the confidence c in each point is defined as:

$$c = \lambda_{max} - \lambda_{min} \tag{4.1}$$

as proposed in [6].

Once the structure tensor and the corresponding eigenvalues are calculated one has a confidence map that provides the information depicted in Figs. 2.1 (b) and (c).

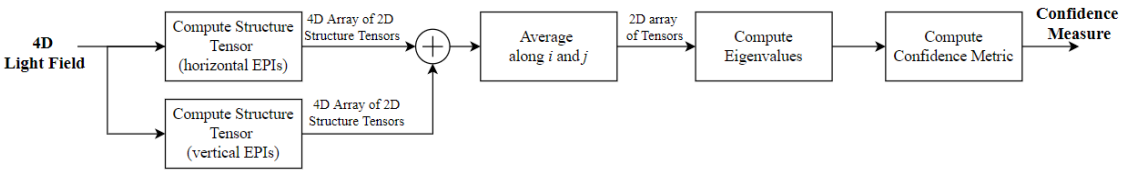
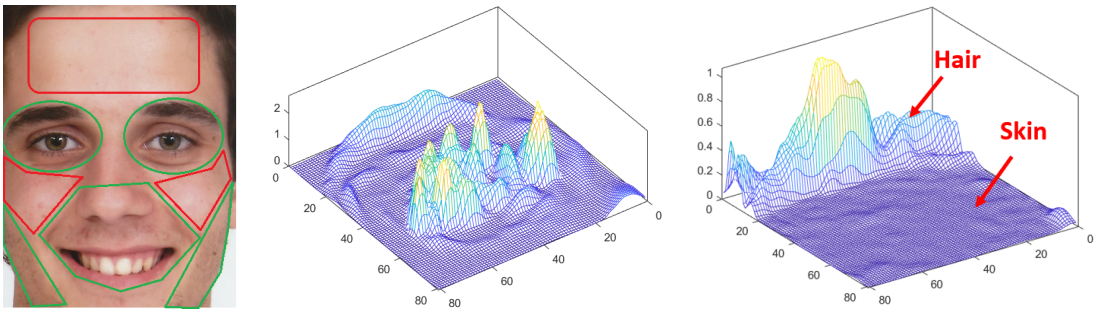


Figure 4.4: Block diagram of the confidence calculation process.

Figure 4.4 shows the confidence calculation process. Starting with the 4D light field, we compute the structure tensor in both horizontal and vertical EPIs resulting in a 4D array of 2D structure tensors. The arrays are then added and the result averaged in i and j yielding in a 2D array of tensors. We compute the eigenvalues for each tensor in this array, and then assign a confidence value calculated according to the formula presented in 4.1. An example of a confidence map pertaining the whole image as shown in Fig. 4.7 (a).

4.2.2 Patches defined by Level Sets

In section 2.1 we described the types of gradient regions found in the face, see Fig. 4.5 (a), and in section 4.2.1 we relate the gradient in the image with the reconstruction confidence metric extracted from the structure tensor.



(a) high/low gradient green/red areas

(b) green area, eye, 3D reconstruction confidence

(c) red area, forehead, 3D reconstruction confidence

Figure 4.5: Types of gradient regions found in the face. In (a) general mapping of regions, with red outlining low gradient regions and green the high gradient regions. In (b) mesh plot detail of the confidence information found in the eye region (green ellipse), where discernible features are usually found. In (c) mesh plot detail of the confidence information found in the forehead, with hair and skin regions highlighted.

An example of the confidence in smooth regions is presented in Fig. 4.5 (c), where the frontier between hair and forehead (at the top) displays medium/high confidence and then the forehead presents low confidence. Conversely, the eye region in Fig. 4.5 (b) comprises mostly medium to high confidence due to the high gradients present.

To attain a less constrained patch division, we use the confidence measure provided by the structure tensor, as described in 4.2.1 and shown in Fig. 4.7 (a). A binary confidence map is obtained via threshold of the confidence measure, thus determining which areas to reconstruct. An example of the confidence map is depicted in Fig. 4.7 (b), where lighter regions correspond to high confidence and darker regions correspond to low confidence relative to the threshold.

The segmentation is performed on top of the confidence map, resorting to the distance transform which enables the extraction of level curves. An intermediate step filters the level curves by size to choose only interesting regions, not too big nor too small. In Fig. 4.7 (b) we see the level curves overlaid on the confidence map and we verify that they are over the expected smooth regions of a human face.

The level sets, defined as the region between level curves, comprise regions with similar confidence level and are assumed to have similar depth. Thus, they can be split onto patches as shown in Fig. 4.7 (c).

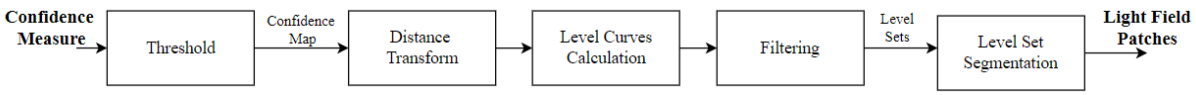
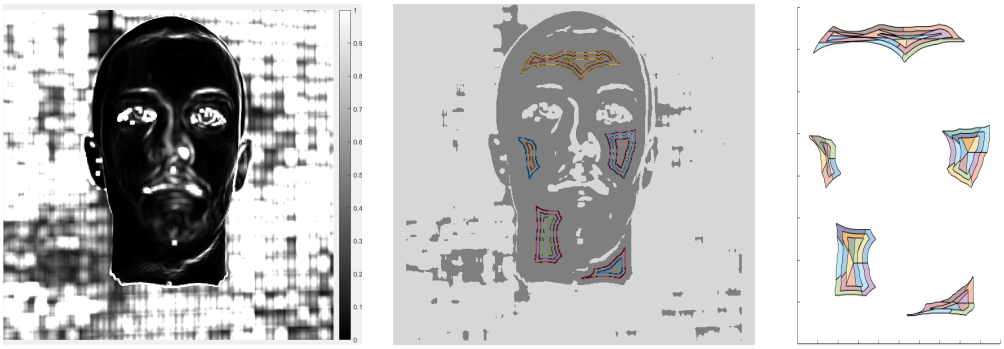


Figure 4.6: Block diagram of the process to attain light field patches from reconstruction confidence.

This process, summarized in Fig. 4.6, enables the segmentation to focus on areas where the confidence is low, as opposed to the indiscriminate patch segmentation in a regular grid scheme.

Once the light field segmentation is attained we possess a set of light field patches. Each one passes through the process depicted in Fig. 4.8.



(a) Confidence measure (b) Level curves over confidence map (c) Level set segmentation

Figure 4.7: Stages of patch segmentation using reconstruction confidence. In (a) example of the confidence measure for the scene presented in Fig. 5.4. In (b) confidence map with the filtered level curves overlaid, dark gray represents zones below the threshold and light gray the zones above. In (c) radial segmentation of the level sets shown in (b), yielding small patches.

The first step is shearing for a set of disparities uniformly distributed inside the estimated disparity range, resulting in a set of sheared light fields. Then, we proceed to compute photo-similarity across the viewpoints of each sheared light field, which is calculated as the difference between the central viewpoint and every other viewpoint. In this step a mask is used to ensure only the pixels inside the patch are considered in the calculation.

The resulting array of errors for each patch, caused by the differences in the viewpoints, enables depth assignment through the minimization of the error value.

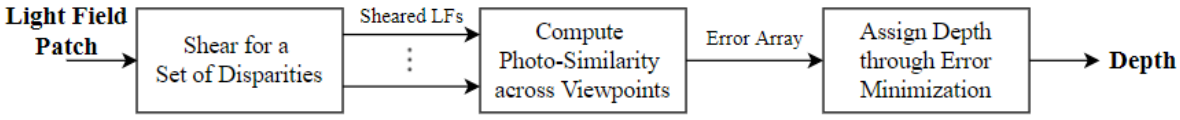


Figure 4.8: Block diagram of the depth estimation based on light field patches extracted from the reconstruction confidence.

4.2.3 Optimization of the Estimated Depths

To have a more robust and complete algorithm an optimization step is included, complementing the described algorithm. Specifically, the 3D reconstruction from patches step, depicted in Fig. 4.8, is included in the optimization loop.

We propose an optimization scheme that targets individual zones, meaning each cluster of level sets is independently optimized. One of such clusters is exemplified in Fig. 4.9. There, we see a zone comprised by three level sets, each segmented into four patches. Each patch has an associated depth value, $z_{j,i}$, to be estimated. However, we want the depth estimations to be coherent, i.e., we don't want abrupt changes in neighboring patches since the corresponding real areas do not possess such changes. To achieve this, we can constrain the values of neighboring patches to have some similarity, as indicated by the green and blue arches in Fig. 4.9. Since each zone is globally optimized, we obtain local consistency within the smooth areas.

Let $z = [z_{1,1},\ ...,\ z_{S,P}]$ denote the set of depths corresponding to each patch of a segmented region, where

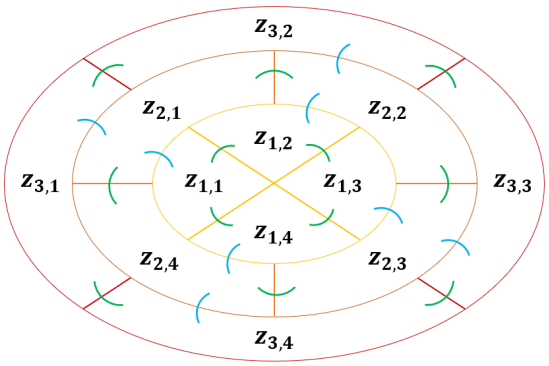


Figure 4.9: Scheme of similarity constraints, represented by arches, used in the optimization. Green arches enforce level similarity while blue arches enforce radial similarity.

S denotes the number of level sets in the region and P the number of patches in a single level set. Then, the optimization problem to solve in each segmented zone can be written as:

$$\min_{z} \quad C_S(L,z) + \omega_1 * C_C(z) + \omega_2 * C_R(z) \tag{4.2}$$

where L denotes the light field, and the regularization weights are ω_1 and ω_2.

The cost function in Eq. 4.2 accounts for three constraints. The first is the shearing cost, C_S, which corresponds to a function that performs shearing and computes the average similarity error in the viewpoints of the sheared light field, as defined in Eq. 3.3.

The remaining constraints enforce neighborhood consistency. To explain this constraints Fig. 4.9 contains an example of a segmented zone, with three level sets ($S = 3$) and four patches per set ($P = 4$). The depths contained in the array z are also marked to aid in the reading of the cost formulas. The second constraint is imposed by a level/crown cost:

$$C_C(z) = \sum_{j=1}^{S} \left(\sum_{i=1}^{P-1} \left(|z_{j,i+1} - z_{j,i}|^2 \right) + |z_{j,1} - z_{j,P}|^2 \right) \tag{4.3}$$

which enforces similarity between neighboring patches within each level set, as shown in Fig. 4.9 by the green arches. The third constraint is imposed by a radial cost:

$$C_R(z) = \sum_{i=1}^{P} \sum_{j=1}^{S} |z_{j+1,i} - z_{j,i}|^2 \tag{4.4}$$

which enforces similarity between neighboring patches in adjacent levels, as shown in Fig. 4.9 by the blue arches.

4.3 Summary of the Proposed Methodology

We propose an algorithm for depth estimation in low gradient areas. This algorithm was designed to serve as a complement to an edge-based method that would perform the first reconstruction. Then, the proposed method should be used to refine the results obtained in smooth light field regions.

The proposed algorithm, fully depicted in Fig. 4.10, comprises two main blocks: extraction of patches from

the reconstruction confidence and 3D reconstruction from patches.

The first block computes the reconstruction confidence and then performs the light field segmentation. It receives a light field as input, computes the structure tensors in the horizontal and vertical epipolar plane images. Then, it averages them along the viewpoint directions (i, j) resulting on a 2D array of structure tensors. From this array, it computes the eigenvalues for every structure tensor and extracts the reconstruction confidence for each point. A threshold is then applied to the confidence measure yielding a confidence map. The distance transform is applied to the confidence map and level curves are computed on the low confidence regions, yielding the level sets. Each level set is then radially split onto pieces and the light field is segmented accordingly, yielding a set of light field patches corresponding to low confidence areas.

The second block performs the reconstruction from the light field patches. Each light field patch, assumed to have constant depth, is sheared for a discrete set of disparities, yielding a number of sheared light fields refocused at different depths. For each sheared light field, the photo-similarity is computed across viewpoints, resulting in an error array. Lastly, we assign to the patch the depth associated with the smallest error (greater photo-similarity).

The optimization step occurs on the "3D Reconstruction from Patches" block (see green dashed block in Fig. 4.10), targeting individual zones for independent optimization. The proposed strategy searches for the shearing depth that minimizes the sum of squared differences between the viewpoint images in the sheared light field, while simultaneously it enforces similarity between adjacent patches, yielding an overall consistent depth estimation for each smooth area.

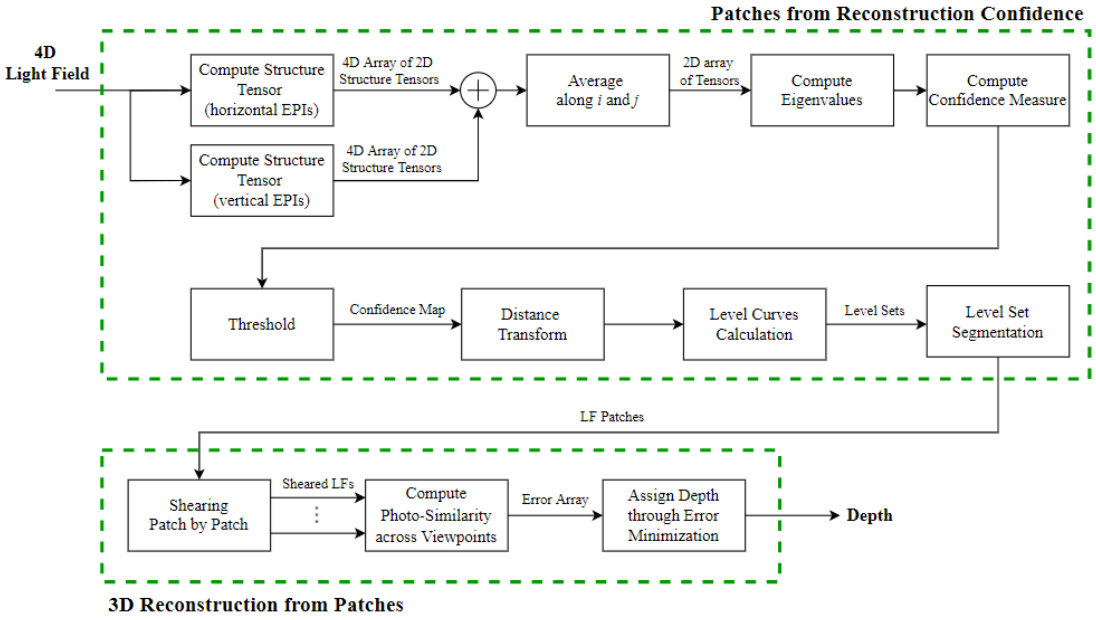


Figure 4.10: Block diagram of the method to estimate depth based on patches from reconstruction confidence.

Chapter 5

Face Reconstruction Experiments

In the previous chapters four methods for face reconstruction were presented. In this chapter, we test those methods on synthetic and real light field data. The first section is dedicated to describing the creation of a synthetic face model and comparing the two methods used for acquisition purposes. The remainder of the chapter is dedicated to the four methods, with one section per method. There, we present experiments to assess the performance of the spinning parallelogram operator (SPO) [58] and light field superpixel segmentation (LFSP) [28] methodologies on faces. In the end, we present the experiments that lead to the creation of our method followed by the results attained with it.

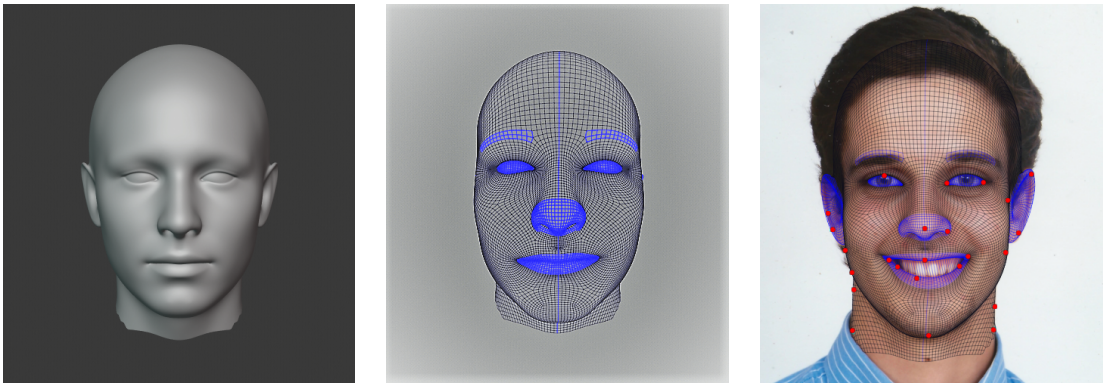
5.1 Synthetic Data

The current Covid-19 pandemic imposed severe constraints on this work, in particular precluded the use of real cameras and setups throughout most of the work. A choice was made to start with artificial data, which lead to the creation of virtual setups and light field datasets, and switch to real data in a later stage. The first challenge in the creation of virtual reality datasets is to obtain a three dimensional model of a face. Then, a scene ought to be composed to include the three dimensional face model in order to acquire a synthetic light field that resembles a real scene.

3D Face Model Generation The solution to the problem of 3D face modelling was found resorting to *Blender* [7] and its add-on *FaceBuilder* [27] from *KeenTools*. These tools allowed a quick and easy creation of 3D face models, without the need for expensive 3D scanners or photogrammetry rigs, requiring only a standard camera such as a webcam or a smartphone camera.

The creation of the model using *KeenTool*'s add-on consists in four main steps. The first step is to add a "blank" model to the *Blender* [7] scene, which consists of a texture-less bust with standard features, like the one displayed in Fig. 5.1 (a).

The second step is to choose reference photos of a face, which will provide texture to the model (skin, eyes, etc.). While one reference picture is enough for a start, as proven by the example in Fig. 5.2 (a), the usage of multiple reference pictures is highly recommended. If the focal length and sensor width of the camera are unknown, *FaceBuilder* [27] estimates the necessary parameters in order to create the best possible result.



(a) Standard Model (b) Face Mesh (c) Adjusted Mesh

Figure 5.1: Face modeling process. In (a) default 3D face model by FaceBuilder [27] without texture. In (b) mesh of the face to be adjusted to the picture, which will enable texture generation and bust sculpting. In (c) face mesh aligned with the provided reference photograph (picture of the author of this document).

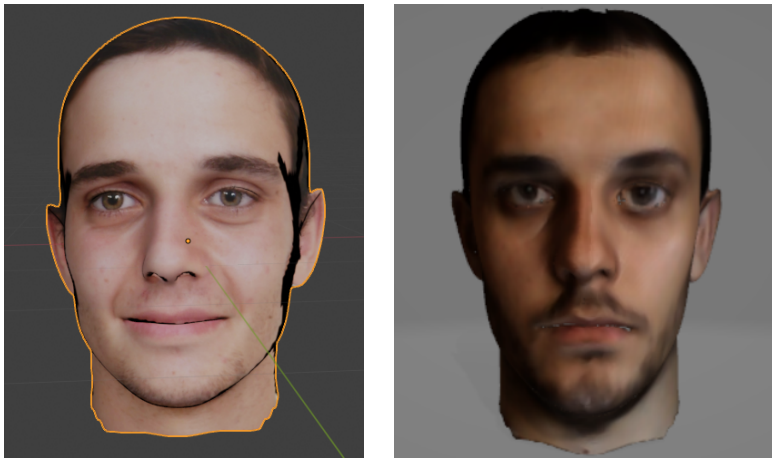
The third step is to shape the model, for which a mesh of a face is used like the one in Fig. 5.1 (b). Each point of the mesh can be adjusted in order to obtain the best fit possible to the face in the reference picture, as shown in Fig. 5.1 (c). Furthermore, it is visible that a small number of points, displayed in red, suffice to adjust the mesh correctly. Also worth mentioning that the facial expression, in this case the smile, can be captured but leads to extra adjustment points for the model. This step can, and should, be repeated with pictures from several perspectives in order to create a more complete model. However, all pictures must be taken with the same camera and have no lens distortion.

The last step is to use the *FaceBuilder* [27] to generate the texture based on the previous step, yielding a 3D model with a "real" texture, as shown in Fig. 5.2. The effects of using only one reference picture to create the texture are shown in Fig. 5.2 (a), particularly, on the right side of the face there is a black zone of undefined texture, which is due to the lack of information about that area in the provided reference picture. It also proves the ability *FaceBuilder* [27] has to model facial expressions. A second model, shown in Fig. 5.2 (b), was created using multiple reference pictures. The lack of illumination in one of the reference pictures is visible by the presence of shadows in the texture of the model, despite that a model without texture gaps was obtained.

Dataset acquisition Since the datasets are intended to resemble real world scenarios, scenes ought to be composed including the 3D face model (obtained as previously described) in a more realistic environment. Two approaches were used for the purpose of dataset creation and acquisition.

The first approach resorts to VRML [12] for scene composition and to Matlab's Virtual Reality toolbox for the light field acquisition. This approach served as an entry point for dataset creation and allowed preliminary results to be obtained. However, with this approach the intrinsic matrix $\mathbf{H}$ has to be estimated and the ground truth information is harder to obtain.

The second approach comprises both scene composition and light field acquisition in Blender [7]. This approach enables a more advanced scene composition, with fully customizable features such as image resolution, scene illumination and specularity. The light field was acquired resorting to an add-on for Blender [24], which outputs not only the light field's viewpoint images, but also ground truth information, camera parameters and metadata.

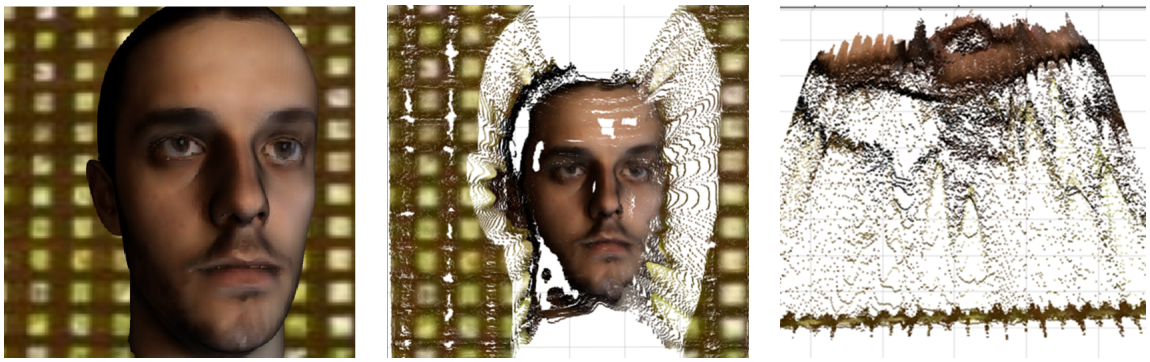


(a) Single source model (b) Multi source model

Figure 5.2: Resulting 3D models: (a) obtained using only one reference picture, the black zones have no attributed texture (for instance the sides of the face); (b) obtained using multiple reference pictures.

It is important to note that the scene created with Blender is not, by design, in metric units. However, for reference: the focal distance is 8 units, model width (distance between ear tips) is 1.5 units, model height is 2.3 units and the background is 3 units away from the tip of the nose. Please refer to Annex B for a sub-sampling of the viewpoint images of the Blender generated light field.

To further evaluate the quality of the datasets a depth reconstruction algorithm, proposed by Simão Marto [32], was applied to a similar dataset produced by each of the approaches.



(a) Central Viewpoint (b) 3D Reconstruction (Front) (c) 3D Reconstruction (Profile)

Figure 5.3: Reconstruction using the VRML + Matlab method. In (a) central viewpoint of the light field. In (b) frontal view of the resulting reconstruction. In (c) profile view of the resulting reconstruction.

The reconstruction results are shown in Fig. 5.3 for the VRML/Matlab approach and in Fig. 5.4 for the Blender approach. In both figures the central viewpoint is shown in (a), a frontal perspective of the reconstruction is shown in (b) and a profile view of the reconstruction is shown in (c). The differences between both central viewpoints (Fig. 5.3 (a) and Fig. 5.4 (a)) lie in camera pose and ambient illumination.

Regarding the reconstruction results a fundamental difference is found, even tough the VRML approach yields a larger reconstructed area (see Figs. 5.3 and 5.4 (b)) it is much noisier than the Blender counterpart. This is

(a) Central Viewpoint (b) 3D Reconstruction (Front) (c) 3D Reconstruction (Profile)

Figure 5.4: Reconstruction using the Blender method. In (a) central viewpoint of the light field. In (b) frontal view of the resulting reconstruction. In (c) profile view of the resulting reconstruction.

particularly noticeable, when comparing the face silhouettes in the profile perspective (see Figs. 5.3 and 5.4 (c)) where the VRML reconstruction shows blurred features and the Blender reconstruction clearly distinguishes small features such as the gap between the lips. This results are influenced by the number of bits encoding each image, $8\ bits$ for the VRML setup and $64\ bits$ for the Blender setup.

To conclude, the Blender approach [7, 24] is a clear choice not only for the reconstruction quality displayed but also for the level of customization it enables in the scene composition and the fact that it provides ground truth information plus camera information.

5.2 Spinning Parallelogram Operator

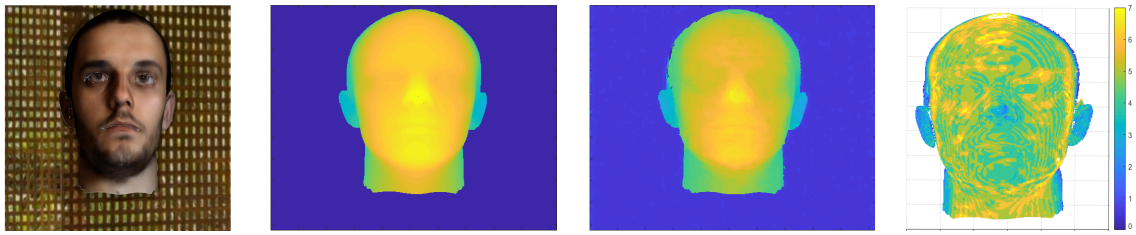
In this section we evaluate the performance of the spinning parallelogram operator (SPO) on both synthetic and real data. The SPO code made available with [58] outputs a depth map containing only discrete labels. The conversion to metric structure required to assess the quality of the results, however an intrinsic matrix **H** is required. We start by analyzing the results in synthetic data and then we progress onto real data.

Synthetic Data We first test this method with the light field acquired in the Blender setup, described in the previous section and whose central viewpoint is depicted in Fig. 5.5 (a). In the absence of the intrinsic matrix **H** a qualitative analysis was used, followed by quantitative analysis resorting to a linear regression to recover metric information.

A choice was made to use more than the empirical knowledge extracted from looking at the results for the qualitative analysis. Therefore, the ground truth information was labeled so as to match the labeling performed by the method, under the assumption of a regular labeling. The resulting labels are shown in Fig. 5.5 (b), which can now be compared to the SPO output.

Initial comparison between Figs. 5.5 (b) and (c) reveals high similarity, despite some difference in the color levels, which corresponds to a good depth discrimination.

The labeling error is defined as the difference between the ground truth and estimated depth labels. Since the object boundaries generate high errors, due to the background/foreground frontier, we filter the error so as to consider only the face region (see Fig. 5.5 (d)).



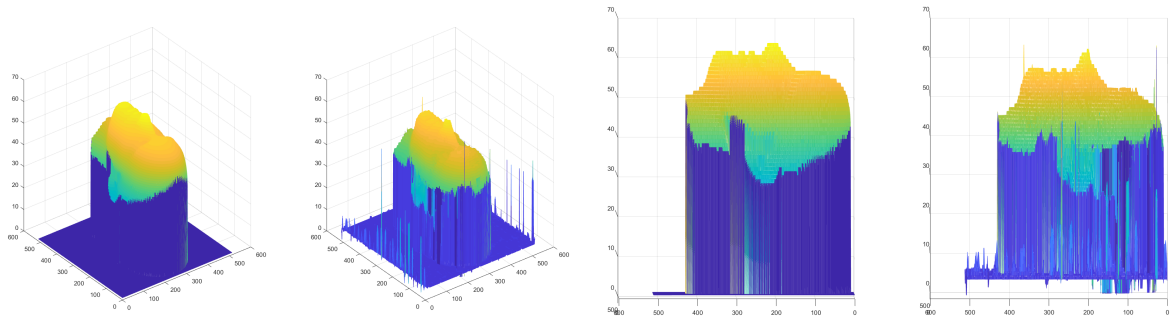
(a) Central viewpoint image (b) Ground truth (c) SPO estimation (d) Filtered error

Figure 5.5: Graphics used in the quantitative analysis of the SPO. In (a) central viewpoint image of the acquired light field. In (b) ground truth (GT) information labeled to match the SPO output. In (c) SPO estimation of depth labels. In (d) the difference between GT and estimated depth labels, i.e. the labeling error, pertaining the face region.

The error in the background is approximately constant, presenting an average offset of 4 labels of the estimated labels against the ground truth. Regarding the foreground, we see a slightly higher labeling error, going up to 7 labels of difference in some points. One must also assume that quantization errors exist due to the labeling process, thus explaining part of the errors visualized.

To obtain a clearer impression of the qualitative results, a mesh plot of both ground truth and SPO depth labels was created. First, we analyze both meshes in perspective as shown in Fig. 5.6 (a). The presence of outliers in the SPO estimation is clear in the corresponding mesh, predominantly in the silhouette of the face and in the image borders, therefore a filtering step can be applied to handle outliers. Moreover, despite the aforementioned offset in the labels, a clear background/foreground separation is attained, with similar contours to the ground truth image.

For a deeper analysis of the obtained contours, a profile view is also considered, as shown in Fig. 5.6 (b). The contour line from the neck to the forehead is faithful to the ground truth, with the details in the lips that being well captured, however the estimation caused a slight elongation in the nose and some discontinuities in the forehead region. Furthermore, it is visible that the sides of the face present an increased challenge to the estimation, since the contours are not very well defined.



(a) Perspective view (b) Profile view

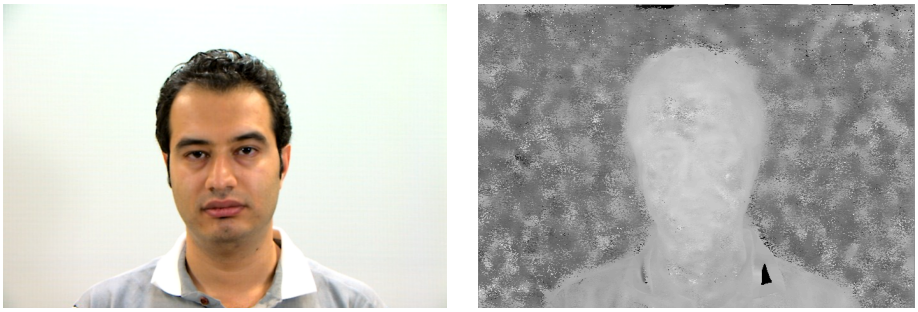
Figure 5.6: Side by side comparison of the mesh plots of the labeled depths, with the ground truth on the left and the SPO labels on the right. In (a) perspective view where it is noticeable a good background/foreground separation. In (b) profile view where the face silhouette details are noticeable.

For the quantitative analysis metric values are required. We perform a linear regression to estimate an affine transform from the depth labels to the metric values provided in the ground truth image. Since our interest lies

within the region of the face the transform will be first calculated there. Then, we compute the mean absolute error (MAE) of the estimated depth against the ground truth information, yielding a value of 0.02 units for the face region and 0.03 units for the whole image (approximately 4mm).

This approach, despite being successful in converting to metric structure uses information that should only be used in the comparison stage. Thus, the obtained error values are small and mostly due to the labeling errors. Additionally, since the face silhouette presents high error, considering the whole image drives the mean absolute error up.

Real Data Up to this point we only have an indication that the approach might work, however it requires further validation. To do so, we evaluate the results obtained for real data from the IST-EURECOM face database [45]. In the end we present the results attained with real data acquired by us.



(a) Central viewpoint (b) Depth labeling

Figure 5.7: Application of the SPO to a face in the IST-EURECOM Face Database [45]. In (a) central viewpoint image. In (b) obtained depth labeling.

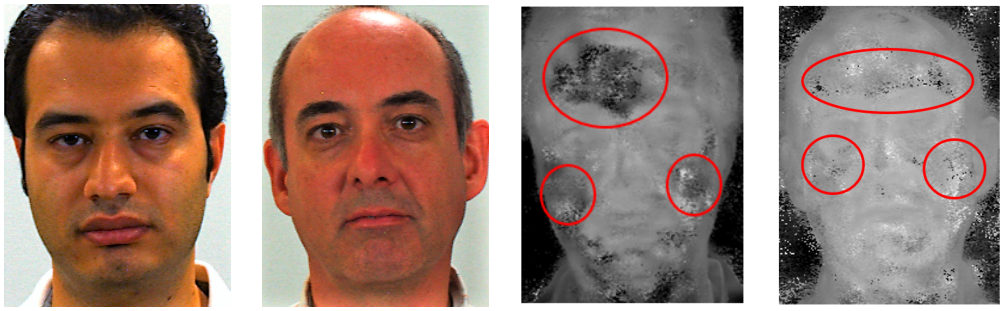
A first run was performed using the whole light field, whose central viewpoint image is depicted in Fig. 5.7 (a). The results, shown in Fig. 5.7(b), present a decent background/foreground separation but lack in detail for the face region. Furthermore, the processing of the whole light field consumes a significant amount of time.

Since only about 20% of the image is relevant to our purposes of face reconstruction, we can reduce the light field to a more significant area. The face database [45] provides a bounding box for all the faces, therefore we reduced the light field to the region of the face using the given bounding box coordinates plus a 20 pixel frame. The resulting light fields' central viewpoint images are depicted in Figs. 5.8 (a) and (b).

The application of the SPO in this trimmed light field not only yields results significantly faster but also shows improvement in the quality of the results, specifically in the level of detail. The results for both subjects are presented in Figs. 5.8 (c) and (d).

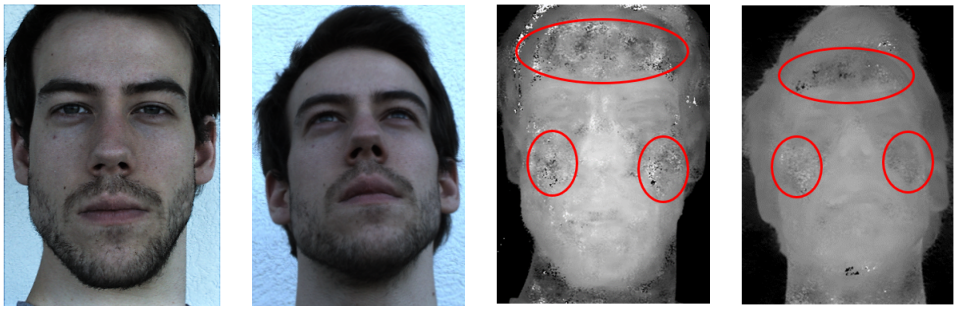
A good background/foreground separation was attained as well as better discrimination of the labels in the face. However, as predicted, the method struggles with low gradient regions and yielding estimation errors. The highlighted regions in Fig. 5.8 (c) show that the depth estimates were close to background depth. For the second subject, in Fig. 5.8 (d), the errors are not so clear but are still present.

Lastly, we evaluate the method on light fields acquired by us with a Lytro Illum camera. Specificaly, two light fields acquired in "selfie" position, yielding a depth range for the scene between 0.6 and 2 meters. As done with the data from the IST-EURECOM database [45], we consider only a fraction of the light field, corresponding to a bounding box of the face on the central viewpoint image. The cropped central viewpoint images are presented in Figs. 5.9 (a) and (b) and depict the same subject captured from two distinct perspectives.



(a) Subject 001 (b) Subject 002 (c) Sub. 001 Depth (d) Sub. 002 Depth

Figure 5.8: Application of the SPO to two faces in the IST-EURECOM Face Database [45]. Central viewpoint images: (a) for subject 001, and (b) for subject 002. Depth labeling with bad result highlight: (c) for subject 001, and (d) for subject 002.



(a) Pose 1 (b) Pose 2 (c) Pose 1 Depth (d) Pose 2 Depth

Figure 5.9: Application of the SPO to data acquired by us. Central viewpoint images: (a) for pose 1, and (b) for pose 2. Depth labeling with bad result highlight: (c) for pose 1, and (d) for pose 2. [Credit for the light field acquisition goes to Miguel Rodrigues (in the pictures).]

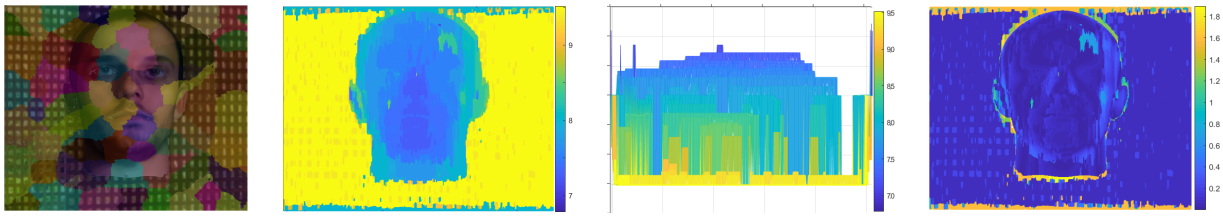
Regarding the depth estimation results, first and foremost we attain a good background/foreground separation. Furthermore, the attained depth estimation, shown in Figs. 5.9 (c) and (d), captured interesting features like the lips, nose and eyes even if in a coarse representation.

However, we see once more a struggle with smooth regions, highlighted in Figs. 5.9 (c) and (d), reveal inconsistency in the depth assignment. Despite not being such a relevant zone, some regions in the neck also reveal this struggle (see darker spots in Fig. 5.9 (c)). Nonetheless, this further adds to the conclusion that the edge-based reconstruction algorithms would benefit of a method for estimation in low gradient areas.

5.3 Light Field Superpixel

In this section we evaluate the performance of the light field superpixel (LFSP) in depth estimation, on both synthetic and real data. The LFSP method implicitly produces a disparity estimation, therefore it is possible to estimate depth knowing the intrinsic matrix **H**. We start by analyzing the results in synthetic data and then we progress onto real data.

Synthetic Data The synthetic data used is the same as in the SPO. Several superpixel sizes were tested from 10 to 50, in increments of 10, as well as sizes 75 and 100. However, the best results for this data were attained with a superpixel of size 50, whose segmentation is depicted in Fig. 5.10(a). The segmentation near the boundary of the face tends to include foreground and background areas, this is due to the large superpixel size chosen. This choice was focused on another criteria, the clustering of smooth regions of the face, e.g. portions of the forehead or cheeks, which was accomplished.



(a) Segmentation obtained (b) Depth estimation (front) (c) Depth estimation (profile) (d) Absolute error obtained

Figure 5.10: Light Field Superpixel results. In (a) central viewpoint image of the acquired light field with the superpixel segmentation overlaid. In (b) depth estimation from a frontal perspective. In (c) depth estimation from a profile perspective. In (d) difference between GT and obtained depth estimation, i.e., the error.

The depth estimation, shown in Figs. 5.10 (b) and (c), yielded a good background/foreground separation and a correct depth range. However, a significant lack of detail in the minor structures is noticeable. For instance, the nose is not clearly outlined, having just the tip protruding in the estimation (see Fig. 5.10 (c)).

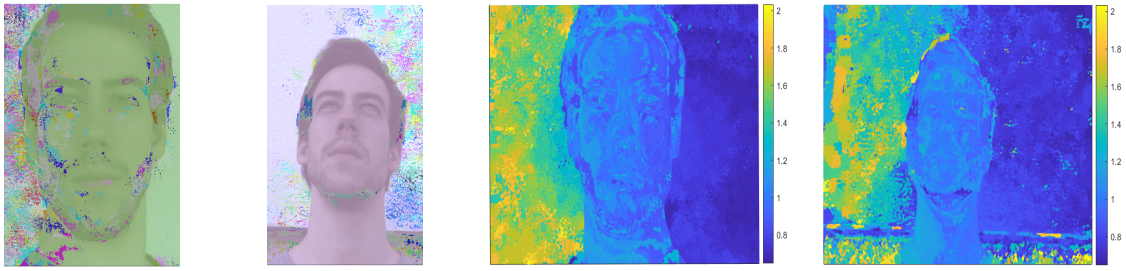
Comparing the estimated depth with the ground truth information yields the error map shown in Fig. 5.10 (d). There, an high error in the silhouette of the face is visible, which is due to errors in the background/foreground separation. The mean of absolute differences between ground truth and estimated depth is 0.17 units (approximately 2.3 cm). This value is high but explained by the error in the object frontiers, each background/foreground mistake accumulates around 2 units in error (approximately 27 cm). To test this hypothesis we compute the median error, yielding 0.07 units (approximately 9 mm), thus supporting the previous claim.

Several regions of the face stand out for having above average error, depicted by lighter blue in Fig. 5.10 (d). These regions are characterized by low gradient and higher brightness, thus reinforcing the need for better depth estimation in smooth areas.

We conclude that the method yields a correct depth range, correctly estimating the general distance of objects, however failing to provide correct depth for finer details.

Real Data To further evaluate this method's performance we use real data acquired by us. Specifically, we use two light fields acquired with a Lytro Illum camera in "selfie" position, yielding a depth range for the scene between 0.6 and 2 meters. We experiment with the same superpixel sizes used in synthetic data. However, only two superpixel sizes, 20 and 50, produce relevant results. The intrinsic matrix $\mathbf{H}$ obtained via camera calibration is used to obtain a metric reconstruction.

The segmentation with superpixels of size 20 is rather poor and small, Figs. 5.11 (a) and (b) show a crop of the central viewpoint images with the segmentation overlaid. Regarding the depth estimation, shown in Figs. 5.11 (c) and (d), a correct depth range was attained for the face, despite great estimation error in the background (see dark blue regions which correspond to foreground depth). Nonetheless, asymmetric depth estimation in smooth areas is noticeable with differences up to 15cm in the cheeks.



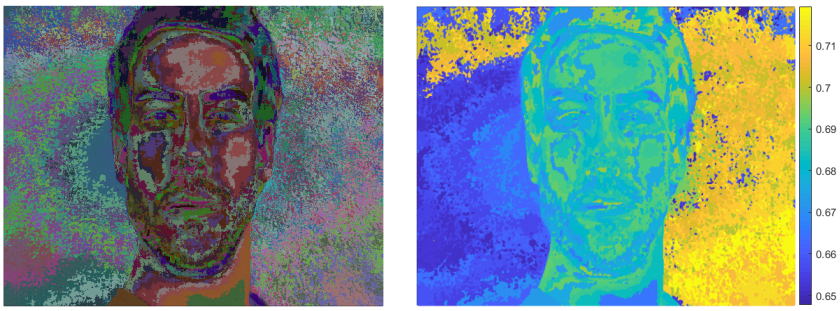
(a) Image A segmentation (b) Image B segmentation (c) Image A depth estimation (d) Image B depth estimation

Figure 5.11: Light Field Superpixel results for a size 20 superpixel. In (a) and (b) we present a crop of the central viewpoint images of the acquired light fields (A and B respectively) with the superpixel segmentation overlaid. In (c) and (d) we present the depth estimation obtained for both light fields (A and B respectively). [Credit for the light field acquisition: Miguel Rodrigues (in the pictures).]

The superpixel size of 50 yielded an interesting segmentation for both light fields, shown in Figs. 5.12 (a) and (c). The segmentation yielded better clustering of smooth regions such as cheeks and forehead. The depth estimation, depicted in Figs. 5.12 (b) and (d), maintains a large error in the background estimation (as seen for the size 20 superpixel) but still provides an acceptable depth range for the face. An unexpected problem appeared in Image B, the struggle to correctly estimate large gradient areas, noticeable by the yellow zones in Fig. 5.12 (d). There we see background depth being assigned to eyebrows, jaw, lips and nostrils. These results can be influenced by the perspective in which the light field was acquired. Furthermore, smooth regions seem to yield uneven depth estimation in both cases.

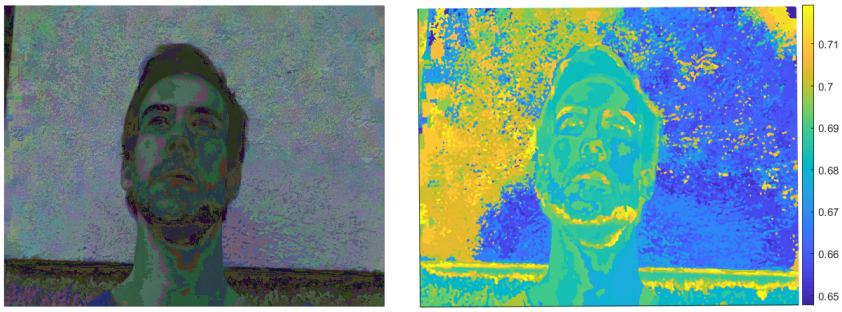
To further evaluate the results on the face and assess the severity of the inconsistencies in smooth areas, we isolated the face region for Image A.

The isolated depth estimation, depicted in Fig. 5.13, reveals a reasonable depth range and exposes some inconsistencies in depth assignment related to the superpixel segmentation. This is particularly clear in the cheeks and forehead where a heterogeneous and inconsistent depth assignment is found, resulting in “walls” or “trenches” between smooth regions. Furthermore, the level of detail presented in the reconstruction is low and clearly affected by the estimation errors previously described. Therefore, we obtained compelling evidence that this method struggles in smooth areas.



(a) Image A segmentation

(b) Image A depth estimation



(c) Image B segmentation

(d) Image B depth estimation

Figure 5.12: Light Field Superpixel results for a size 50 superpixel: In (a) and (c) we present the central viewpoint images of the acquired light fields (A and B respectively) with the superpixel segmentation overlaid. In (b) and (d) we present the depth estimation obtained for both light fields (A and B respectively). [Credit for the light field acquisition: Miguel Rodrigues (in the pictures).]

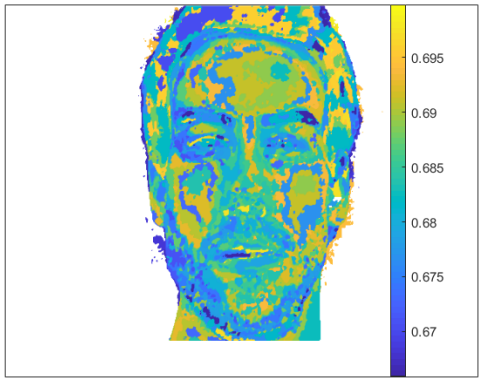


Figure 5.13: Light field superpixel (size 50) face depth estimation isolated.

5.4 Square Patches

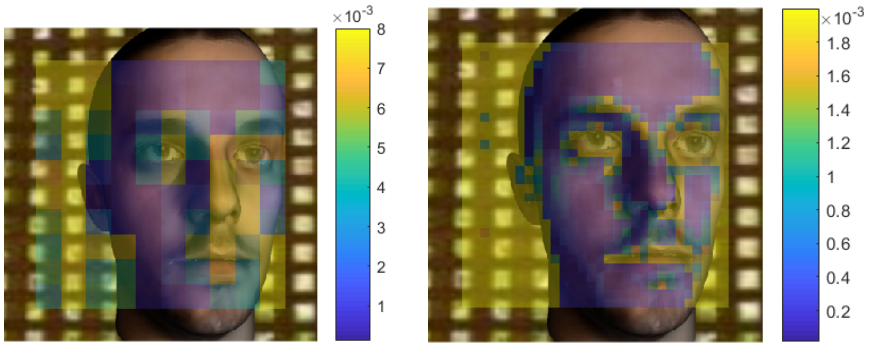
In this section we illustrate the experiments performed in faces with the method proposed in section 3.3. We first study the impact of the grid size, then we proceed to the reconstruction results.

This method was tested on the VRML generated data, with the central viewpoint image depicted in Fig. 5.3 (a). The depth range for the scene is estimated to be between 0.05 and 0.1 units with the foreground/background frontier at around 0.07 units.

Error Metrics Evaluation The first experiment evaluates the calculation of the sum of squared differences on the grid squares of a centered window, assessing the impact of the grid's size. We start with a set of global shearings and select one according to the depth range of the foreground. Specifically, from a set of ten shearings for depths uniformely spaced (inside the depth range) we selected the fifth shearing.

The calculation of the error metrics in the selected shearing was repeated several times, changing the grid size used. In detail, we start with a 3 by 3 grid (corresponding to 100 px patches) and progress through to a 30 by 30 grid (corresponding to 10 px patches). Grids bigger than 30 by 30 are considered too thin for the purpose of this study and are therefore disregarded. Furthermore, post processing was applied to the computed values, namely a threshold of the SSD value was searched in order to refine the level of detail in the region of interest of the surface plot.

The result for grids of size inferior to 10 by 10 is a coarse segmentation between foreground and background areas, with the latter yielding higher error values than the former.



(a) Saturated error on a 10 by 10 grid (b) Saturated error on a 30 by 30 grid

Figure 5.14: Central viewpoint image with saturated SSD overlaid. In (a) error computed on a 10 by 10 grid and saturated at 0.008 units. In (b) error computed on a 30 by 30 grid and saturated at 0.002 units.

For a grid of 10 by 10 interesting results emerge, shown in Fig. 5.14 (a), as the person's forehead, most of the right cheek and a portion of the left cheek, i.e. smooth regions, present a lower error value. The detection of a smaller area on the right side of the image is due to the camera's pose, which results in partial occlusion and on that half of the face occupying a smaller area of the image than it's counterpart. Furthermore, areas where the grid patches include high gradient tend to have medium/high error values, specifically regions around the eyes, nose and mouth. This can be explained by the lack of smoothness of the region, since when high gradient regions are considered the shift from one viewpoint image to another is clearer, causing the error to spike.

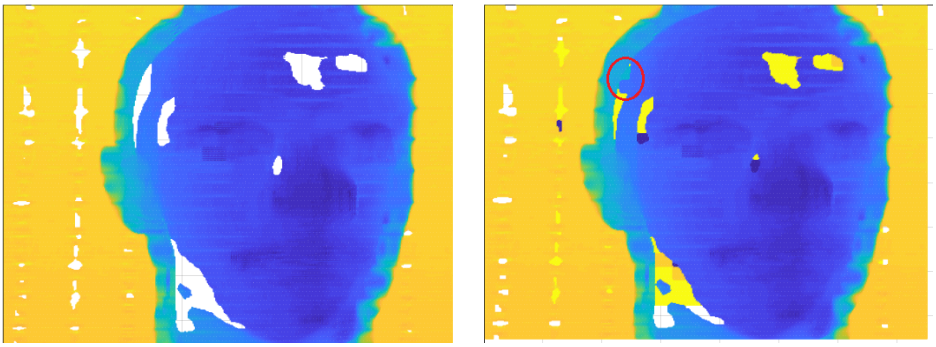
For a grid of 30 by 30 finer detail on the above analysis is achieved, as shown in Fig. 5.14 (b). The regions around the eyes, nose, ears and mouth are better defined, also leading to a better definition of smooth regions while

still displaying a low SSD value. A clear example is the region on the person's left cheek, that with a size 10 grid displayed high error level except in one patch (see Fig. 5.14 (a)). However, with a size 30 grid displays a larger number of patches with low error comprising also a larger area (see Fig. 5.14 (b)), better defining a smooth region.

Regarding computational effort, the time required to calculate the error metrics in a size 30 grid is about 30 times higher than for a size 10 grid.

Reconstruction Results The study on the impact of the grid size (N by N squares) continues, this time analyzing the reconstruction results variation with N. The obtained results were compared to the depth map obtained with the gradient based depth estimation method proposed in [32], specifically trying to fill in the blanks in regions were this method had little confidence in the depth estimation (see Fig. 5.15 (a)).

For values of N lower than 20 the results present insufficient level of detail resulting in erroneous depth estimation. For N equal to 20 we start to see accurate estimates, highlighted in Fig. 5.15 (b), however most of the estimation is still not accurate.



(a) Original depth map from [32]

(b) Joint depth from a size 20 grid

Figure 5.15: Depth maps obtained: (a) using [32], where the blank regions are to be filled with our results; (b) with our estimation in a size 20 grid filling in the blanks.

A high gradient frontier is shown (in Fig. 5.16) between the bust of the person and the background. However, it is noticeable that low gradient regions pose a challenge for the estimation (see Figs 5.15 (a) and 5.16), for instance in the center of the forehead the gradient is too low to provide information.

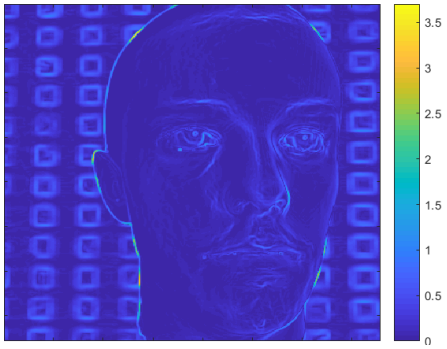
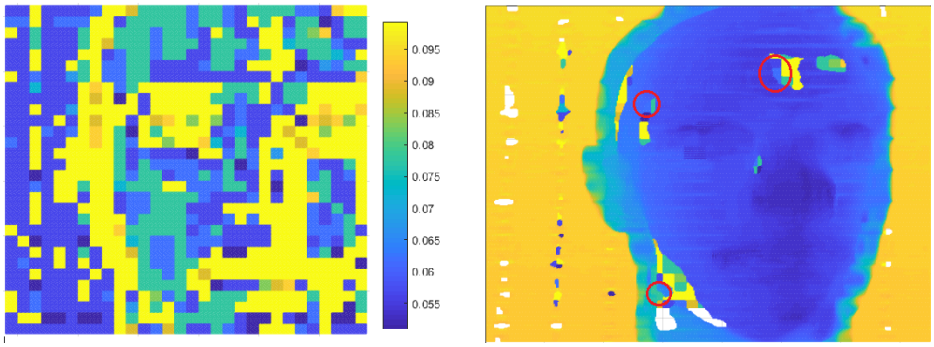


Figure 5.16: Gradient magnitude calculated in the central viewpoint.

In Fig. 5.17 (a) we present the full depth map estimated by the algorithm for N equal to 30. An outline of the

face is clear, with smooth regions presenting coherent depth estimates, specifically, we can distinguish the subject's right cheek and temple.



(a) Full depth estimation (b) Joint depth estimation

Figure 5.17: Depth maps obtained: (a) full depth estimation using our method in a 30 by 30 grid; (b) our method's results filling the blanks left by the gradient based depth reconstruction [32].

In Fig. 5.17 (b) we present the our estimates complementing the depth map presented in Fig. 5.15 (a), with the regions where the attained results are good highlighted in red.

Throughout the experimentation process the results presented in Fig. 5.17 were the best. Also, it was observed that each grid size yields different regions being correctly estimated, this is visible comparing the highlights in Figs. 5.15 (b) and 5.17 (b), and was observed for the remaining values of N tested (up to 60). For N bigger than 60 the computation time becomes too long and the results start loosing accuracy since the area considered becomes too small.

This results point to the need of adapting the size and shape of the region considered in the local shearing operation. We also conclude that its worth testing this technique with real data, since the results with synthetic data seem promising. Furthermore, the inclusion of edge points in the areas where the gradient is too low could also aid in the estimation process, namely if the edge points define the frontier of the region.

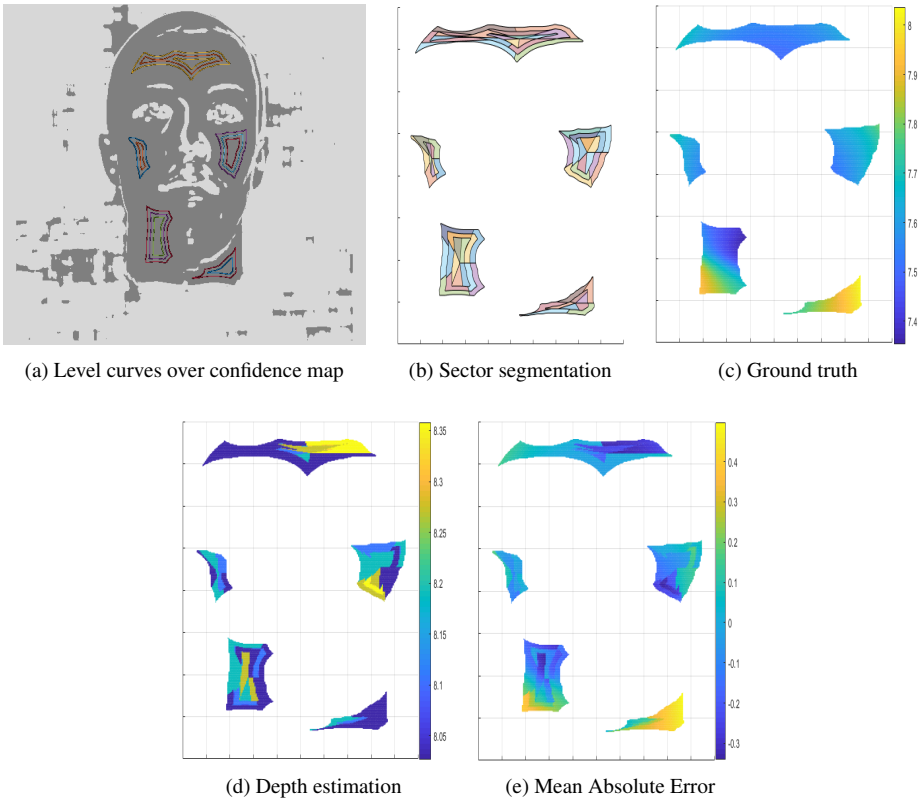
5.5 Level Sets on Reconstruction Confidence

This section is dedicated to the application of the method proposed in chapter 4 to faces. Specifically, we assess the method's performance and evaluate its strengths and weaknesses. This method aims at improving depth estimation from edge-based methods in low gradient regions. To do so, it relies on the reconstruction confidence obtained via structure tensor.

Synthetic Data We start with the application of this method to the synthetic data generated with Blender, whose central viewpoint is depicted in Fig. 5.4(a). Our main goal is to assess if this method can be used as a complement to an edge-based reconstruction algorithm, like the one proposed in [32], to improve estimation in low gradient areas.

We show the obtained confidence map for this data in Fig. 4.7 (a), where the light gray areas correspond to confidence above a threshold and dark gray areas correspond to confidence below that same level. Additionally, the figure shows the filtered level curves over the confidence map. Thus, it becomes clear that the regions with low

confidence are the smoother regions of human faces, such as forehead and cheeks.



(a) Level curves over confidence map (b) Sector segmentation (c) Ground truth

(d) Depth estimation (e) Mean Absolute Error

Figure 5.18: Depth estimation using level sets of the reconstruction confidence. In (a) reconstruction confidence map, obtained via threshold, with the level curves overlaid: dark gray represents zones below the threshold and light gray zones above the confidence threshold. In (b) level set segmentation into sectors. In (c) ground truth information for the patches of interest. In (d) depth estimation obtained. In (e) mean absolute error, ground truth minus the estimated depth.

The level sets with area within a range are segmented radially, as shown in Fig. 4.7 (b), resulting in small sectors with low confidence and assumed to have constant depth. We now possess light field patches based on each of these sectors, which are then individually sheared. After the local shearing, the viewpoint similarity is computed for each sheared light field patch enabling a depth assignment through its maximization. For this step the chosen parameters are: a confidence threshold of 0.3, a set of 30 shearings and a division of 5 sectors per level set. The resulting depth estimation is presented in Fig. 4.7 (d). Since the data was generated in Blender the ground truth information, shown in Fig. 4.7 (c), is available for comparison.

Comparing the depth ranges, we immediately see a reduction of the estimated depth range when comparing against the ground truth information. Furthermore, an estimation offset of approximately 0.65 units (around 9 cm) is visible. The shrinkage of the depth range can be explained by the assumption of constant depth within the patches, since abrupt depth variations within a patch are not captured.

To assess the estimation performance we calculate the mean absolute error (MAE). Since we have already observed that an offset exists, the MAE was also calculated with the data minus the centroid of the depth. Thus, yielding a metric that focuses on the deviations around the center of mass, which we called structural mean absolute error (SMAE).

The error map, shown in Fig. 4.7 (e), reveals a third large error area, on the top patches. Furthermore, we see some incorrect estimations on individual patches alongside accurate estimates. This phenomenon is explained by the independence of patches among themselves, which can be mitigated using a regularization step that weighs the estimate on the patch with its neighbors.

Overall the reconstruction results are positive. The obtained MAE value was 0.502 units (around 7 cm) reflecting the aforementioned estimation offset. Regarding the SMAE, we obtained a value of 0.165 units (around 2 cm), which is mostly justified by the lack of coherence between patches. However, a contribution is also made by the assumption of constant depth within the patches which prevents curvature and significant depth changes to be captured. This is particularly noticeable on the bottom patches from Fig. 4.7 (e), where abrupt changes from the jawline to the neck cannot be captured due to this hypothesis of constant depth and thus increasing the error.

Real Data To further evaluate this method's performance we use two light fields acquired with a Lytro Illum camera in "selfie" position, yielding an estimated depth range for the scene between 0.6 and 2 meters. A crop of the central viewpoint image is shown in Fig 5.19 (a).

The method starts by the estimation of the reconstruction confidence, which yields a confidence map and the level curves in smooth areas. The confidence map, depicted in Fig. 5.19 (b), has lighter areas corresponding to low values of reconstruction confidence and darker areas corresponding to high values. The level curves, overlaid to the confidence map in Fig. 5.19 (b), selected five smooth areas to perform depth estimation in.

Regarding the depth estimation results, depicted in Fig. 5.19 (c), we first observe that the estimated depth range is within the real range of the scene. Furthermore, it is visible that most of the estimation yields a value close to 0.7 meters which corresponds to the real distance of the face to the camera.

However, some inconsistencies are found in the depth assignment within zones or even level sets themselves. For instance, in the forehead region we see (in yellow) a region with significantly higher depth (20 cm) which is inconsistent with the smooth depth variations usually found in such area of the face. This effect is seen on a smaller scale in all zones, where occasional spikes in depth estimation occur, as shown by the yellow regions in Fig. 5.19 (c).

To validate the proposed methodology we estimated the depth with the edge-based method proposed in [32], which results in the depth map depicted in Fig. 5.19 (d). There we see several smooth areas where the method struggled to estimate depth. The estimated depth map was then complemented with our estimates in low confidence areas, as proposed, and the result is depicted in Fig. 5.19 (e).

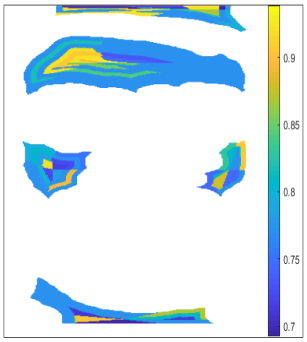
The previously described issues with our estimation are noticeable in the integrated depth reconstruction, particularly in the left side of the forehead and in the left cheek. Despite that we see that our estimation is coherent with the initial reconstruction, blending in well with the first reconstruction. This is a compelling result which demonstrates that the complementarity of both approaches can be leveraged. Note that no information from the first estimation is used in our estimation, i.e., no frontier conditions nor neighboring points information.

Another experiment was performed to assess the quality of the optimization technique proposed. Several iterations were performed with different values of the regularization parameters ω_1 and ω_2.

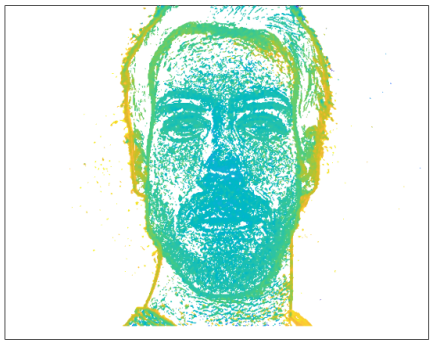
We now discuss the results obtained for $\omega_1 = 1000$ and $\omega_2 = 1$. The gradient based depth estimation remains the same as before, and is depicted in Fig. 5.20 (a). The optimized depth estimation, depicted in Fig. 5.20 (b), presents a correct depth range. However, one of the patches presents high estimation error, around 30 cm. Despite that, the remaining estimations are correct and coherent among themselves, with details like the curvature of the forehead captured. The problem of inconsistency within zones and level sets is removed with the optimization

(a) Crop of central viewpoint image

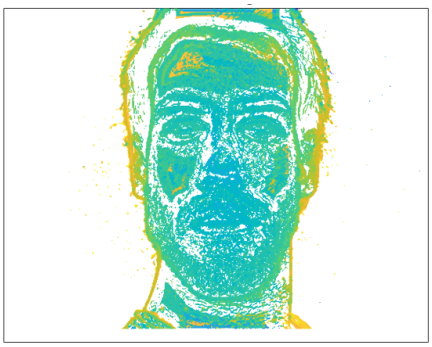
(b) Computed level curves



(c) Estimated Depth



(d) Edge-based depth estimation



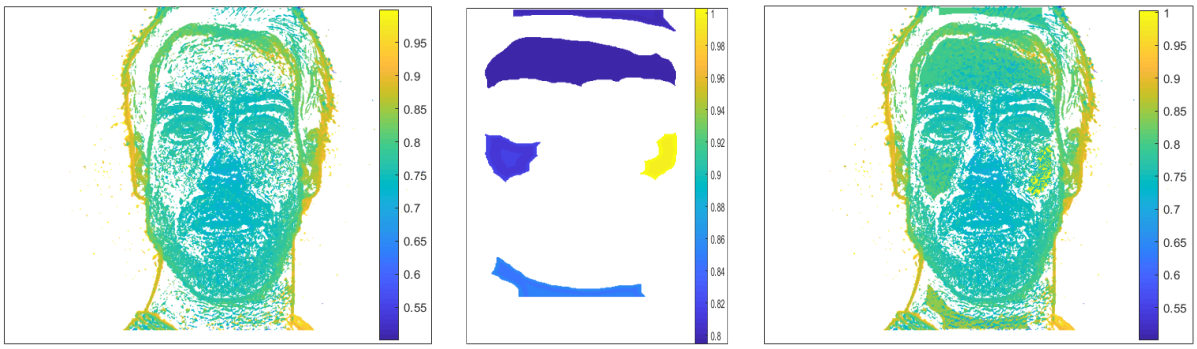
(e) Integrated depth estimations

Figure 5.19: Depth estimation using level sets of the reconstruction confidence on real data. In (a) crop of the central viewpoint image. In (b) reconstruction confidence map, obtained via threshold, with the level curves overlaid. In (c) depth estimation obtained without optimization. In (d) estimation obtained with gradient based method. In (e) gradient based depth estimation complemented with our estimation. [Credit for the light field acquisition: Miguel Rodrigues (in the picture).]

process, clearly observed when comparing Fig. 5.19 (c) against Fig. 5.20 (b). Furthermore, we see that the neck has a greater depth than the remaining portions of the face, which is coherent with the reality.

Again, to validate the proposed methodology we replace the non-estimated areas in the first reconstruction with our estimates. The resulting depth map, depicted in Fig. 5.20 (c), is overall coherent with the first reconstruction values. Specifically, the forehead presents estimations highly consistent with the closest high confidence areas (on the hair/forehead frontier). The left cheek also presents consistent estimations, even tough a small estimation error exists, around 2 cm. Two areas stand out in the integrated depth estimation. The right cheek, which as previously mentioned presents a high estimation error. The other area is in the neck, which the first reconstruction estimates with very low confidence. Analyzing the relation between the estimation on the forehead and the estimation on the neck we see a difference of approximately 5 cm, which is consistent with real average distances.

We conclude that the optimization step is valuable as it improves consistency between patch estimations. Furthermore, the complemented estimation presents a convincing result, as it fills areas that have low reconstruction confidence with depth estimates coherent with the closest high reconstruction confidence areas of the first reconstruction.



(a) Edge-based estimation (b) Our estimation (c) Combined depth estimation

Figure 5.20: Depth estimation obtained with the optimized method (real data). In (a) depth estimation obtained with gradient based method. In (b) our optimized depth estimation. In (c) gradient based depth estimation complemented with our optimized estimation.

Execution Time Analysis For a more insightful analysis of the method's performance a last experiment was performed, which evaluated the computation times of each method. The tests were performed by running the Matlab code on a machine with an Intel® Core™ i7-7500U CPU and 8GB of RAM.

The procedure comprises the application of each method multiple times, to the same light fields. For this purpose, real data was cropped to yield three different light field sizes: (i) full light field, with 625 by 433 px viewpoint images; (ii) big face bounding box, with 280 by 393 px viewpoint images; (iii) small face bounding box, with 203 by 318 px viewpoint images. From these three light fields we consider the big face bounding box (ii) the most relevant because it contains data pertaining the whole face without cropping anything, as opposed to the small bounding box, but also has the minimum clutter, as opposed to the full light field.

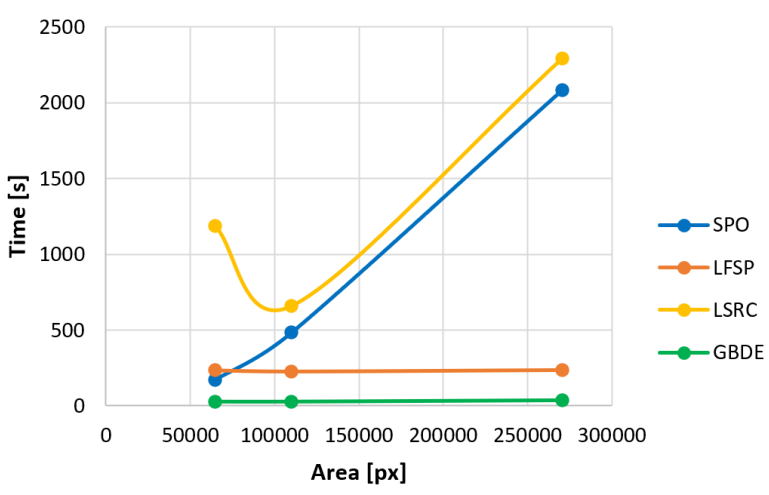


Figure 5.21: Run time vs light field size, i.e. average execution time against number of pixels considered for the methods presented in this chapter: (blue) the performance of the spinning parallelogram operator [58]; (orange) the performance of the light field superpixel segmentation [59]; (gold) the performance of the proposed method, level sets on reconstruction confidence; and (green) the performance of the gradient based depth estimation [32] used as a first reconstruction technique. Detailed values can be found in Table 5.5.

Table 5.1: Average run time (in seconds) per light field size (pixels of each viewpoint image). Tested methods: gradient based depth estimation [32] used as a first reconstruction technique; (SPO) spinning parallelogram operator [58]; (LFSP) light field superpixel segmentation [59]; and the proposed method, (LSRC) level sets on reconstruction confidence.

Area [px]	**Average Time [s]**			
	GBDE	**LFSP**	**SPO**	**LSRC**
270625	37,2	237	2083,2	2294,2
110040	29,2	228	483,8	657,8
64554	29	234,8	171,8	1189,2

The methods were applied five times to each light field size and their execution was timed. The experiment's result is summarized in Table 5.5 and graphically depicted in Fig. 5.21, with the average execution time on the vertical axis and the total number of pixels on the horizontal axis.

The gradient based depth estimation (GBDE) [32] runs in an average of 29 seconds for the cropped light fields and 37 seconds for the full light field, thus displaying a good performance and scalability. The light field superpixel segmentation (LFSP) [59] leverages parallel computing and averages just under 4 minutes for all light field sizes. Despite its good scalability this is still a high time for direct real world application. The spinning parallelogram operator (SPO) [58] outperforms the LFSP in the small bounding box, averaging just under 3 minutes. However, this method does not scale well and averages around 8 minutes for the big bounding box and 41 minutes for the full light field. Furthermore, with the full light field the results show an amplitude of approximately 5 minutes between executions on similar circumstances. Also worth noting that the method does not produce a metric depth estimation, but rather a depth labeling.

The proposed method, level sets on reconstruction confidence (LSRC), has a performance slightly below the SPO for the big bounding box and the full light field, averaging 2.9 and 3.5 minutes longer execution times, respectively. From these times, an average of 5% of the time is used for segmentation, while the remaining 95% are used for estimation. The additional run time is expected as LSRC provides reconstruction at low gradient areas of the light field, missing in GBDE or lesser precise in LFSP and SPO, and improves that reconstruction with a regularization approach.

Chapter 6

Conclusion and Future Work

The work described in this thesis comprised the study of both the plenoptic camera and the light fields acquired by it, targeting the usage of light field imagery to perform 3D reconstruction of human faces. To this goal, we studied the camera model proposed by Dansereau et al. [14], including the projection and back-projection models. These models established the theoretical foundation that enables depth estimation from light field imagery.

A conceptual study was presented to assess if reliable depth estimation could be achieved in low gradient areas by searching for the disparity that places a given area in focus. This study culminated in a basic, general purpose, reconstruction algorithm for smooth areas. This method was then refined to yield a robust and grounded reconstruction algorithm. Using the reconstruction confidence extracted from the structure tensor, enabled targeting low confidence areas (with small gradients) for reconstruction. In particular we studied the shearing operation as a tool for photo-similarity depth estimation methods.

To understand how the methods handled face information we created synthetic data and acquired real light field images. Multiple reconstruction methodologies were presented and their performance evaluated on the context of face applications, using this data. The results demonstrated the need for better estimation in low gradient areas.

Then, our method was tested and validated as a complement for the gradient based depth estimation [32]. We started by testing the method without an optimization step and attained overall correct estimates with some inconsistencies between neighboring patches. To improve these results the optimization step was introduced and the optimization parameters tuned. The optimization step presents improvements over its predecessor. By enforcing consistency between neighboring patches, in the same level set and in inner/outer level sets, we obtain consistently smooth depth estimations. This consistency comes at a cost that when there is an estimation error it no longer pertains one patch, it is reflected in all of them.

In future work the inclusion of edge points in the frontier of smooth areas should be considered. Specifically, enabling the use of techniques such as Poisson blending to propagate information from the edge points to the inside of smooth areas [38]. In the same line, since human faces are mostly symmetric it would be interesting to leverage that information, as done in [55]. Lastly, since the estimation consumes about 95% of the computation time, it would be interesting to explore the application of methods to accelerate convergence. Regarding plenoptic setups an effort to investigate the potential of smartphones for light field imagery acquisition [33] would allow the general public to benefit from its capabilities. Also, it would be interesting to explore a stereo of plenoptic cameras alongside view interpolation.

Appendix A

Access Control Use Cases

The *PlenoFace* research project involves as partners Imprensa Nacional Casa da Moeda S.A. (INCM), the Institute for Systems and Robotics (ISR) poles of Lisbon and Coimbra, and the Institute for Telecommunications (IT) pole of Lisbon.

The project intends to contribute to the implementation of strong authentication directives using facial biometrics, to develop ID-doc validation methodologies, and, represent faces with plenoptic imaging to improve the effectiveness of authentication. It is then required to develop face representations and detection/authentication tools that enable secure and trustful access.

Assessment of alternatives to conventional facial biometrics systems is a priority, with special focus on smartphones, which are pointed as the "devices of choice for strong and private authentication". Specifically, setups with multiple cameras available nowadays in high-end smartphones. The ID-doc validation methodologies proposed will also be based on data acquired with smartphones combined with new watermarking methodologies and design of printer-proof security elements.

Use Cases

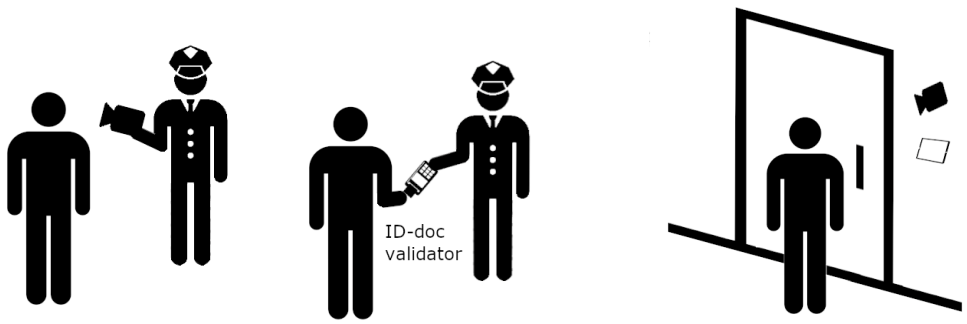
Three use cases are presented to demonstrate possible applications of this research. These are core use cases in the *PlenoFace* project proposal.

All cases are based on a plenoptic setup, an ID-doc containing biometric information, a device to perform the ID-doc validation and a CPU to perform the facial recognition. The relation to this thesis is the face detection and capture stage present in each case.

Access Control

The first use case is access control to installations (shops, companies, warehouses, etc.) resorting to visual recognition (see Fig. A.1). It is already common to find areas of buildings with restricted access, whether with security guards at the entrance or with authentication keypads for door unlocking.

The proposal is to have a initialization step where a person's ID-doc is validated and the ID-doc facial biometrics information is inserted into the local access-control database. The database enables visual recognition. In this way, future accesses can be granted without the ID-doc. Future access can be granted by: a guard equipped with a camera (Setup 1) or a camera setup at the entry point (Setup 2).

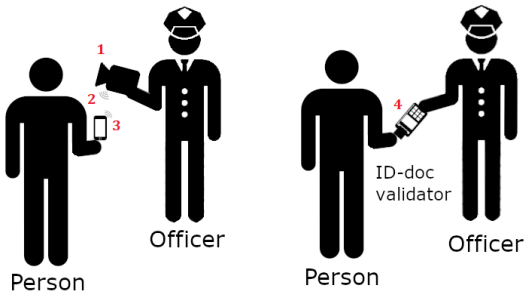


(a) Setup 1 (b) Setup 2

Figure A.1: Access control setup examples. (a) setup 1 where a guard checks the facial biometrics and validates the ID-doc. (b) setup 2 demonstrates automated access with a facial recognition setup.

Help the Officer

The second use case is proposed as a help to law enforcement. It is a combination of ID-doc validation with authentication, since ID-docs may be tampered in their security elements, particularly in the photograph, which is the most used element by officers. The proposal is, as shown in Fig. A.2, (1) to have a picture of the face taken, (2) ask the ID-doc for the facial biometrics, (3) to receive the data from the ID-doc and compare both facial biometrics, and (4) in case of some suspicion verify if the ID-doc is legitimate.



(a) Standard procedure (b) Optional step

Figure A.2: Help the officer. (a) Standard procedure: 1. take a picture of the face. 2. Ask the ID-doc for the facial biometrics information. 3. Receive the information from the ID-doc. (b) Optional step: 4. ID-doc validation.

Portable Notary

The third use case consists of multi-factor authentication, particularly, in the signature of documents. In a way, one is implementing an automated notary (see Fig. A.3).

When someone wants to buy a car, house, building or business there is a need to verify that the buyer and seller are legitimate. Many public administration services require signatures, which we propose to replace by the digital signatures. This proposal involves a multiple step process: (1) a ID-doc validation, (2) ask the ID-doc for the biometric information, (3) receive the biometrics from the ID-doc, and (4) check the received biometrics against

a picture taken. Once this process is finished, the digital signature is transferred from the ID-doc. If other parties exist, they will perform the same validation to have their signature on the document.

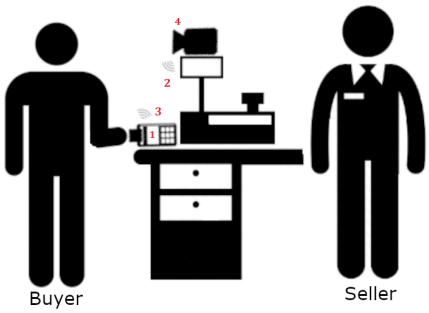


Figure A.3: Digital signature via ID-doc. First the ID-doc is validated (1), then facial biometric information is asked to the ID-doc (2), when the ID-doc sends the facial biometrics information (3) a picture is taken for comparison (4).

Appendix B

Synthetic Face Light Field

When acquiring light field images, multiple slightly different perspectives are captured. In this work after the creation of a 3D face model, resorting to virtual reality setups, a scene was composed for the acquisition of the synthetic light field. The acquisition resulted in a light field composed by 121 viewpoint images, since 11 by 11 viewpoints were used in our setup. A 5 by 5 sampling of the viewpoints was performed by choosing every other column in every other row, the result is shown in Fig. B.1.

As mentioned earlier, the distance between viewpoints is very small. Consequently, the shift in perspective is also small. When iterating over the different viewpoints in a single light field image, resorting to Dansereau's Light Field Toolbox, the shift is clear.

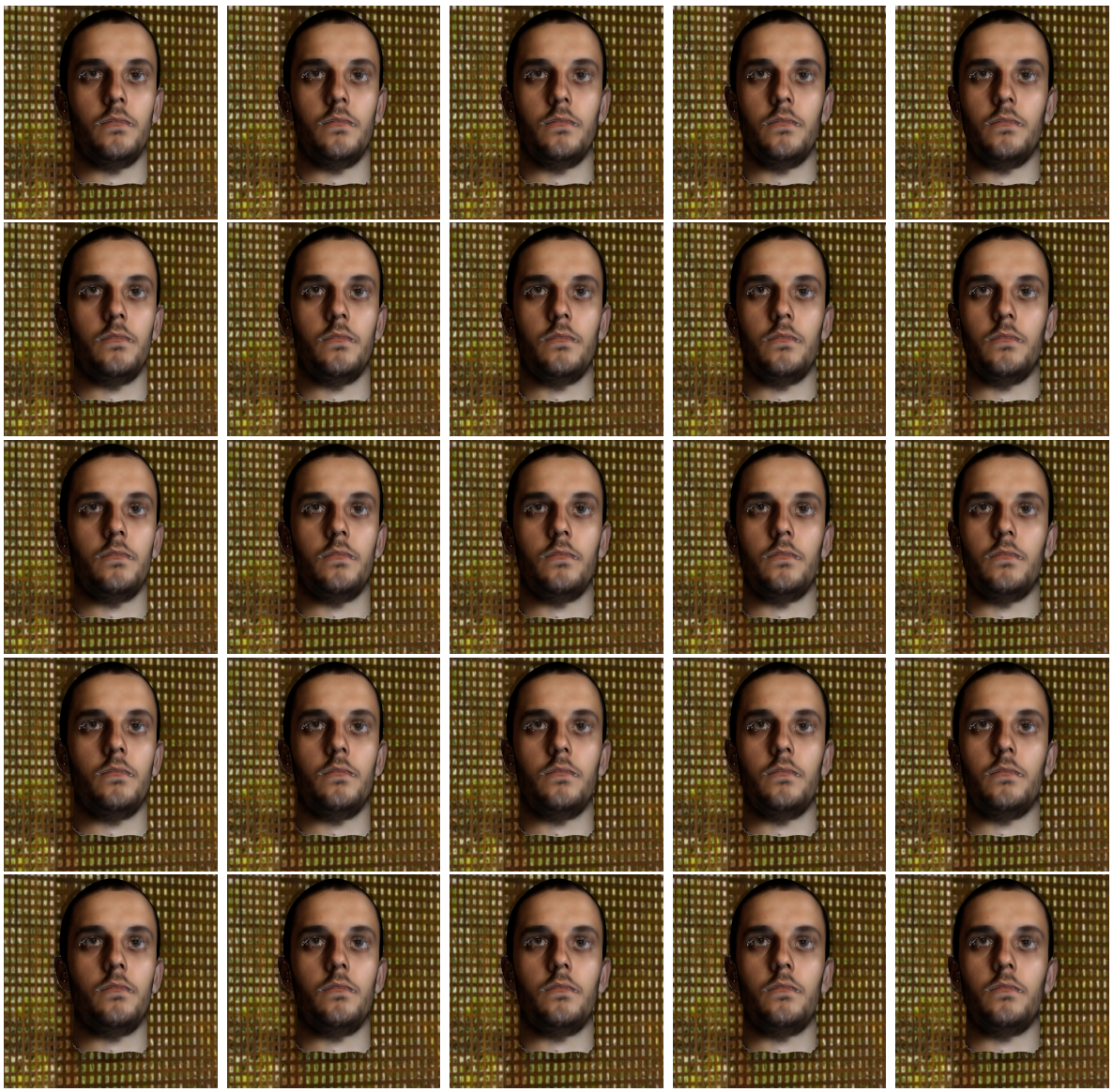


Figure B.1: Synthetic light field viewpoints: a 5 by 5 sub sampling of the 11 by 11 set of viewpoints.

Bibliography

[1] Edward H Adelson, James R Bergen, et al. *The plenoptic function and the elements of early vision*, volume 2. Vision and Modeling Group, Media Laboratory, Massachusetts Institute of Technology, 1991.

[2] Somaye Ahmadkhani and Peyman Adibi. Face recognition using supervised probabilistic principal component analysis mixture model in dimensionality reduction without loss framework. *IET Computer Vision*, 10(3):193–201, 2016.

[3] Timo Ahonen, Esa Rahtu, Ville Ojansivu, and Janne Heikkila. Recognition of blurred faces using local phase quantization. In *2008 19th Int. Conf. on Pattern Recognition*, pages 1–4. IEEE, 2008.

[4] Anna Alperovich, Ole Johannsen, Michael Strecke, and Bastian Goldluecke. Light field intrinsics with a deep encoder-decoder network. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 9145–9154, 2018.

[5] Marian Stewart Bartlett, Javier R Movellan, and Terrence J Sejnowski. Face recognition by independent component analysis. *IEEE Transactions on Neural Networks*, 13(6):1450–1464, 2002.

[6] Josef Bigun. Optimal orientation detection of linear symmetry. In *First Int. Conf. on Computer Vision*, pages 433–438. Linköping University Electronic Press, 1987.

[7] Blender Online Community. *Blender 2.79b - a 3D modelling and rendering package*. Blender Foundation, Blender Institute, Amsterdam.

[8] Robert C Bolles, H Harlyn Baker, and David H Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *Int. Journal of Computer Vision*, 1(1):7–55, 1987.

[9] J. Y. Bouguet. Camera calibration toolbox for Matlab. `http://www.vision.caltech.edu/bouguetj/calib_doc/`, 2008.

[10] Donald G Burkhard and David L Shealy. Flux density for ray propagation in geometrical optics. *JOSA*, 63(3):299–304, 1973.

[11] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.

[12] W3 Consortium. Virtual Reality Modeling Language. `http://www.w3.org/MarkUp/VRML/`.

[13] Don Dansereau and Len Bruton. Gradient-based depth estimation from 4D light fields. In *2004 IEEE Int. Symposium on Circuits and Systems (IEEE Cat. No. 04CH37512)*, volume 3, pages III–549. IEEE, 2004.

[14] Donald G Dansereau, Oscar Pizarro, and Stefan B Williams. Decoding, calibration and rectification for lenselet-based plenoptic cameras. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1027–1034, 2013.

[15] Leandro Dihl, Leandro Cruz, and Nuno Gonçalves. Exemplar Based Filtering of 2.5D Meshes of Faces. In *Eurographics (Posters)*, pages 25–26, 2018.

[16] Mehmet Dikmen. 3D face reconstruction using stereo vision. *Master's Thesis, Supervisor: Ugur Halıcı, Middle East Technical University, Ankara*, 2006.

[17] Faraz Farzin, Chuan Hou, and Anthony M Norcia. Piecing it together: infants' neural responses to face and object structure. *Journal of Vision*, 12(13):6–6, 2012.

[18] Mingtao Feng, Syed Zulqarnain Gilani, Yaonan Wang, and Ajmal Mian. 3D face reconstruction from light field images: A model-free approach. In *Proceedings of the European Conf. on Computer Vision (ECCV)*, pages 501–518, 2018.

[19] Rodrigo Ferreira, Joel Cunha, and Nuno Goncalves. Multi-focus plenoptic simulator and lens pattern mixing for dense depth map estimation. In *Proceedings of the 37th Annual Conf. of the European Association for Computer Graphics: Short Papers*, pages 37–40, 2016.

[20] Rodrigo Ferreira and Nuno Gonçalves. Accurate and fast micro lenses depth maps from a 3d point cloud in light field cameras. In *2016 23rd Int. Conf. on Pattern Recognition (ICPR)*, pages 1893–1898. IEEE, 2016.

[21] Rodrigo Ferreira and Nuno Goncalves. Fast and accurate micro lenses depth maps for multi-focus light field cameras. In *German Conf. on Pattern Recognition*, pages 309–319. Springer, 2016.

[22] Michael D Grossberg and Shree K Nayar. The raxel imaging model and ray-based calibration. *Int. Journal of Computer Vision*, 61(2):119–137, 2005.

[23] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. In *Proceedings of the 28th Annual Conf. on Computer Graphics and Interactive Techniques*, pages 327–340, 2001.

[24] Katrin Honauer, Ole Johannsen, Daniel Kondermann, and Bastian Goldluecke. A dataset and evaluation methodology for depth estimation on 4D light fields. In *Asian Conf. on Computer Vision*. Springer, 2016.

[25] Ole Johannsen, Katrin Honauer, Bastian Goldluecke, Anna Alperovich, Federica Battisti, Yunsu Bok, Michele Brizzi, Marco Carli, Gyeongmin Choe, Maximilian Diebold, et al. A taxonomy and evaluation of dense light field depth estimation algorithms. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, pages 82–99, 2017.

[26] Ole Johannsen, Antonin Sulc, and Bastian Goldluecke. What sparse light field coding reveals about scene structure. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 3262–3270, 2016.

[27] KeenTools. *FaceBuilder for Blender - a 3D face modelling and rendering package*.

[28] Numair Khan, Qian Zhang, Lucas Kasser, Henry Stone, Min H Kim, and James Tompkin. View-consistent 4D Light Field Superpixel Segmentation. In *Proceedings of the IEEE/CVF Int. Conf. on Computer Vision*, pages 7811–7819, 2019.

[29] Chaochao Lu and Xiaoou Tang. Surpassing human-level face verification performance on LFW with GaussianFace. In *Twenty-ninth AAAI Conf. on Artificial Intelligence*, 2015.

[30] Simao Graça Marto, Nuno Barroso Monteiro, Joao Pedro Barreto, and José António Gaspar. Structure from plenoptic imaging. In *2017 Joint IEEE Int. Conf. on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 338–343. IEEE, 2017.

[31] Simão Pedro da Graça Oliveira Marto, Nuno Barroso Monteiro, and José António Gaspar. Locally affine light fields as direct measurements of depth. 2018.

[32] Simão Marto. Structure reconstruction using plenoptic camera. Master's thesis, Instituto Superior Técnico, University of Lisbon, 11 2017.

[33] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019.

[34] Nuno Barroso Monteiro, Joao P Barreto, and José António Gaspar. Standard plenoptic cameras mapping to camera arrays and calibration based on DLT. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[35] Nuno Barroso Monteiro, Joao Pedro Barreto, and José Gaspar. Dense lightfield disparity estimation using total variation regularization. In *Int. Conf. on Image Analysis and Recognition*, pages 462–469. Springer, 2016.

[36] Ren Ng, Marc Levoy, Mathieu Brédif, Gene Duval, Mark Horowitz, Pat Hanrahan, et al. Light field photography with a hand-held plenoptic camera. *Computer Science Technical Report CSTR*, 2(11):1–11, 2005.

[37] Omkar M Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. 2015.

[38] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In *ACM SIGGRAPH 2003 Papers*, pages 313–318. 2003.

[39] Christian Perwass and Lennart Wietzke. Single lens 3D-camera with extended depth-of-field. In *Human Vision and Electronic Imaging XVII*, volume 8291, page 829108. Int. Society for Optics and Photonics, 2012.

[40] Diogo Portela. Deep depth from plenoptic images. Master's thesis, Instituto Superior Técnico, University of Lisbon, 10 2018.

[41] Raghavendra Ramachandra and Christoph Busch. Presentation attack detection methods for face recognition systems: A comprehensive survey. *ACM Computing Surveys (CSUR)*, 50(1):1–37, 2017.

[42] Nalini K Ratha, Jonathan H Connell, and Ruud M Bolle. An analysis of minutiae matching strength. In *Int. Conf. on Audio-and Video-Based Biometric Person Authentication*, pages 223–228. Springer, 2001.

[43] NG Ren. Digital light field photography. *Ph. D. thesis Stanford University*, 2006.

[44] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 815–823, 2015.

[45] Alireza Sepas-Moghaddam, Valeria Chiesa, Paulo Lobato Correia, Fernando Pereira, and Jean-Luc Dugelay. The IST-EURECOM light field face database. In *2017 5th Int. Workshop on Biometrics and Forensics (IWBF)*, pages 1–6. IEEE, 2017.

[46] Alireza Sepas-Moghaddam, Fernando Pereira, and Paulo Lobato Correia. Light field-based face presentation attack detection: reviewing, benchmarking and one step further. *IEEE Transactions on Information Forensics and Security*, 13(7):1696–1709, 2018.

[47] Alireza Sepas-Moghaddam, Fernando M Pereira, and Paulo Lobato Correia. Face recognition: A novel multi-level taxonomy based survey. *IET Biometrics*, 2019.

[48] Michael W Tao, Pratul P Srinivasan, Jitendra Malik, Szymon Rusinkiewicz, and Ravi Ramamoorthi. Depth from shading, defocus, and correspondence using light-field angular coherence. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1940–1948, 2015.

[49] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[50] Nannan Wang, Xinbo Gao, Dacheng Tao, Heng Yang, and Xuelong Li. Facial feature point detection: A comprehensive survey. *Neurocomputing*, 275:50–65, 2018.

[51] Sven Wanner and Bastian Goldluecke. Globally consistent depth labeling of 4D light fields. In *2012 IEEE Conf. on Computer Vision and Pattern Recognition*, pages 41–48. IEEE, 2012.

[52] Sven Wanner, Christoph Straehle, and Bastian Goldluecke. Globally consistent multi-label assignment on the ray space of 4D light fields. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1011–1018, 2013.

[53] Mika Westerlund. The emergence of deepfake technology: A review. *Technology Innovation Management Review*, 9(11), 2019.

[54] Laurenz Wiskott, Norbert Krüger, N Kuiger, and Christoph Von Der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):775–779, 1997.

[55] Shangzhe Wu, Christian Rupprecht, and Andrea Vedaldi. Unsupervised learning of probably symmetric deformable 3D objects from images in the wild. In *Proceedings of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, pages 1–10, 2020.

[56] Zhaohui Wu, Yueming Wang, and Gang Pan. 3D face recognition using local shape map. In *2004 Int. Conf. on Image Processing, 2004. ICIP'04.*, volume 3, pages 2003–2006. IEEE, 2004.

[57] Jingyi Yu, Leonard McMillan, and Steven Gortler. Surface camera (scam) light field rendering. *Int. Journal of Image and Graphics*, 4(04):605–625, 2004.

[58] Shuo Zhang, Hao Sheng, Chao Li, Jun Zhang, and Zhang Xiong. Robust depth estimation for light field via spinning parallelogram operator. *Computer Vision and Image Understanding*, 145:148–159, 2016.

[59] Hao Zhu, Qi Zhang, and Qing Wang. 4D Light Field Superpixel and Segmentation. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 6384–6392, 2017.