

Article

Card3DFace—An Application to Enhance 3D Visual Validation in ID Cards and Travel Documents

Leandro Dihl ^{1,*}, [†], , Leandro Cruz ^{1,†},  and Nuno Gonçalves ^{1,2,†}, 

¹ Institute of Systems and Robotics, University of Coimbra, 3030-194 Coimbra, Portugal; lmvacruz@isr.uc.pt (L.C.); nunogon@deec.uc.pt (N.G.)

² Portuguese Mint and Official Printing Office (INCM), 1000-042 Lisbon, Portugal

* Correspondence: leandro.dihl@isr.uc.pt

† Current address: R. Silvio Lima, 3030-194 Coimbra, Portugal.

‡ These authors contributed equally to this work.

Abstract: The identification of a person is a natural way to gain access to information or places. A face image is an essential element of visual validation. In this paper, we present the Card3DFace application, which captures a single-shot image of a person's face. After reconstructing the 3D model of the head, the application generates several images from different perspectives, which, when printed on a card with a layer of lenticular lenses, produce a 3D visualization effect of the face. The image acquisition is achieved with a regular consumer 3D camera, either using plenoptic, stereo or time-of-flight technologies. This procedure aims to assist and improve the human visual recognition of ID cards and travel documents through an affordable and fast process while simultaneously increasing their security level. The whole system pipeline is analyzed and detailed in this paper. The results of the experiments performed with polycarbonate ID cards show that this end-to-end system is able to produce cards with realistic 3D visualization effects for humans.

Keywords: 3D reconstruction; 3D face visualization; shape and surface 3D modelling; lenticular lenses



Citation: Dihl, L.; Cruz, L.; Gonçalves, N. Card3DFace—An Application to Enhance 3D Visual Validation in ID Cards and Travel Documents. *Appl. Sci.* **2021**, *11*, 8821. <https://doi.org/10.3390/app11198821>

Academic Editor: Paraskevi Nomikou

Received: 3 July 2021

Accepted: 15 September 2021

Published: 23 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent times, security issues have become fairly prominent in our daily routines, not only digitally but also visually. The necessity to authenticate an individual in order to grant access to a restricted area or task is now common practice. The identity verification that represents a confirmation that a given identity as real and that the individual claiming the identity is entitled to it has become of major importance. Generally, identity verification is required when there is a risk associated with dealing with the wrong person. The level of confidence in identity claims depends on the risk related to incorrect identification and in the liability distribution among involved parties.

The usage of a face image in authentication cards and travel documents is considered the simplest and the most common method, thus rendering it the first step in the forging of a document. Consequently, the counterfeiting techniques for forging documents have also increased and improved, especially regarding visual authentication, which represents an easier way to forge documents. The use of techniques to make this task more difficult or even impossible has attracted significant attention.

Regarding facial recognition, for instance, recent technological advances, boosted by deep learning architectures, have already demonstrated their ability to solve the problem of recognizing a person from a single photo. Systems based on architectures such as ArcFace [1] or CosFace [2], for instance, have recently presented an extremely high confidence level in predictions, even outperforming a human operator. On the other hand, the Face Recognition Vendor Tests (FRVT) 1:1 and 1:n challenges, continuously promoted by the National Institute of Standards and Technology (NIST), reveal a high variety of commercial

and non-commercial solutions for the facial recognition problem. The research topics in this area are now becoming particularly focused on the fairness of the population distribution in terms of gender, skin color, ethnicity and other characteristics of the subjects [3,4].

Furthermore, in the context of using facial recognition for security purposes, new approaches can either improve the security of the portrait photo [5] by providing an embedding of the face in the document itself or can take advantage of the three-dimensional information of faces. Particularly, in [6], it is shown that 3D facial recognition can achieve more accurate and reliable recognition results by exploring the inherently 3D shapes of faces.

Understanding the expressions of humans is also important for the issuing of ID and travel documents. In fact, document photos must be free of facial expressions to be compliant with the standards [7,8], which helps in the recognition task. In [9], an overview on this topic is presented, particularly focusing on 3D faces.

As a result, many systems currently lean towards facial recognition with liveness detection, which is designed for automatic decision systems such as airport gates or at the entrance of buildings, for instance, although there is still a strong need for physical documents such as ID cards and travel documents (sovereign issued documents) or civil ID cards for commercial and business purposes. The counterfeiting of such documents places considerable pressure on issuing entities to improve security and prevent presentation attacks.

Consequently, portrait photos in ID and travel documents have been used in different formats and secured with different elements and technologies. The main photo of these documents is usually plain, sometimes secured with hidden elements as in the IPI™ solution [10] or the Lasink™ solution [11], to name only a few of the commercial solutions adopted by the industry. Additionally, the portrait photo of the citizen is often stored in the document's chip or in a secondary photo printed with any digital transformation or even in a lenticular structure in polycarbonate cards, as in the CLI/MLI™ solution [12].

Particularly in this last element—the CLI/MLI (Changeable or Multiple Laser Image) element—the portrait photo is personalized in the polycarbonate card in a lenticular structure that creates a two-layer image effect, usually with the portrait photo in one layer and an alphanumeric string or code in the other layer that is visible from a different viewing angle.

In this paper, we are particularly interested in the elements printed in lenticular structures and in the three-dimensional visualization effect that appears when the photo is personalized on the card surface, under the lenticular structure, through the use of photos generated from a 3D model of the person's head. This set of generated images (head views) is intended to be printed on cards with a vertical lenticular structure, providing a 3D visualization effect and thereby allowing a more accurate visual face validation. Besides the improvement in visual validation, this 3D effect in the card also makes it more difficult to forge.

To build the 3D model of a face, several techniques can be used, depending on the type of cameras or images available. Although our technique can be applied to a setup with multiple cameras, for instance, our system is focused on 3D cameras, either using light-field, time-of-flight, stereo or structured light technologies, as long as the information can be obtained from a single shot. This topic is important for our systems as, in the context of ID and travel documents, the citizen's photos are usually obtained from portals or single camera setups as it is often not practical to have a multi-camera setup (thus also avoiding synchronization and alignment problems).

There is, however, a drawback concerning occlusions when using a single camera. In fact, some parts of the face are not seen, which limits the reconstruction of the 3D model. In [13], the authors present an overview of occlusion detection and restoration techniques for 3D models of faces in the recognition context.

Our application was thus named Card3DFace, and it is the outcome of an innovation project by the University of Coimbra and the Portuguese Mint and Official Printing Office

(Imprensa Nacional-Casa da Moeda—INCM), the manufacturer of ID card and travel documents in Portugal.

Most of the available systems often make use of multiple facial images, sometimes taken simultaneously, as inputs. They also tend to use complex pipelines for model building and fitting [14]. This application is aimed to be an easy and rapid way to provide a solution for the use of 3D face images in ID cards and travel documents for authentication. Since our solution is based on single image acquisition (which is much more convenient for acquisition portals and sites), it does not require very sophisticated equipment for the acquisition process. In effect, the solution relies on an affordable 3D camera. This solution is designed to be integrated into the process of producing ID cards and travel documents.

In summary, the motivation of our initial study was the research and development of an end-to-end application to produce polycarbonate ID and travel documents with visualization effects using lenticular lenses. In the current article, we present not only the filtering of 3D models, as presented in the author's previous work [15], but indeed the whole system, from the acquisition of images to the printing of the cards, by analyzing each of the pipeline phases and by discussing the available technology and options taken. This article thus seeks to be an inspiration for other engineering articles in the industry of security printing and the authentication of persons.

This paper has the following structure: in Section 2, we mention work related to the reconstruction of faces and RGB-D characteristics; Section 3 introduces the system characterization, with an explanation of the main steps of the proposed system; Section 4 demonstrates the interface and its functionalities; in Section 5, we present the results obtained with the system; finally, Section 6 sheds light on the main conclusions.

2. Related Work

As we present an end-to-end system to produce identity cards with the 3D visualization effect in this article, our main contribution is related to the way that pieces are put together to produce the expected visualization effect. This related work section is thus organized according to the most important research topics, while disregarding some of the less relevant engineering aspects.

2.1. Three-Dimensional Face Reconstruction

Three-dimensional face reconstruction is the task of reconstructing a human face from an image into a 3D form (or mesh). Over the last few decades, researchers have expended great efforts on this matter due to its challenges and its huge applicability. As a result, substantial progress has been made, which has led to novel and powerful algorithms that obtain impressive results, even in the very challenging case of reconstruction from a single RGB or RGB-D camera. According to Zollhöfer et al. [16], unlike RGB cameras, RGB-D sensors capture both color and depth data at real-time rates. This helps to solve the inherent depth ambiguity of the monocular reconstruction problem, since a coarse geometry estimate is available at least. The use of an RGB-D also enables a more reliable and realistic reconstruction of the face.

The approaches that work with RGB-D as an input with depth sensors can be classified as passive or active devices. Passive depth sensors are most commonly implemented via a stereo camera setup. If a point is found in both views of a calibrated stereo setup, the 3D point can be reconstructed through triangulation. Active cameras work with a light projector. These structured light cameras are widespread (e.g., Microsoft Kinect, Primesense Carmine, Intel Realsense) and provide relatively good depth in the near range, which is important for face-tracking tasks. Time-of-flight (ToF) cameras are another type of active depth camera. These RGB-D cameras compute depth by measuring the round-trip time of a light pulse (e.g., Creative Senz3D or Microsoft Kinect One).

In the work proposed by Thies et al. [17], one of the main contributions is a new real-time algorithm to reconstruct the high-quality facial performance of each actor in real-time from an RGB-D stream, captured in a general environment with large Lambertian

surfaces and smoothly varied lighting. The proposed method uses a parametric face model that spans a PCA space of facial identities, face poses and corresponding skin albedo. This model, which learns from real face scans, works as a statistical prior and an intermediate representation, which finally enables the photo-realistic re-rendering of the entire face.

Hsieh et al. [18] proposed an approach that unites facial tracking, segmentation and tracking model personalization from an RGB-D model. They detect dynamic occlusions caused by temporal shape and texture variations using an outlier voting scheme in superpixel space. The model demonstrates robust and high-fidelity facial tracking on a wide range of subjects with highly incomplete and largely occluded data.

Another interesting work that uses RGB-D as input was proposed by Bouaziz et al. [19]. This work demonstrates that online model building can replace user-specific training and manual calibration for the facial performance of capture systems while maintaining high tracking accuracy. It simply requires a low-cost 3D sensor and no manual assistance of any kind. The authors introduced an adaptive dynamic expression model that, in turn, combines a dynamic expression template, an identity PCA model and a parameterized deformation model in a low-dimensional representation, which is suitable for online learning.

One of the main problems with RGB-D cameras is the intrinsic noise that can affect the final result in face reconstruction. Filtering is a solution, but in some cases, it may not provide a satisfactory result. A face has some particular properties; thus, it is important that the filtering preserves them so as not to defeature the reconstruction. In this sense, we presented a filter approach based on exemplars [15].

In recent years, several filtering approaches have been proposed to process mesh geometry, denoising and smoothing meshes, targeting the output from 3D scanners or RGB-D cameras. The image processing literature considers that methods of mesh denoising may be classified as isotropic or anisotropic [20,21]. On the one hand, isotropic methods are independent of surface geometry; that is, they normally remove noise and high-frequency features together. For example, low-pass filters were some of the first models proposed to treat meshes. These filters remove high-frequency noises but also smoothen sharp features [22,23]. Isotropic methods therefore have difficulty in preserving geometric features. On the other hand, anisotropic filters are based on anisotropic geometric diffusions, which are inspired by scale space and anisotropic diffusion in image processing [24]. They are often needed to preserve features such as sharp edges and corners.

In our case, the main objective is to present a robust filtering model that keeps the information coherent and reliable even under adverse circumstances. This correction process is similar to that which is solved in texture-based synthesis techniques [20].

To model the specific deformation of 3D human faces used for novel view synthesis, which is necessary to print a lenticular card, one possible method is presented by [25]. In this method, the reconstructed visual hull from the shape-from-silhouette approach is used to refine the 3D model by iterating for photo-consistency, image contour and surface smoothness.

Furthermore, a very recent work [26] presents a stereo camera based on a new sensor that particularly explores the rotation of an image sensor and the parallax generated by the stereo pair of images. This work is also suitable for use in Card3DFace, as well as conventional consumer stereo cameras, since their small baseline can generate the 3D reconstruction of faces.

2.2. Head Model Reconstruction and Filtering

Fields such as computer graphics or geometric modeling are quite advanced regarding representation and mesh processing. Some of the wide variety of modeling techniques that represent data structures of meshes focus on faces (Face Set and Shared Vertices) [27], and others focus on edges (Winged-Edge and Half-Edge) [28].

Face-based representations are considered the simplest kind of representation and are implemented in the most common file formats, such as OFF, OBJ and STL. However, they do not provide any connectivity information for triangles; thus, edge representations

present themselves as a more complete strategy for storing a mesh geometry, such as the position of the vertices, the incident faces on an edge and the vertices that make up a face.

As for the filtering, it represents a phase where the mesh points that render the mesh irregular and defective are smoothed. Since 3D camera-acquired models are invariably noisy, the process of mesh smoothing is of great importance. There are several techniques in the literature used to obtain results for denoising [21]. Some of these techniques are based on the use of filters, such as the Gaussian filter and the Savitzky–Golay filter, among others. However, these filters are generic, which for faces that have intrinsic characteristics for the model itself may lead to distortions such as the flattening of the nose or deformation of the mouth and eyes.

3. System Building Blocks

Card3DFace is a system based on four main steps that address the acquisition, modeling, generation of face views and printing phases, as illustrated in Figure 1. The first step seeks to obtain an image of a person's face based on a single-shot image acquisition. The second phase, modeling, includes two specific steps: the reconstruction and filtering processes. This phase is responsible for transforming the acquired image into a 3D model. The generation of head views corresponds to the third phase, where perspective images of the generated 3D model are obtained and used for printing in lenticular cards, which corresponds to the fourth step.



Figure 1. The pipeline of the proposed application.

In this section, we focus on the four steps of the system pipeline separately.

3.1. Acquisition

For this application, it is necessary to use cameras that are capable of obtaining not only the visual characteristics but also the geometric information of the scene either directly or indirectly. In this case, the scene corresponds to faces positioned in front of the camera so that information related to depth can be extracted. In the development of this application, three types of cameras were considered and studied, namely plenoptic cameras (also known as light-field cameras), time-of-flight cameras and stereo cameras.

3.1.1. Studied Camera Types Technologies and Selection

Plenoptic (or light-field) cameras present a different architecture from conventional cameras. This difference lies in the fact that conventional cameras are composed of the main lens and an image sensor, whereas plenoptic cameras have a microlens array between the image sensor and the main lens of the camera. This microlens array allows the light field to be captured from various points of view, forming a 4D light field with a 2D image, so that it is possible to estimate the depth of the scene [29]. This estimate of the depth is obtained by using the redundancy created by multi-view geometry, where a 3D point is projected onto the image several times.

There are two kinds of plenoptic cameras: standard and multifocus. These two cameras were studied using the Lytro Illum camera and the Raytrix R42 camera, respectively. The difference is mainly the focal length of the lenses. The focal length of the microlenses, in a standard plenoptic chamber [30], corresponds to the distance between the image sensor and the array of microlenses; thus, all microlenses have the same focal length. Each lens contributes only one pixel value to the final image, with the resolution of this image equal to the number of microlenses. This feature drastically reduces the image resolution compared to sensor capacity; however, the computational power required to process the

image is also lower, making it suitable for compact cameras. An example of this type of camera is the Lytro Illum in Figure 2 (left). On the other hand, the multifocus plenoptic camera has the microlens array placed in front of the image sensor, where each microlens has a different focal length from its neighboring lenses, which are thus classified as different lens types. This design allows a better combination between effective resolution and depth of field size, resulting in higher-resolution images.



Figure 2. Plenoptic and time-of-flight cameras (left to right: Lytro Illum, Raytrix R42, DepthSense325, RealSense).

An example of a multifocus plenoptic camera is the R42 camera model developed by Raytrix GmbH, presented in Figure 2 (right). Since they are multifocal (there are micro lenses with three different focal lengths), these cameras have additional calibration issues. As such, there is a large number of parameters that need to be adjusted, making it disadvantageous.

Two cameras of the time-of-flight camera type were also studied: the DepthSense325 and the RealSense (Figure 2). These camera types employ time-based imaging, processing distance estimation based on the speed of light by measuring the time spent in the round trip for a light signal between the camera and the object for each image point. These cameras use an artificial light (provided by a laser or LED) to estimate this distance for each point in the image. They are also affordable and relatively easy to use. This type of camera, due to the simplicity of its use and the results presented, was selected to be included in this application.

The technology of stereo cameras was also taken into consideration for the defined purpose. A stereo camera is a type of camera that possesses two lenses, with an image sensor on each lens. This camera calculates the disparity between the two images at different positions, allowing them to simulate human binocular vision. This provides the ability to capture three-dimensional images—a process known as stereoscopy that can be used to create 3D images. The camera employed for the tests was the Stereo ZED, shown in Figure 3. For this camera, the acquired images revealed that the distance between the camera and the object was not adequate for face images, as the minimum distance for acquisition was larger than the distance used to capture a face image, hence the reconstruction process that estimates the depth parameter did not display reliable values.



Figure 3. Stereo camera: ZED.

3.1.2. Set-Up for Image Acquisition Conditions

Light represents one of the major components in the acquisition of an image, and it directly interferes with the obtained results. In order to improve image quality, which is highly important for the reconstruction of the 3D model, it is necessary to consider artificial light in the scene. Tests have been conducted that take this parameter into consideration; namely, acquisitions made with different light conditions, both artificial and natural. For the artificial light, we used two softboxes with 5500K lights, which are commonly used for photography lighting.

It is also necessary to include the background factor, which needs to be homogeneous, in accordance with the general recommendations of international standards for ID and travel documents. The use of light and a uniform background can enhance the model reconstruction owing to the fact that it decreases the existence of outliers in the reconstruction process. The tests with variations in lighting and background are presented in Figure 4, where one can easily see the importance of having good lighting and background conditions.



Figure 4. Lighting/background variations (above: natural light, below: artificial light).

3.2. Modeling

To reconstruct the model, it is necessary to obtain a mesh that represents and stores the facial data structure obtained in the acquisition. For this data storage, the volume of data needs to be taken into account as well as the ease of handling it. As the aim of this work was to obtain a mesh that represented the data structure of the acquired face, the use of an RGB-D data structure was considered an advantage. This structure stores not only the color information of each pixel (RGB) but also the depth information (D) in the scene. The RGB-D structure has shown great advances in scene reconstruction in terms of algorithmic concepts and with respect to different application scenarios [16].

Figure 5 represents the whole modeling process of one sample image. The process starts with the acquisition of the RGB image, followed by the depth map estimation, then the 3D reconstruction and the mesh construction.

This model generation is thus divided into two main steps: reconstruction and filtering.

3.2.1. Reconstruction

In the reconstruction from the initial image acquired by a camera, an RGB-D structure is generated. Information is stored regarding the 2D image (color at each point in the point cloud), as well as a point cloud according to the captured 3D scene (spatial position x, y, z). This information is used to construct the model.

The reconstruction of the face is based on the point cloud P obtained from the input device (camera). These points with a spatial position of x, y, z need to be aligned and organized into a mesh structure that can be easily manipulated and visualized to obtain the head views. The first step of reconstruction consists of a preprocessing step that is conducted to eliminate the outliers. These outliers are generated due to noises or lighting problems at the time of acquisition. The removal of these outliers is an approach based on the distance of the face to the camera and its depth. In order to obtain the position of the face and verify which points belong to it or are considered outliers, we use the facial landmark detector proposed by [31]. It estimates the location of 68 (x, y) coordinate pairs that map to facial structures on the face. These landmarks can be visualized in Figure 6.

After estimating the landmarks, we calculate a circular region C_n around the face's nose (red circle in Figure 6 (left)). The C_n center is given by the x and y coordinates of the 31° landmark calculated according to [31], and the C_n radius corresponds to the distance between the 30° and 31° points.

$$P_i = \begin{cases} 0 & d_i > mD \wedge d_i < mD + \tau \\ 1 & \text{otherwise} \end{cases} \quad (1)$$

where d_i is the depth of P_i , and τ is the threshold calculated based on the depth of a human face. After eliminating the outliers, the remaining points P are rearranged to a regular mesh. Firstly, we compute a triangular mesh using a Delaunay triangulation. Hence, we

recreate the depth map by matching a surface of the form $Z = F(X, Y)$. The inputs X and Y are 2D grid coordinates based on the coordinates contained in the vectors x and y from P . The grid is represented by the coordinates X and Y , with $length(y)$ rows and $length(x)$ columns. X , Y and Z are the new coordinates of a regular mesh that represents the 3D surface of the face.

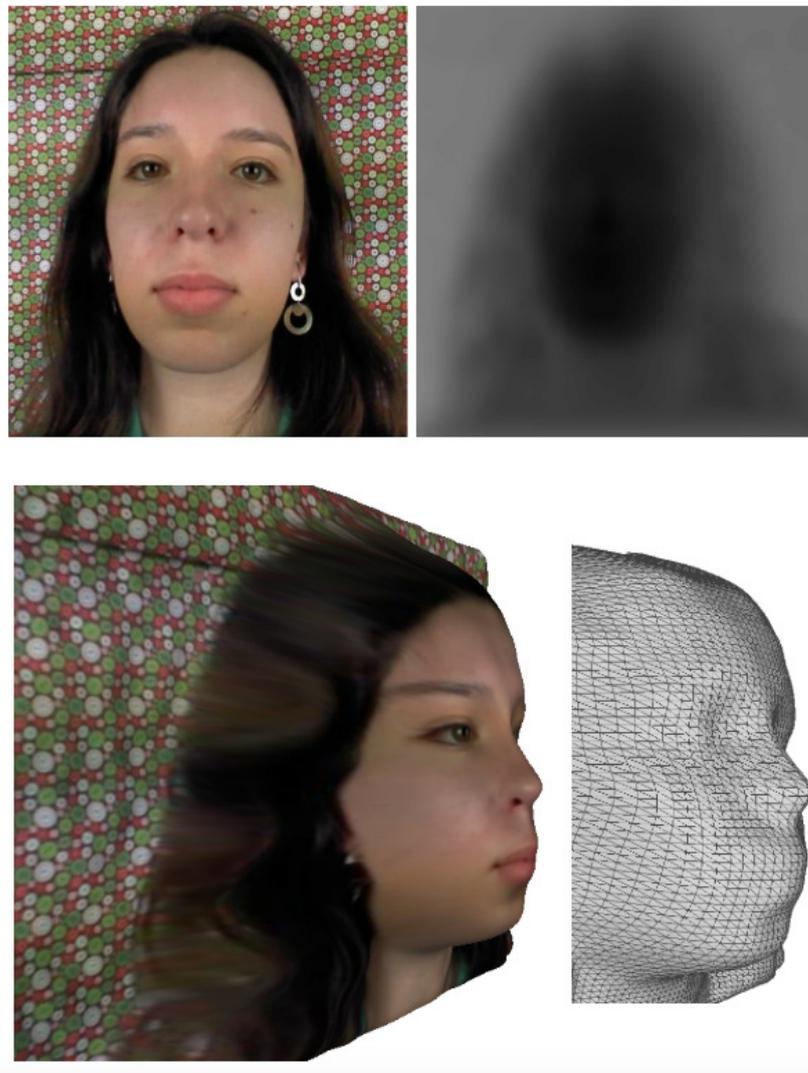


Figure 5. Representation of the modeling process for an RGB-D image. The process starts with the RGB image, passing by the depth map, 3D reconstruction and mesh.

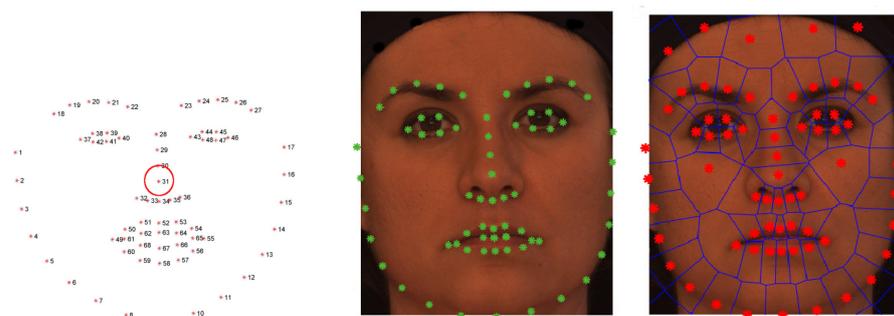


Figure 6. Estimated facial landmarks (left), landmarks on the face (center) and the Voronoi Diagram (right).

The texture of the model is mapped using *UV* mapping that projects the 2D image obtained from the photo to the 3D surface.

3.2.2. Filtering

The next step in the system pipeline is the filtering, which is necessary to smooth the noisy 3D reconstruction. We thus developed a specific filter for face meshes: a content-aware filter for RGB-D faces that proposes the smoothness of each point of a given mesh by comparing the local neighborhood using a set of exemplars [15]. This filter consists of an exemplar-based neighborhood matching, where all models are in a frontal position, avoiding rotation and perspective. We take advantage of prior knowledge of the models (faces) to improve the comparison. We first detect facial feature points, create the point correctors for regions of each feature and only use the corresponding regions to correct a point of the filtered mesh. As a result, the model is invariant to depth translation and scale. The proposed method is evaluated on a public 3D face dataset with different levels of noise. The results show that the method is able to remove noise without smoothing the sharp features of the face. Figure 7 illustrates the proposed filtering method used [15].

The filtering model comprises two main steps: the model standardization and the filtering itself. The goal of the model standardization is to allow different scales and sampling frequencies to be handled due to the distinct acquisition processes—notice that here we propose a system that can handle different types of cameras. The model standardization process consists of changing the frequency sampling of a given model, named target Π , according to a base model, named exemplar Ω . In cases when we use more than one exemplar to define the filter, one of them is chosen to be the base, and the others are also resampled. Firstly, we use a set of facial feature points, also known as facial landmarks [31] (see Figure 6, left), to align and resample the faces. After defining the landmarks of the two models to be aligned, an Iterative Close Point (ICP) [32] algorithm is then computed. This method returns a scale $s \in \mathbb{R}$, a rotation $R(2 \times 2$ matrix) and a translation $c \in \mathbb{R}^2$ that when applied over the second model aligns it with the first model, thus minimizing the difference between the two point sets. It is an affine transformation that can be represented in homogeneous coordinates by the following matrix:

$$\Gamma = \left[\begin{array}{cc|c} sR & & c \cdot x \\ & & c \cdot y \\ \hline 0 & 0 & 1 \end{array} \right]_{3 \times 3} \quad (2)$$

After the alignment, we can perform the target resampling process. This consists of creating a rectangular grid of target points, named

$\Delta = \{(x_{kl}, y_{kl}, z_{kl}, u_{kl}, v_{kl}); k = 1 \dots M, l = 1 \dots N\}$. It is performed by the definition of the coordinates *XYZ* and *UV* of $M \times N$ points (Δ dimension) taken regularly into the target texture space. The resampling is based on a triangulation of points of Π in *UV* space.

The definition of M and N (Δ dimension) is based on the target *FFPs* transformed into the exemplar texture space. Each target *FFP* (into the target texture space) is multiplied on homogeneous coordinates by Γ . An oriented bounding box is created around these transformed points. Finally, M and N are the dimensions of this box.

Once these dimensions are defined, it is necessary to create the $M \times N$ points of Δ . The resampling starts with texture (samples inside the face bounding box in target texture space), and for each sample, we need to define the respective *XYZ* and *UV* coordinates. It is also necessary to define (j, i) for each sample into target texture space and then the respective $(u, v) = \psi^{-1}(j, i)$. The coordinates (x, y, z) are obtained based on this (u, v) . We create a Delaunay triangulation of the *UV* coordinates of all points of Π and detect the triangle composed of p_a, p_b and p_c that contains (u, v) . It is noteworthy that $p_a, p_b, p_c \in \Pi$ and that they have *XYZ* and *UV* coordinates. Let $\lambda_a, \lambda_b, \lambda_c \in [0, 1]$ be the respective barycentric coordinates; then, the *XYZ* coordinates of this point are given by $(x_{j,i}, y_{j,i}, z_{j,i}) = \lambda_a(x_a, y_a, z_a) + \lambda_b(x_b, y_b, z_b) + \lambda_c(x_c, y_c, z_c)$. Therefore, this completes all coordinates of Δ points.

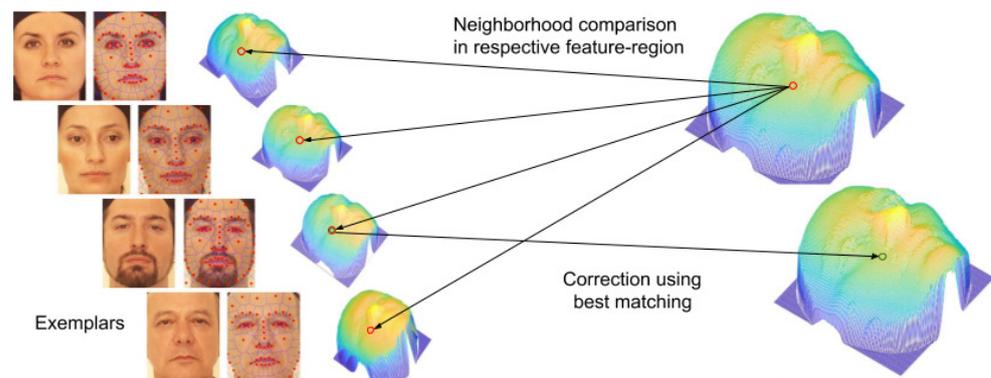


Figure 7. The pipeline of the filtering method.

After the resampling, we perform a filtering process that consists of modifying the Z value of the target points. This is achieved through a neighborhood comparison between that point and the neighborhood of equivalent points in the exemplar. In this step, both the target and sample are regular grids at the same sampling frequency. This phase is divided into two parts: the Predictor Definition and the Correction Process.

The filter is a Nearest Neighbor Predictor whose input is a neighborhood of a target point containing $k \times k$ Z -values (centered at this point), and the output is the respective normalized Z -value of the central point of the closest neighborhood from exemplars (normalization is explained below). We create one predictor per FFP region, and each one is trained by using all neighborhoods in all exemplars that belong to the respective FFP region.

The definition of the FFP regions is given by a Voronoi Diagram. For each exemplar, a Voronoi Diagram of all $FFPs$ is created (Figure 6 on the right), and for each region, in turn, all points inside it are used to train the respective predictor.

Once the points per region are defined, it is necessary to achieve a normalization of each neighborhood by subtracting the average and dividing by its variance. This guarantees that all neighborhoods can be compared, since all of them are at the same scale and depth translation.

The next step is the correction process, which consists of modifying the Δ points position according to the predictor. For each point $p \in \Delta$, (i) we determine its respective region (FFP), (ii) we obtain its neighborhood and normalize it (with the respective mean and variance), and finally (iii) we apply the projection to the base of the PCA. We use the normalized and reduced neighborhood in the prediction process. The predictor returns the normalized Z -value of the central point according to the best-matching neighborhood. Therefore, we take this value and multiply it by the variance and add it to the average of the $p \in \Delta$ neighborhood. The normalization of the exemplar and target neighborhood ensures that we can compare them irrespective of scale (division by variance) and translations in depth (subtraction by the mean). In addition, it is noteworthy that a neighborhood of the exemplar is normalized with its mean and variance, but the process of denormalization is performed by using the mean and variance of the neighborhood of the target that is being corrected. Thus, we transfer the neighborhood feature of the exemplar to the target, with the invariance mentioned above. Figure 7 illustrates this step.

3.3. Head Views

The process for generating head views consists of rotating the previously created model around a vertical axis between the eyes. Depending on the number of head views intended to be generated, a value for the angular rotation is defined for both sides of the face. The application presented here allows a selection of between 5 or 7 generated head views, corresponding to angles of 9° and 6.5° , respectively. These values were defined based on the balance between the alignment of head views and the smooth and realistic

transition between them when printed on the lenticular card. It is necessary to avoid a jump in the image when we rotate the card to visualize the 3D effect.

3.4. Lenticular Printing

The different head views generated are meant to be printed on lenticular cards, thus providing the desired 3D effect. The lenticular printing technique is a process that has been widely used to produce optical effects such as 3D perception or image flipping. Lenticular technology involves exhibiting numerous sets of images, which change when being viewed from different angles. The effect is created when the viewer sees the image from a slightly different viewpoint with each eye. The images to be viewed are sliced into strips and interlaced with each other (Figure 8).

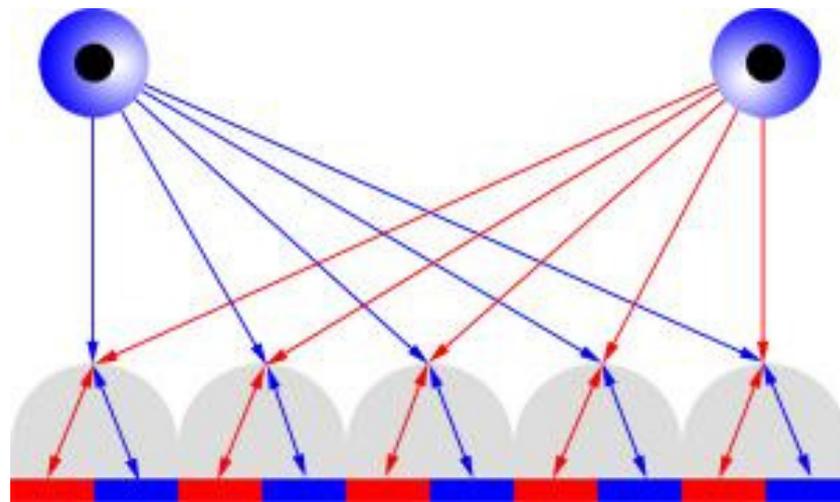


Figure 8. Lenticular lenses.

In a specific scenario, the printing process consists of the usage of a sheet of a cylindrical lens array placed on top of a high-resolution LCD in such a way that the LCD image plane is located at the focal plane of the lenses [33]. The printing techniques for lenticular lenses are still considered to be of interest in the reconstruction of a 3D image and are still used in commercial applications with advantages in terms of their low-cost and easy fabrication [34]. In our case, the printed depth in the lenticular lenses was established at 450 μm , as this depth displayed a more realistic effect of the 3D image in experiments.

4. Application Interface

The Card3DFace application was developed in the C++ language and Qt environment, which is widely used for developing multi-platform user interfaces that can run on desktops and mobile devices. This application presents four main areas that encompass the controls (1 on Figure 9) and the three principal stages of the model (2, 3 and 4 on Figure 9). It possesses a block for image acquisition, a block for the generated model and a block for rendering the head views for printing.

The controls for the application are gathered in a single ribbon at the top. In this command bar, we can select the tasks to perform; namely, uploading from a previously generated model (Figure 9a), camera acquisition (Figure 9b), filtering (Figure 9c), the selection of the number of head views (Figure 9d), the generation of the head views (Figure 9e) and file storage (Figure 9f).

There is a defined procedure to follow when running this application. Firstly, it is necessary to load a previously generated model (a) or take a picture of a person's face (b). In the next step, it is necessary to apply the filtering technique (c), and after defining the quantity of head views to be generated (d), we can proceed with the generation of head views (e). There is also the possibility of saving the obtained images (f) for later use for printing.

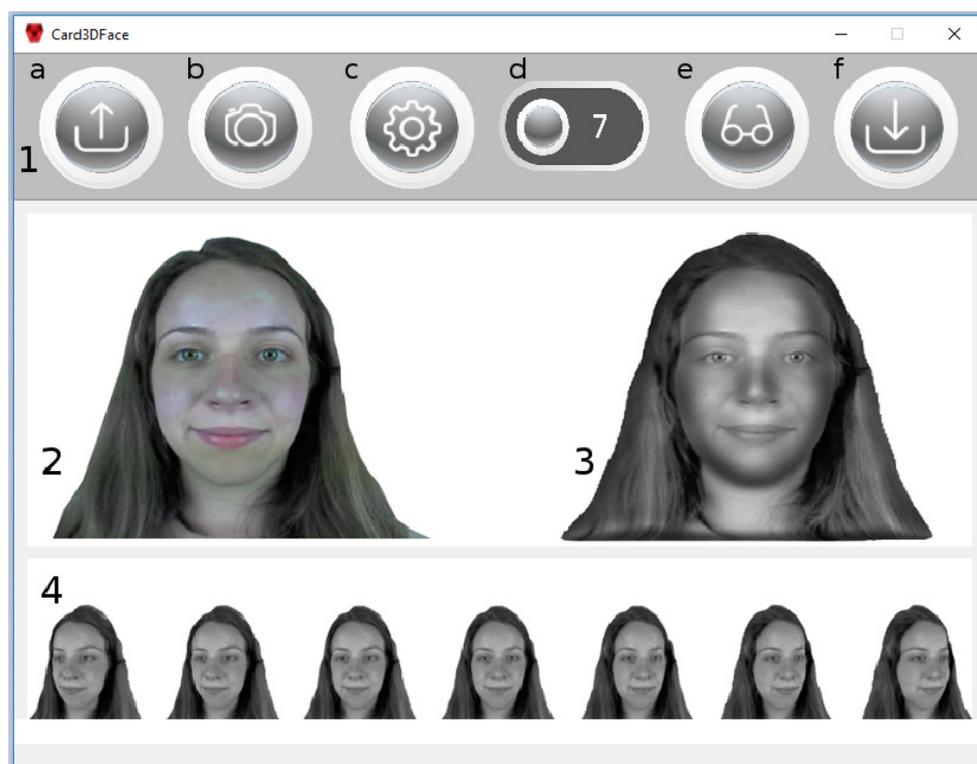


Figure 9. Application interface.

5. Experiments and Results

The experimental setup for the Card3DFace system was composed of a rig for the portrait acquisition and a polycarbonate personalization machine (printer). The acquisition rig was composed of several 3D cameras (plenoptic, stereo and time-of-flight), a photographic studio with soft-boxes and a homogeneous background and the software application. Regarding the personalization of the cards, this was achieved with a specialized printer with laser technology, which is commonly used for the personalization of Portuguese ID cards at the Portuguese Mint and Official Printing Office (INCM).

After the acquisition and modeling processes, the generated head view images could be used for printing on the lenticular cards. The calibration of the personalization machine was conducted by expert operators. This process involved the calibration of the exact position of the laser beam reaching the lenticular lenses and the exact calibration of the angle of the card holder with respect to the laser beam. This calibration guaranteed that the generated view images were personalized on the card surface below the lenticular lenses. As this calibration process depends on the specific printer and its inner design, its description does not fall within the scope of this article.

We present some examples in Figures 19–23 of the head views generated for printing, and an imprinted card prototype example is presented in Figure 25. Figure 24 shows the same card viewed from different perspectives. Notice that the printed 3D face will commonly have small dimensions and usually occupies a reduced area of the card and travel document.

Before evaluating the whole process, we describe some ablation studies made for the phases of the process.

5.1. Reconstruction Evaluation

Although the 3D reconstruction is relatively straightforward for each type of input camera, we review the cameras used and describe in greater detail the steps from the input image until the estimation of the mesh (Figures 10 and 11).

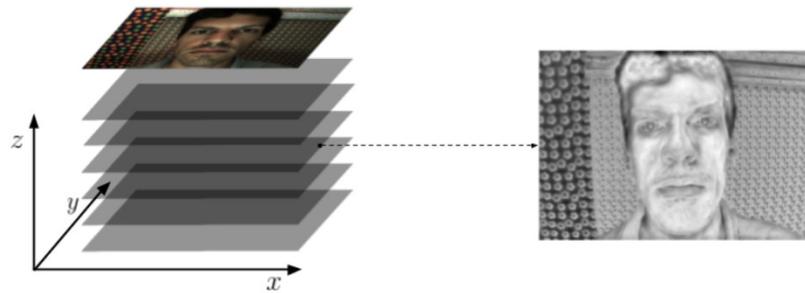


Figure 10. Representation of a cost volume, where light colors represent less probability that the points are at this depth.

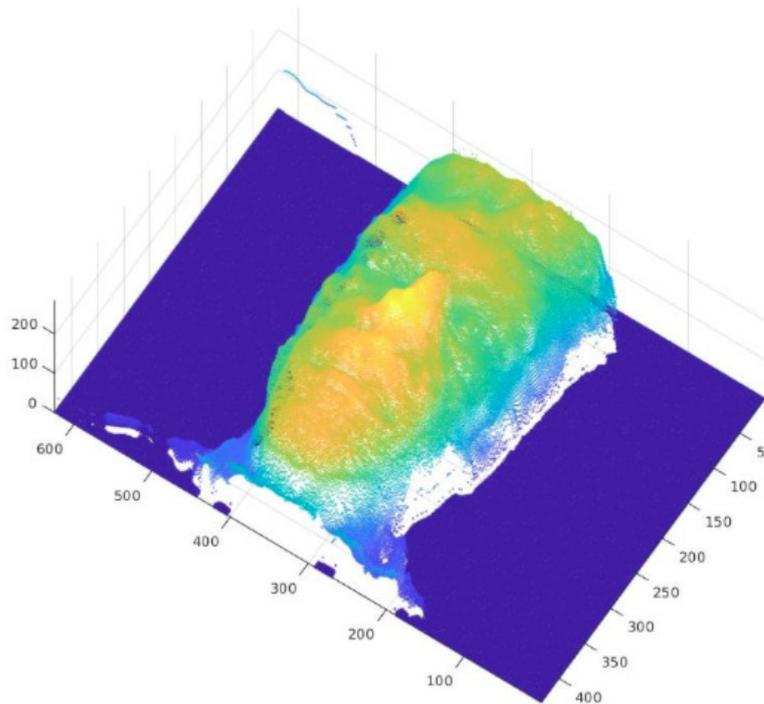


Figure 11. Resulting depth map after roughness restriction [35] and cost volume refinement [36].

5.1.1. Three-Dimensional Reconstruction Model—Lytro Illum

To estimate depth with Lytro Illum cameras, we followed one of the most popular approaches in current research: the use of cost volumes [35,36]. A cost volume is a volumetric structure in which each layer represents the cost of assigning the pixels of the image to a given depth. The lower the cost, the more likely the pixel is to be at that same depth. In a way, one can set the cost volume C to

$$C_{x,y,z} = \frac{1}{p(x,y,z)} \tag{3}$$

where p is an estimate of the probability of a coordinate in the image $\{x,y\}$ being at a given depth z . The advantage of using a cost volume instead of a “probability volume” is associated with the fact that depth estimation can be treated as a minimization problem.

For a given image, we generated the cost volume using the method [35] and its method of post-processing the volume. Then, we applied our roughness restriction to further refine the results. In this method, given an infinite-dimension C cost volume on the depth axis, z , the l depth is estimated by selecting the lowest cost layer for each pixel:

$$l_{x,y} = \operatorname{argmin}_z Cx, y, z \quad (4)$$

The roughness constraint is defined as a modification in the depth solution:

$$l'_{x,y} = \operatorname{argmin}_z C'x, y, z = \operatorname{argmin}_z Cx, y, z + \lambda R(x, y, z) \quad (5)$$

where λ is a number in the range $[0, 1]$ and $R(x, y, z)$ is the constraint function. The restriction encourages each pixel to be placed near its neighbors. Given a kernel of neighborhood N and size n , the roughness restriction is given by

$$R(x, y, z) = \sum_{i=1}^n G(d_{(x,y)+N_i}, z) \quad (6)$$

where G is the incentive function for a given depth t . Using inverted Gaussian distributions, G is defined as

$$G(t, z) = 1 - e^{-\frac{(z-t)^2}{8}} \quad (7)$$

Finally, the depth of the image was reconstructed from the final finite volume, using an existing common method—approaching parabolas. The lowest cost layer for each pixel was found, $l_{x,y}$, and then it was approached as a second degree polynomial at the cost of the $l_{x,y}$ layer and its neighboring layers $l_{x,y+1}$ and $l_{x,y-1}$. Finally, the depth of the pixel $d_{x,y}$ was set as the minimum of the polynomial. This allowed solutions to be found at points between layers, resulting in smoother and more accurate predictions.

Additionally, some cost volume refinement could be performed by applying the method of [36] in order to improve the depth results. As mentioned by the authors, the refinement in a prior stage (cost volume) is beneficial in terms of final accuracy over the refinement of the depth map.

5.1.2. Three-Dimensional Reconstruction Model—Raytrix

The methods presented by Ferreira et al. [37,38] to estimate the depth map of an image of a multi-focus plenoptic camera can be used with the Raytrix camera. This method takes advantage of the different focal lengths to perform ray tracing in order to obtain depth (back-projecting the pixels to the array of microlenses). The method begins with finding salient points and their matches in neighbor microlenses using a scaled value from the sum of absolute differences. To perform the ray tracing and to obtain more robust results, the method filters the noisy results using a RANSAC approach, eliminating unwanted results.

Raytrix cameras have additional calibration problems as they are multifocal. As such, there are several parameters that need to be adjusted. The main objective during our vision system configuration was to determine the best camera setup for estimating a depth map, including both the photographic environment and calibration parameters.

The results obtained the method of Ferreira et al. [38] are presented in Figure 12, and a 3D reconstruction example using the software of the manufacturer Raytrix is presented in Figure 13.

5.1.3. Three-Dimensional Reconstruction Model—Time of Flight

The 3D reconstruction model for images obtained by the DepthSense camera—a time-of-flight (ToF) camera—was an alternative 3D reconstruction solution to models using plenoptic cameras. In addition to presenting itself as an alternative solution, with this model, we aimed to verify the quality of 3D reconstruction using a low-cost camera. For this camera type, we opted to use the software from the manufacturer.

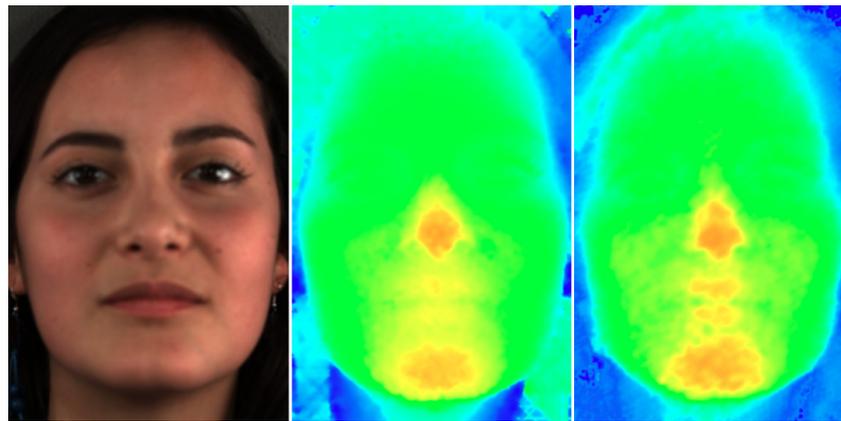


Figure 12. Resulting depth map using a Raytrix light-field camera. **(Left):** original face image; **(center):** Raytrix software output depth map; **(right)**—output depth map using [38].



Figure 13. Depth map and textured reconstruction using Raytrix software.

The image acquisition obtained by the DepthSense camera of a given scene is described in two files: a file that stores a 2D image of the scene and a file that describes a point cloud according to the captured 3D scene. These two files were used in the initial phase (input) of our proposed method: a file that stored the 3D information (x, y, z) for each point in the cloud and a file that stored the color of each point of the point cloud.

Figure 14 illustrates the data initially captured by the DepthSense camera.



Figure 14. Image and 3D reconstruction views of the point cloud estimated by the DepthSense software.

In the next step, outliers that corresponded to noisy points that had been improperly captured by the camera were removed. Based on the remaining points, a regular mesh was created with the aim of constituting the mesh of the 3D reconstruction. As previously stated, filtering was the following phase, as described in the next subsection.

5.1.4. Three-Dimensional Reconstruction Model—Stereo

As previously stated, the stereo camera used in our system was a ZED camera. However, since the work distance of this camera was not appropriate for face acquisition in a studio environment, we opted to discard these experiments.

Nonetheless, stereo cameras are generally suitable for the 3D reconstruction of scenes and consequently for the estimation of meshes of the reconstructed scene.

5.2. Filtering Evaluation

In order to evaluate the results of the filtering, we performed experiments by applying white noise [39] to the set of different models. We varied the noise intensity and compared the results with Bilateral [40] and Gaussian filters. We then calculated the mean-squared error (MSE) [41] between the noisy models and filtered models. Table 1 shows the quantitative errors on the models used in this experiment. Our method presents the lowest error compared to other filters and preserves the details of the mesh (sharpness). These sharp details can be seen in Figure 16d.

Table 1. The quantitative errors of the models used in this experiment. The first column is the noise level. Columns 2, 3 and 4 are the results of the mean-squared error (MSE) [40] between the noisy models and filtered models.

Noise level	Bilateral	Gaussian	Ours
0.0	0.6615	1.8459	0.4859
0.5	0.6782	1.8484	0.4991
0.7	0.6970	1.8595	0.5289
1.0	0.7117	1.8819	0.5383
2.0	0.8236	1.9450	0.7218

Figure 15 shows the differences between the results obtained by the tested methods. Column (a) illustrates the applied noise, while columns (b), (c) and (d) show the difference between the noisy model and the model filtered by the Bilateral and Gaussian filters and our filter, respectively. The noise is a random number in the interval $[-\alpha\sigma, \alpha\sigma]$, where σ is the standard deviation of the target model. The alpha values per row are 0, 0.5, 0.7, 1.0 and 2.0.

Figure 16 exhibits the results obtained by the tested filters. Column (a) illustrates the noisy model (α equals 0.5, 0.7, 1.0 and 2.0), while columns (b), (c) and (d) show the models filtered by the Bilateral and Gaussian filters and our filter, respectively.

Additionally, Figure 17 illustrates the correction of a mesh obtained using the Depth-Sense camera [23]. We first standardized the scale and sampling frequency with relation to the database, and then we corrected it using our filter.

Figure 18 demonstrates the results obtained by our filter on different models. Column (a) illustrates the model with texture, column (b) shows the ground truth of the mesh obtained by a 3D scanner, (c) is the same mesh after noise, (d) shows models filtered by our method without texture and (e) illustrates the textured filtered models.

The use of the Bosphorus Database [42] allowed us to use these models as a ground truth. We used 20 randomly chosen models as examples. Future work may involve the determination of the minimum amount of examples that minimize filtering error. Reducing the amount of neighborhoods per FFP region (by removing intraclass redundancy) is left for future work.

Finally, our filtering approach was based on a division of the model into regions in which all points have an intrinsic geometric similarity. We presented how to define these regions for the specific case of faces with the usage of facial features in Section 3.2.2. A future research direction would be to define general descriptors that can be used for general-purpose filtering.

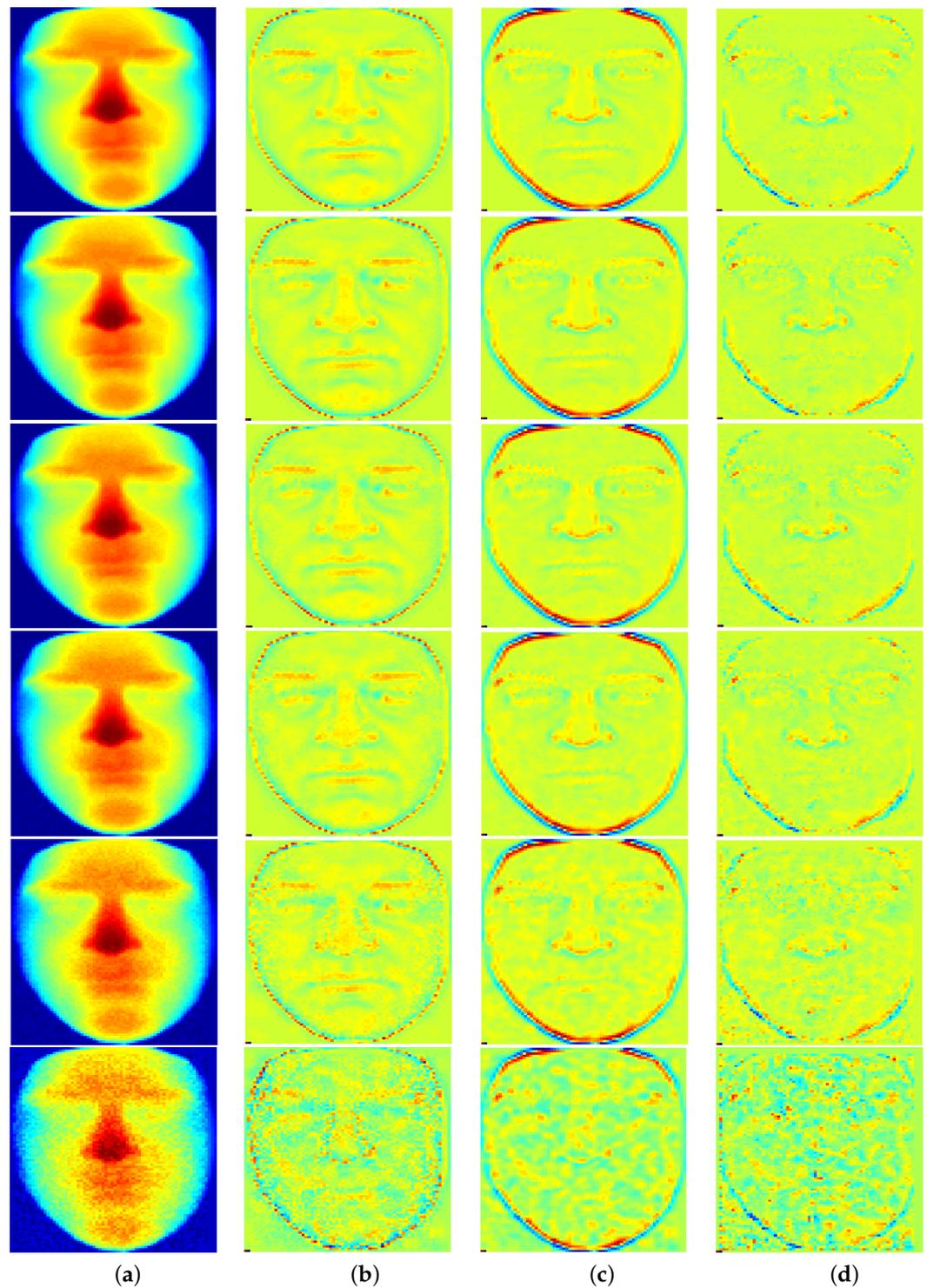


Figure 15. The differences of the results of the tested methods. Column (a) illustrates the applied noise, while columns (b–d) show the difference between the noisy model and the model filtered by the Bilateral and Gaussian filters and our filter, respectively. The noise is a random number in the interval $[-\alpha\sigma, \alpha\sigma]$, where σ is the standard deviation of the target model. The alpha values per row are 0, 0.5, 0.7, 1.0, 2.0 and 5.0.

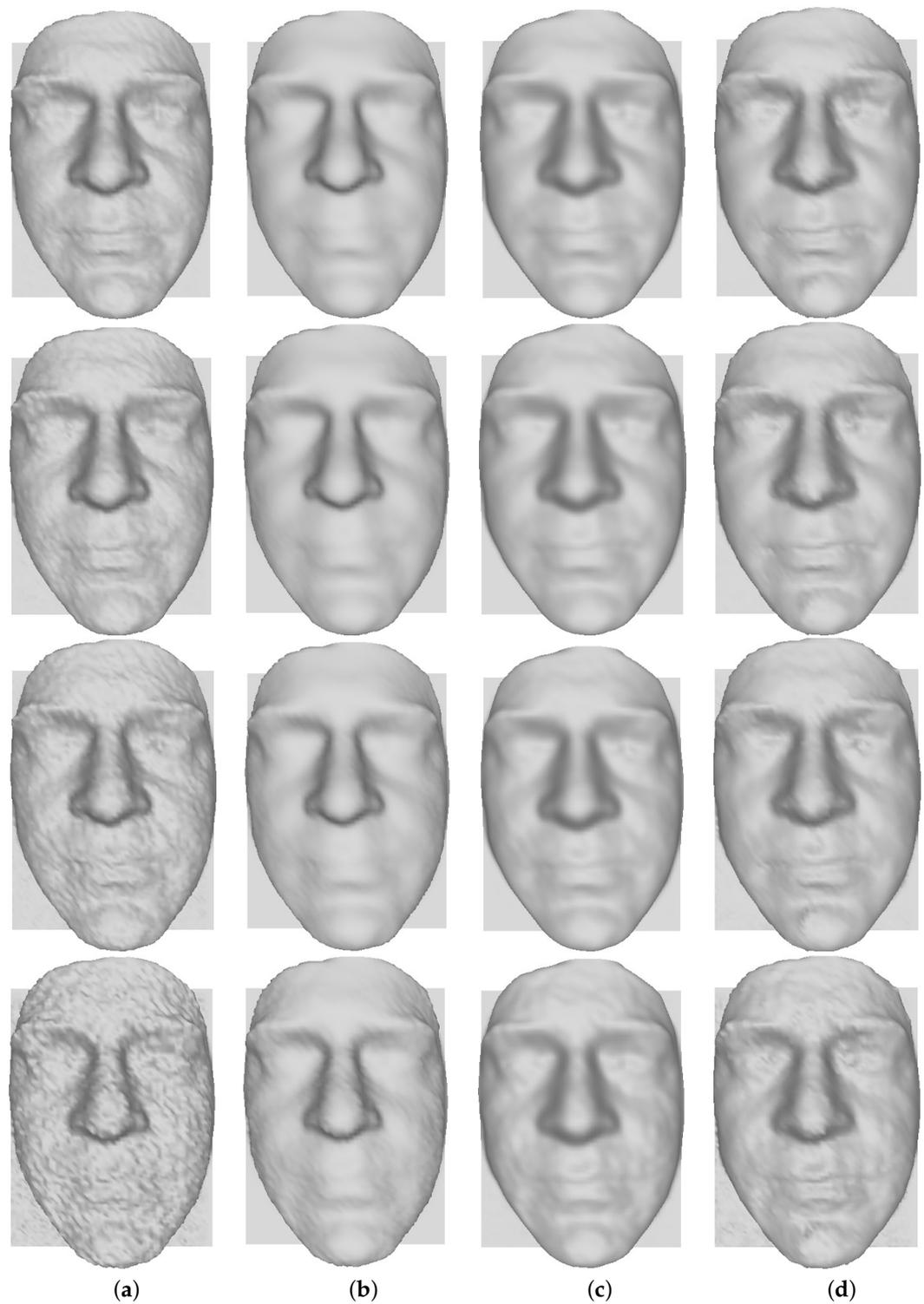


Figure 16. The results obtained by the tested filters. Each row shows the noise levels applied to the models, which are 0.5, 0.7, 1.0 and 2.0 (respectively) multiplied by standard deviation. Column (a) illustrates the noisy model, while columns (b–d) show models filtered by the Bilateral and Gaussian filters and our filter, respectively.

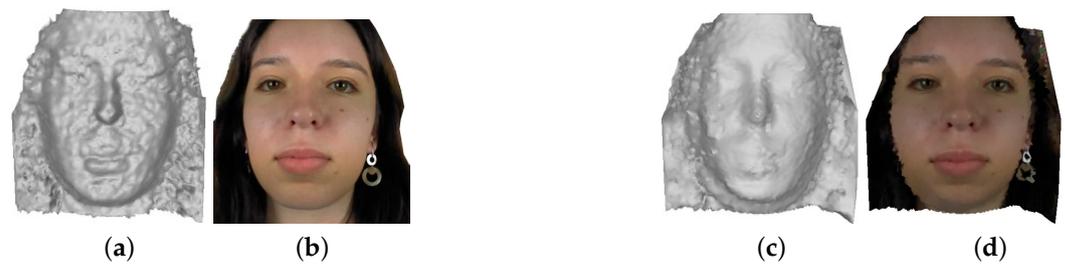


Figure 17. The correction of a mesh obtained using DepthSense camera [43]. We first standardized the scale and sampling frequency in relation to the database, and then we corrected using our filter. Figure (a) shows the acquired noisy mesh, (b) is only the texture, (c) is the filtered mesh and (d) is the filtered mesh with texture.

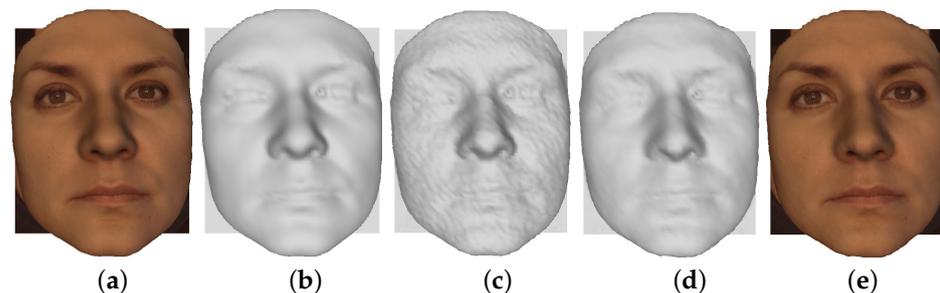


Figure 18. The results obtained by our filter on different models. Column (a) illustrates the model with texture, column (b) shows the ground truth of the mesh obtained by a 3D scanner, (c) is the same mesh after noise, (d) shows the models filtered by our method without texture and (e) shows the textured filtered models.

Datasets

As mentioned before, the 3D model dataset used for the image synthesis was the Bosphorus Database [42]. This dataset allowed us to test and improve the last two steps of the pipeline: the rendering of new views and the printing phase.

As for the first two steps of the pipeline—the acquisition and the modeling (3D reconstruction and filtering)—we used an dataset built in-house and comprising 20 persons. For the filtering phase, we also used the Bosphorus dataset to provide the database of exemplars, as described previously.

5.3. System Evaluation and Discussion

Figures 19–23 are examples of the head views. The visual inspection of the resulting cards (represented by two examples in Figures 24 and 25) clearly shows the three-dimensional visualization effect that was expected to be produced in the viewing of the cards.

The Card3DFace application offers a fast and affordable way to increase security in authentication cards and travel documents. It does not need highly sophisticated camera equipment for 3D image effect creation and does not use proprietary or sophisticated software packages, allowing for the easy implementation of this application. The results present themselves as reliable and adequate, providing a 3D effect of the face in lenticular cards through the laser printing techniques of the respective head views generated.

It is worth noting that the angle difference in the generated head views is relatively low, although it is enough to create the 3D visualization effect. This angle difference is usually up to ± 20 degrees in relation to the frontal view of the face. Higher angle differences are not allowed due to two factors: on the one hand, the most important factor is the limitation of the viewing angle to keep the visibility of the card surface by a laser beam through the lenticular structure; on the other hand, the laser personalization machines themselves have physical limitations on the rotation angle of the card, which varies from machine to machine. Additionally, the 3D model of the head is also limited as it is reconstructed

from a single frontal photo for practical reasons. Despite these limitations, the results demonstrated that a good 3D visualization effect is created in the printed cards, which validates the approach to this technology and its use to secure a person's authentication using ID and travel documents.

Furthermore, due to the fast development and latest breakthroughs in the smartphone industry—devices which are very likely to generally have time-of-flight technology in their cameras (Huawei, Apple, Honor and others) in the future—reflecting on the possibility of using this application in a smartphone as well is relevant. In future work, we should also consider improvements in adapting some techniques related to the removal of radial distortion in images.



Figure 19. Example head views generated for printing.



Figure 20. Example head views generated for printing.



Figure 21. Example head views generated for printing.



Figure 22. Example views generated for printing.



Figure 23. Example head views generated for printing.



Figure 24. Imprinted lenticular card prototype—these images show the same card viewed from different perspectives.



Figure 25. Imprinted lenticular card prototype.

6. Conclusions

In this paper, we presented an application that is capable of producing different views of a person's head and face based on an image from a single-shot acquisition. These views are meant to be printed on lenticular cards, thus providing a 3D visualization effect and a sense of depth of an individual's image.

In our view, this application presents itself as a considerable and important step to achieving a higher level of security and improving the authentication capability of documents. By improving the 3D views of images on an ID card and travel document, we are intrinsically improving authentication control.

The system requires a 3D camera (several technologies are included in this specification) that is able to output an image from which we can reconstruct a 3D model, and these cameras are now affordable and common. Considering the specific type of image effect intended (3D) and the type of card used for printing the subject image area, this technology also represents an obstacle for forgery, in addition to the printing technology, which in itself is much more complex.

Author Contributions: Conceptualization and methodology, L.D., L.C. and N.G.; software, L.D. and L.C.; investigation, data curation and writing—original draft preparation, L.D. and L.C.; writing—review and editing, L.D. and N.G.; and supervision, project administration and funding acquisition, N.G. All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to thank the Portuguese Mint and Official Printing Office (INCM) and the University of Coimbra for the support of the project Card3DFace. This work has also been supported by Fundação para a Ciência e a Tecnologia (FCT) under the project UIDB/00048/2020.

Institutional Review Board Statement: Not applicable. The study involved the usage of the human face images, which were taken from the publicly available datasets and some inhouse datasets with persons that authorized the use of their face images in the study and article dissemination.

Informed Consent Statement: The study involved the usage of the human face images, which were taken from the publicly available datasets and some inhouse datasets with persons that authorized the use of their face images in the study and article dissemination.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript. The publication of the manuscript was authorized by the board of INCM—The Portuguese Mint and Official Printing Office.

References

1. Deng, J.; Guo, J.; Xue, N.; Zafeiriou, S. Arcface: Additive angular margin loss for deep face recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4690–4699.
2. Wang, H.; Wang, Y.; Zhou, Z.; Ji, X.; Gong, D.; Zhou, J.; Li, Z.; Liu, W. Cosface: Large margin cosine loss for deep face recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5265–5274.
3. Robinson, J.P.; Livitz, G.; Henon, Y.; Qin, C.; Fu, Y.; Timoner, S. Face recognition: Too bias, or not too bias? In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 1–10.
4. Kosinski, M. Facial recognition technology can expose political orientation from naturalistic facial images. *Sci. Rep.* **2021**, *11*, 1–7. [[CrossRef](#)] [[PubMed](#)]
5. Medvedev, I.; Shadmand, F.; Cruz, L.; Gonçalves, N. Towards Facial Biometrics for ID Document Validation in Mobile Devices. *Appl. Sci.* **2021**, *11*, 6134. [[CrossRef](#)]
6. Wang, Z.; Zhang, X.; Yu, P.; Duan, W.; Zhu, D.; Cao, N. A New Face Recognition Method for Intelligent Security. *Appl. Sci.* **2020**, *10*, 852. [[CrossRef](#)]
7. Portrait Quality—Reference Facial Images for MRTD. ICAO. Available online: <https://www.icao.int/Security/FAL/TRIP/Documents/TR-PortraitQualityv1.0.pdf> (accessed on 25 August 2021).
8. Information Technology—Extensible Biometric Data Interchange Formats—Part 5: Face Image Data. ISO/IEC 39794-5:2019. Available online: <https://www.iso.org/standard/50867.html> (accessed on 25 August 2021).
9. Nonis, F.; Dagnes, N.; Marcolin, F.; Vezzetti, E. 3D Approaches and Challenges in Facial Expression Recognition Algorithms—A Literature Review. *Appl. Sci.* **2019**, *9*, 3904. [[CrossRef](#)]
10. ID Documents Solutions. Available online: <https://jura.hu/travel-id-documents-mrtd-solutions/> (accessed on 15 September 2021).
11. Preventing Forgery of Identity Documents. Available online: <https://www.idemia.com/lasink/> (accessed on 15 September 2021).
12. Security Printing Solutions: Gemalto Surface Features—The Power of a Team. Available online: <https://www.gemalto.com/govt/security-features/secure-surface-features/> (accessed on 15 September 2021).
13. Dagnes, N.; Vezzetti, E.; Marcolin, F.; Tornincasa, S. Occlusion detection and restoration techniques for 3D face recognition: A literature review. *Mach. Vis. Appl.* **2018**, *29*, 789–813. [[CrossRef](#)]
14. Jackson, A.S.; Bulat, A.; Argyriou, V.; Tzimiropoulos, G. Large pose 3d face reconstruction from a single image via direct volumetric cnn regression. In Proceedings of the International Conference on Computer Vision 2017, Venice, Italy, 22–29 October 2017.
15. Dohl, L.; Cruz, L.; Monteiro, N.; Gonçalves, N. A Content-aware Filtering for RGBD Faces. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*; SCITEPRESS—Science and Technology Publications: Prague, Czech Republic, 2019; Volume 1, pp. 270–277.
16. Zollhöfer, M.; Stotko, P.; Görlich, A.; Theobalt, C.; Nießner, M.; Klein, R.; Kolb, A. State of the Art on 3D Reconstruction with RGB-D Cameras. *Comput. Graph. Forum* **2018**, *37*, 625–652. [[CrossRef](#)]
17. Thies, J.; Zollhöfer, M.; Nießner, M.; Valgaerts, L.; Stamminger, M.; Theobalt, C. Real-time expression transfer for facial reenactment. *ACM Trans. Graph.* **2015**, *34*, 183:1–183:14. [[CrossRef](#)]
18. Hsieh, P.-L.; Ma, C.; Yu, J.; Li, H. Unconstrained realtime facial performance capture. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1675–1683.
19. Bouaziz, S.; Wang, Y.; Pauly, M. Online modeling for realtime facial animation. *ACM Trans. Graph.* **2013**, *32*, 1–10. [[CrossRef](#)]
20. Wei, L.-Y.; Lefebvre, S.; Kwatra, V.; Turk, G. State of the art in example-based texture synthesis. In *Eurographics 2009, State of the Art Report, EG-STAR*; Eurographics Association: Munich, Germany, 2009.

21. Wei, M.; Huang, J.; Xie, X.; Liu, L.; Wang, J.; Qin, J. Mesh Denoising Guided by Patch Normal Co-Filtering via Kernel Low-Rank Recovery. *IEEE Trans. Vis. Comput. Graph.* **2019**, *25*, 2910–2926. [[CrossRef](#)] [[PubMed](#)]
22. Taubin, G. A signal processing approach to fair surface design. In *SIGGRAPH '95: Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*; ACM: New York, NY, USA, 1995; pp. 351–358.
23. Desbrun, M.; Meyer, M.; Schröder, P.; Barr, A.H. Implicit fairing of irregular meshes using diffusion and curvature flow. In *SIGGRAPH '99: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*; ACM: New York, NY, USA, 1999; pp. 317–324.
24. Perona, P.; Malik, J. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.* **1990**, *12*, 629–639. [[CrossRef](#)]
25. Chia, M.H.; He, C.H.; Lin, H.Y. Novel view image synthesis based on photo-consistent 3D model deformation. *Int. J. Comput. Sci. Eng.* **2013**, *8*, 316. [[CrossRef](#)]
26. Lin, H.-Y.; Tsai, C.-L.; Tran, V.L. Depth Measurement Based on Stereo Vision with Integrated Camera Rotation. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–10. [[CrossRef](#)]
27. Luebke, D. A developer's survey of polygonal simplification algorithms. *IEEE Eng. Med. Biol. Mag.* **2001**, *21*, 24–35. [[CrossRef](#)]
28. Baumgart, B.G. A polyhedron representation for computer vision. In *AFIPS '75: Proceedings of the May 19–22, 1975, National Computer Conference and Exposition*; ACM Press: New York, NY, USA, 1975; pp. 589–596.
29. Wanner, S.; Goldluecke, B. Globally consistent depth labeling of 4d lightfields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Providence, RI, USA, 16–21 June 2012; pp. 41–48.
30. Georgiev, T.G.; Lumsdaine, A. Superresolution with Plenoptic 2.0 Cameras. In *Frontiers in Optics/Laser Science*; The Optical Society: San Jose, CA, USA, 2009; p. STuA6.
31. Kazemi, V.; Sullivan, J. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, 23–28 June 2014; pp. 1867–1874.
32. Zhang, Z. Iterative point matching for registration of free-form curves and surfaces. *Int. J. Comput. Vis.* **1994**, *13*, 119–152. [[CrossRef](#)]
33. Kovács, P.T.; Balogh, T. 3D visual experience. In *High-Quality Visual Experience: Creation, Processing and Interactivity of High-Resolution and High-Dimensional Video Signals*; Grgic, M., Kunt, M., Mrak, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2010; pp. 391–410.
34. Xie, H.; Zhao, X.; Yang, Y.; Bu, J.; Fang, Z.; Yuan, X. Cross-lenticular lens array for full parallax 3-d display with crosstalk reduction. *Sci. China Technol. Sci.* **2011**, *55*, 735–742. [[CrossRef](#)]
35. Tao, M.W.; Hadap, S.; Malik, J.; Ramamoorthi, R. Depth from combining defocus and correspondence using light-field cameras. In *Proceedings of the 2013 IEEE International Conference on Computer Vision*, Sydney, Australia, 2–8 December 2013; pp. 673–680.
36. Cardoso, J.L.; Goncalves, N.; Wimmer, M. Cost volume refinement for depth prediction. In *Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR)*, Milan, Italy, 10–15 January 2021; pp. 354–361.
37. Ferreira, R.; Gonçalves, N. Fast and accurate micro lenses depth maps for multi-focus light field cameras. In *German Conference on Pattern Recognition*; Springer Science and Business Media LLC: Cham, Switzerland, 2016; pp. 309–319.
38. Ferreira, R.; Gonçalves, N. Accurate and fast micro lenses depth maps from a 3D point cloud in light field cameras. In *Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR)*, Cancun, Mexico, 4–8 December 2016; pp. 1893–1898.
39. Boyat, A.K.; Joshi, B.K. A Review Paper: Noise Models in Digital Image Processing. *Signal Image Process. Int. J.* **2015**, *6*, 63–75. [[CrossRef](#)]
40. Tomasi, C.; Manduchi, R. Bilateral filtering for gray and color images. In *Proceedings of the Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, Copenhagen, Denmark, 28–31 May 2002.
41. Lehmann, E.; Casella, G. *Theory of Point Estimation*; Springer: Berlin/Heidelberg, Germany, 1998.
42. Savran, A.; Sankur, B. Non-rigid registration based model-free 3D facial expression recognition. *Comput. Vis. Image Underst.* **2017**, *162*, 146–165. [[CrossRef](#)]
43. Sony: Sony Depthsensing Solutions. 2019. Available online: <https://www.sony-depthsensing.com/> (accessed on 15 September 2021).