IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 1, 2020, accepted May 18, 2020, date of publication May 21, 2020, date of current version June 5, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2996563

A 3D Iris Scanner From a Single Image Using Convolutional Neural Networks

DANIEL P. BENALCAZAR[1,3], (Graduate Student Member, IEEE), JORGE E. ZAMBRANO[1,3], DIEGO BASTIAS[1,3], CLAUDIO A. PEREZ[1,3], (Senior Member, IEEE), AND KEVIN W. BOWYER[2], (Fellow, IEEE)

[1]Department of Electrical Engineering, Universidad de Chile, Santiago 8370451, Chile
[2]Department of Computer Science and Engineering, University of Notre Dame, Notre Dame, IN 46556, USA
[3]Advanced Mining Technology Center, Universidad de Chile, Santiago 8370451, Chile

Corresponding author: Claudio A. Perez (clperez@ing.uchile.cl)

This research has been funded by CONICYT through project FONDECYT 1191610, by the Department of Electrical Engineering, and Advanced Mining Technology Center (CONICYT Project AFB180004), Universidad de Chile.

ABSTRACT A 3D model of the human iris provides an additional degree of freedom in iris recognition, which could help identify people in larger databases, even when only a piece of the iris is available. Previously, we reported developing a 3D iris scanner that uses 2D images of the iris from multiple perspectives to reconstruct a 3D model of the iris. This paper focuses on the development of a 3D iris scanner from a single image by means of a Convolutional Neural Network (CNN). The method is based on a depth-estimation CNN for the 3D iris model. A dataset of 26,520 real iris images from 120 subjects, and a dataset of 72,000 synthetic iris images with their aligned depthmaps were created. With these datasets, we trained and compared the depth estimation capabilities of available CNN architectures. We analyzed the performance of our method to estimate the iris depth in multiple ways: using real step pyramid printed 3D models, comparing the results to those of a test set of synthetic images, comparing the results to those of the OCT scans from both eyes of one subject, and generating the 3D rubber sheet from the 3D iris model proving the correspondence with the resulting 2D rubber sheet and binary codes. On a preliminary test the proposed 3D rubber sheet model increased iris recognition performance by 48% with respect to the standard 2D iris code. Other contributions include assessing the scanning resolution, reducing the acquisition and processing time to produce the 3D iris model, and reducing the complexity of the image acquisition system.

INDEX TERMS 3D iris reconstruction, 3D iris scanner, biometrics, iris recognition, depth estimation.

I. INTRODUCTION

The human iris is composed of two muscle systems and a sphincter to control the amount of light entering the retina [1]. These muscular fibers, as well as the pigmentation, provide a unique texture to each iris that can be used for identification [2]. Traditionally, the texture of the iris has been analyzed using 2D images to produce accurate iris recognition [2]–[8]. However, in recent years, a 3D iris scanning method that exploits the 3D relief of the iris has been proposed [9]–[11]. This method reconstructs a 3D model of the iris surface using images from several perspectives and Structure from Motion (SfM) algorithms [12], [13]. The 3D iris model opens new frontiers for biometric applications, as well as in ophthalmology [10]. For example, the 3D iris model can potentially be used as a screening method for Closure Angle Glaucoma, a disease currently diagnosed with Optic Coherence Tomography (OCT) scans [9], [10], [14].

A method for reconstructing a 3D model of the iris surface from several images was introduced by Bastias *et al.* [9] and improved by Benalcazar *et al.* [10]. The improved method consists of the following steps: First, visible light (VL) images of the iris are captured from different perspectives. These images are acquired with a custom device that illuminates the iris with Lateral and Frontal Visible Light (LFVL) [15]. Then, a modified SfM algorithm estimates the camera pose of every image jointly with a sparse 3D model of the iris [9], [10]. Then, a dense 3D point-cloud reconstruction is performed by extracting Shi-Tomasi keypoints from each image [10], [16]. Finally, the point-cloud

The associate editor coordinating the review of this manuscript and approving it for publication was Derek Abbott.

model is converted into a mesh surface by the Screened Poisson Surface Reconstruction technique [17]. This mesh helps interpolate the depth information in areas of the iris with low texture [10]. The result is a 3D model that incorporates both depth and color information of the iris surface. The additional dimension aims to increase iris recognition accuracy particularly when the iris is occluded by eyelids, eyelashes, and reflections [9], [11]. The system recently developed by Cohen *et al.* [11] tracks fiducial points from two or more near-infrared (NIR) images of the eye to create the 3D model. They then calculate the geometric error between two 3D models as the Mean Square Error (MSE) of candidate matching points. They tested their method on a dataset of 20 irises, correctly classifying all of them.

As previously described, the 3D iris scanning method can produce a complete model of the human iris, but there are limitations to this technique. First, the SfM method requires a moving camera, which adds complexity to the system. Second, SfM was conceived to scan inanimate objects; however, the human iris can dilate from frame to frame, adding a source of distortion. This was solved by acquiring many images per position, and selecting those with a consistent dilation level [10]. This solution increases both acquisition and processing time. Third, because SfM relies on keypoints and descriptors, irises with richer texture generate more 3D points than those with fewer details. Finally, it is difficult to acquire 3D points from areas in the image that present no texture; thus the point-cloud 3D model has an uneven distribution of points in space. The mesh representation solves this issue at the expense of more processing time [10].

However, SfM is not the only method that can produce 3D scene reconstruction from 2D images. In recent years, Convolutional Neural Networks (CNN) have increased accuracy in depth prediction tasks [18]–[20]. Most of the CNNs rely on training an encoder-decoder architecture with the image of a scene as the input, and an aligned depthmap as the target [21]–[23]. As a result, the CNN learns to identify visual cues, such as perspective, that allow prediction of the depth of every object in the scene. The output depthmap captures the depth value of every pixel, even in low texture areas such as uniform color furniture or roads [23]. Therefore, the 3D model is always complete and evenly sampled regardless of the texture in the image.

The main contribution of this paper is to propose a new method to obtain a 3D model of the iris from a single image using CNNs. The method is based on a depth-estimation CNN for the 3D iris model. A dataset of real iris images from 120 subjects, and a dataset of synthetic iris images with their aligned depthmaps were created. Then, depth-estimation CNNs were trained using the real and synthetic irises [18], [19], [24], [25], and two network architectures were combined to improve performance. We analyzed the performance of our method in predicting the iris depth by using real step pyramid printed 3D models, comparing the results to those of a test set of synthetic images, comparing the results to those of the OCT scans from both eyes of one subject and generating the 3D rubber sheet from the 3D iris model, and proving the correspondence with the resulting 2D rubber sheet and binary codes. Other contributions of the proposed method include assessing the scanning resolution, reducing the acquisition and processing time for producing the 3D iris model, and reducing the complexity of the image acquisition system since the camera does not need to move to scan the iris.

II. RELATED METHODS IN DEPTH ESTIMATION USING CONVOLUTIONAL NEURAL NETWORKS

Depth estimation by a CNN can be formulated as a regression problem, in which the input is an image, and the target is the depth value of every pixel, also known as the depthmap. Eigen *et al.* [21] used a single image of an indoor scene as input, and the aligned depthmap of the same scene as the target. Such a depthmap had been acquired previously with an RGB-D camera. As a result, the CNN learned the depth of the walls and objects in indoor environments with great accuracy from their contexts [21]. Since then, several methods have been reported in the literature that have used similar training schemes and improved architectures with excellent depth estimation performance [18]–[20], [22], [23], [25].

The architecture of some depth estimation CNNs has been improved to produce more robust solutions. Eigen and Fergus [22] expanded their previous work to also predicting surface normals and labels. Laina *et al.* [23] trained a ResNet50 [26] based auto-encoder to increase accuracy. Alhashim and Wonka [18] developed DenseDepth, a DenseNet-169 based encoder with upsampling layers in the decoder to obtain high resolution depthmaps of indoor and outdoor scenes. Xu *et al.* [27] integrated Convolutional Neural Fields and a structured attention model to generate pixel precision in depth estimation. Fu *et al.* [19] developed DORN, with a space-increasing discretization strategy to recast depth estimation as an ordinal regression problem. CNNs have been trained to produce more complex methods for map reconstruction and navigation. For example, the CNN SLAM not only estimates depth from a single frame, but also integrates successive predictions of a video feed into a larger and more complete map of the environment [28]. Another deep network, FastDepth, by Wofk *et al.* [20] focused on a real time implementation for robotic navigation.

One limitation of the previously described methods is the need for a large number of aligned depthmaps for training. That is why Godard *et al.* [29] developed Monodepth, an encoder-decoder CNN that is trained with stereo images. The input of that network is the left image and generating the right image is the target. In this sense, the network has to understand the 3D geometry of the scene implicitly to perform the task. Kuznietsov *et al.* [30] combined stereo image information with sparse depthmap ground truth to produce a semi-supervised implementation. Their approach uses a small number of aligned image-depthmap pairs as ground truth in a supervised manner, along with a greater number of stereo image pairs in an unsupervised manner [30].

The latter two methods [29], [30], outperformed previous existing methods in depth estimation. However, the most recent methods, DenseDepth [18], and DORN [19] have already achieved better results.

Another solution for the limited availability of training data in depth estimation is the use of synthetic images. Tian *et al.* [31] trained detection and classification networks using a combination of real and synthetic images. In their work, CNNs trained with real and synthetic data outperformed those trained with only real images [31]. Moreover, Zheng *et al.* [25] developed a depth-estimation CNN (Translation and Task Network, T^2Net) that incorporates the use of synthetic and real images in its architecture. The T^2Net is composed of a Generative Adversarial Network (GAN) that translates synthetic images to the domain of the real ones. The task component is an encoder-decoder that then predicts depth from the translated images [25]. T^2Net achieved state-of-the-art results in widely used datasets, such as NYU-DepthV2, and KITTY [25].

Zheng *et al.* [25] analyzed various strategies for incorporating synthetic data in depth estimation tasks. As a result, they propose that the best alternative is incorporating both the translation and the task in the same training loop. In this way the GAN will learn to modify synthetic images only in their appearance while keeping the main features aligned with their depthmaps. They call it the full approach, and it had the best results among the other strategies analyzed [25].

III. METHODOLOGY

Our methodology for developing a new method to obtain the 3D model of the iris from a single image using CNNs is based on a depth-estimation CNN. First we defined the requirements of the training images so that the CNNs could infer depth information from visual cues. Then, we acquired both real and synthetic iris datasets with the desired characteristics. After that, we used our datasets to train available depth-estimation CNNs for 3D iris scanning. We then analyzed the performance of our method in predicting iris depth, and using printed 3D step pyramid models, we compared the results to those of a test set of synthetic images, compared the results to those of the OCT scans from both eyes of one subject, and generated the 3D rubber sheet from the 3D iris model demonstrating the correspondence between the resulting 2D rubber sheet and binary codes.

A. LEARNING DEPTH INFORMATION

Several visual cues provide depth information to humans. Cutting and Vishton [32] identified nine distinct mechanisms from which humans perceive depth. Occlusions indicate whether an object is behind or in front of another. The relative size of an object also indicates depth. Due to perspective, an object that is closer to a camera appears bigger than another that is farther away [33]. Similarly, the texture density of a cobble road appears to be coarser close to the viewer than farther away [32]. Binocular disparity allows triangulation to compute the distance of an object from the camera depending on how its position changes from one view to the next [12], [13], [33]. These visual cues are exploited by most SfM and CNN systems to reconstruct the precise 3D model of an object or a scene [13], [21].

Depth information of the human iris images has some particular issues that are different from those of general visual scenes. In iris images, the iris is the main object in the scene, and its size is normalized. Therefore, depth information cannot be inferred by occlusions or perspective. However, shadows cast by objects are another type of visual cue that provides depth information [32]. Elevations and craters can be identified by the shadows they cast. Similarly, in our method it is desirable to learn the relationship between the shadows on the surface of the iris, and the depth of the features that produce them.

In order for the iris features to cast shadows, a lateral source of illumination is needed. For this purpose, we developed a device with lateral and frontal lighting [15]. The device has a black frame that blocks external light sources, and has six white LEDs in front of each iris and three white LEDs on the side of each iris (LFVL illumination), to illuminate both eyes. The lateral illumination creates shadows from the relief of the iris surface, increasing the texture in the image [15]. This texture improved results in iris recognition [15]. LFVL illumination has also been used in 3D iris scanning with good results [10]. It was shown in [10] that LFVL improved the iris texture by producing more keypoints for 3D iris reconstruction. In our work, however, the use of LFVL illumination is important because shadows from iris features carry depth information.

B. REAL IRIS DATASET

The real iris dataset contains iris images with a wide range of dilation levels from 120 subjects. The study was properly approved as states the resolution No.011, on May 9, 2019, by the Ethics and Biosafety Committee for Research, Faculty of Physical and Mathematical Sciences, Universidad de Chile. Each of the 120 subjects signed a letter of consent for participating in this study. Iris images were captured under LFVL illumination using the device described in both the previous section, and in [15]. Iris images were captured in 3-second videos of pupil reaction to light changes. The pupil reaction test consisted of dark adaptation for 10 seconds, so that pupils would dilate, followed by turning on the LFVL illumination for 3 seconds. This experiment is harmless to the human eye since the LEDs used in this study are catalogued Risk Group 0-1 [34]. The maximum admissible exposure time is 10,000s for those risk groups, and our subjects were only exposed for 3 seconds [34]. The video captures how the pupil contracts from a dilated state, frame by frame, at 30 f/s (frames per second). Figure 1 shows some frames of the pupil reaction experiment for 3 subjects, while Figure 2 illustrates the evolution of the dilation level over time for one subject. The dilation level is measured as the ratio between the radii of the pupil/iris boundary (Rp) and the iris/sclera

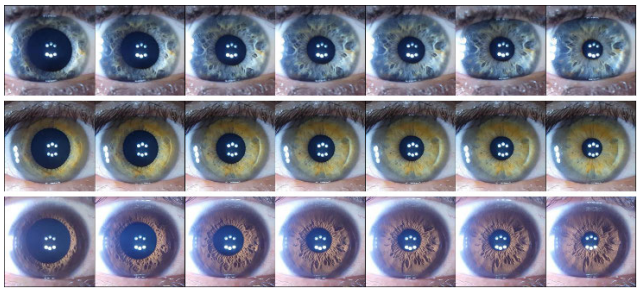


FIGURE 1. Seven frames from the pupil reaction from 3 different subjects with different pupil dilations.

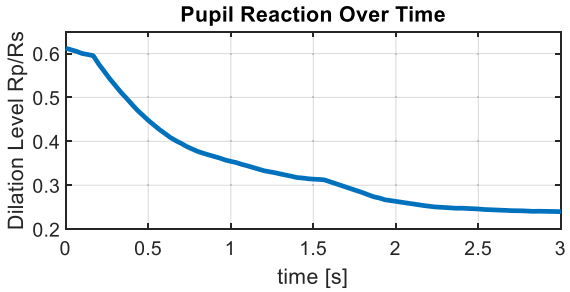


FIGURE 2. Pupil reaction experiment for the right eye of subject 003. Dilation level decreases over time in a non-linear manner due to transition from low light to brighter light.

boundary (Rs) [1], [35]:

$$\lambda = Rp/Rs. \tag{1}$$

In order to remove artifacts and normalize the number of images per subject, 60 valid frames were selected per video. At 30 f/s, each video has 90 available frames; however, some frames in the videos contained motion blur, occasioned by eye movements and blinking. Additionally, there were redundant frames with similar dilation levels, as can be seen in Figure 2 in the interval between 2.5 s and 3 s. Therefore, all images with motion blur or artifacts were removed manually, and 60 frames with different dilation levels were selected from the remaining images. The selection consisted of keeping the images with a steeper slope in the curve of Figure 2, and randomly sampling the images in the plateaus until 60 images were selected. Therefore, all the videos contain exactly 60 valid frames in the dataset. We captured two videos of pupil reaction from each eye of each subject. From the 480 videos of the 120 subjects, 38 were eliminated since the number of available frames without motion blur or artifacts was less than 60. Therefore, a total of 442 were available from the 120 subjects. The total number of iris images available was 26,520.

The dataset was acquired from 120 subjects with an average age of 23.2 $\pm$5.0 years old. Of these subjects, 67% were male and 33% were female. Of the 120 subjects, their iris colors were 48 dark brown, 49 light brown, 19 green, 3 blue, and one gray iris. The average minimum and maximum dilation levels per iris among the subjects were 0.24 and 0.54 respectively in the dataset. However, the overall minimum and maximum dilation levels were 0.16 and 0.77 respectively.

The real iris dataset was partitioned in the following manner: 96 subjects were selected randomly for training, 12 for validation, and 12 for testing. There are, therefore, 20,940 training images, 2,700 validation images, and 2,880 testing images. It is worth mentioning that we have OCT scans available of both eyes of one subject in the dataset. This subject was placed in the test set in order to assess the generalization capacity of the 3D models in comparison to OCTs of that subject. Finally, each video was captured at a resolution of 8 Mpx, and the iris diameter is 800 pixels on average. However, due to GPU limitations, we resized the iris images to a resolution of 256 $\times$ 256. The resized images are similar in size to iris images in current commercial iris sensors.

C. SYNTHETIC IRIS DATASET

In order to acquire a synthetic iris dataset we used Blender, an open-source 3D-design application [36]. Blender can produce 3D models, simulate light sources and materials, render 2D images, and produce aligned depthmaps [36]. These characteristics allowed us to simulate LFVL illumination in virtual irises. We sculpted 100 virtual irises by obtaining texture information from the real iris dataset, and depth information from 36 OCT scans gathered from the internet. Figure 3 illustrates the process of sculpting irises using Blender. In this study, we define the xy plane as the same plane used in 2D iris images, while the z axis represents depth. First, one slide from one OCT is aligned with the yz plane. Then, the iris contour is carefully traced, and a revolution surface is created by revolving the OCT slice around the z axis. The 3D texture is then added to the model so that it will resemble that of the real iris. Each of the 100 virtual irises has a different dilation level, depth profile, and texture. To illustrate, Figure 3d shows four virtual irises that come from different OCTs, and therefore have different textures and dilation levels.

We then rendered synthetic iris images from those 3D models simulating LFVL illumination [15]. Thus, all the images have illumination sources from the side, and from the front. We used the same resolution of the real iris dataset, which is 256 $\times$ 256. In the synthetic images a virtual iris of 12.1 mm in diameter was assigned 230 pixels in the image. This diameter corresponds to the average diameter of a human iris [37]. Figure 4 shows examples of synthetic images and their respective depthmaps. The shadows in a synthetic image (Figure 4) are simulated from the interactions of LFVL light with the 3D relief of virtual irises (Figure 3). Next, we used data augmentation on the 3D models rather than on the 2D images to avoid aliasing and distortions. For this purpose, we changed rotation, translation, scaling, mirroring, and color in the 3D models. We used 4 colors, 9 positions, 5 rotations, 2 scales, and mirroring, generating a total of 720 images per each virtual iris. The synthetic iris dataset therefore has 72,000 images. Since the 3D information of each model is known, the corresponding synthetic images

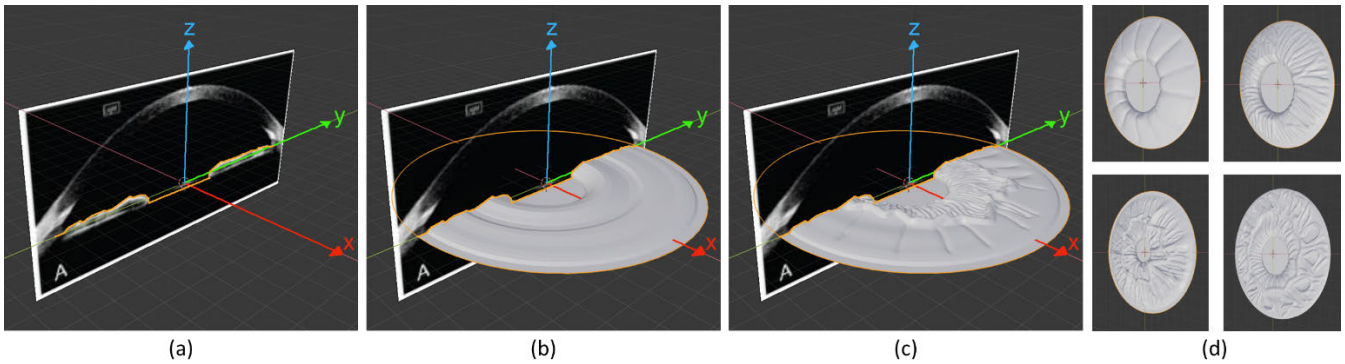


FIGURE 3. Virtual iris formation from OCT images in Blender [36]. (a) The OCT is aligned with the *yz* plane and the contour, in orange, is traced. (b) The contour of the iris is used to make a revolution surface. (c) 3D texture is sculpted in Blender. (d) Four different examples of virtual irises with texture and dilation levels to simulate the variability of those parameters found in the real iris dataset.

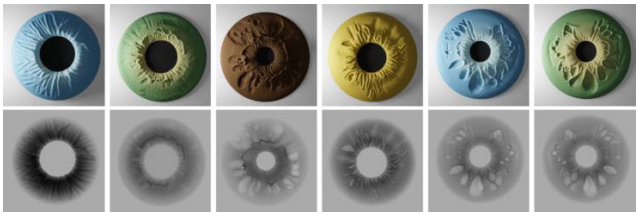


FIGURE 4. Examples of synthetic iris images, without eyelids, and their corresponding depthmaps.

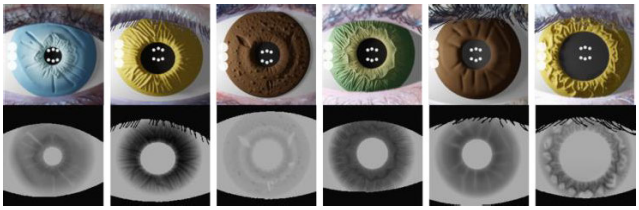


FIGURE 5. Examples of synthetic images with eyelids, eyelashes and reflections.

are accompanied by their aligned depthmaps. However, since color swapping produces the same depthmap, there are only 18,000 depthmaps in the dataset. The depthmaps were encoded using 8 bits (0-255). The scale range of 255 is equivalent to 1.936 mm in Blender for our virtual irises.

We also added eyelids, eyelashes, and reflections to the synthetic images, emulating the real iris dataset. This step also helps the networks to learn to predict depth information even in the presence of specular highlights. This will also allow the network to learn how to segment eyelids and eyelashes from the iris. Figure 5 shows the synthetic images with the characteristics described. Eyelids were given a depth value of 10 on the scale of 0-255. This number was selected to avoid saturations during training using backpropagation.

We then partitioned the synthetic iris dataset randomly, using 80 virtual irises for training, 10 for validation, and 10 for testing. We thus have 57,600 synthetic images for training, 7,200 for validation, and 7,200 for testing. The synthetic iris dataset will be available on GitHub.[1]

[1]https://github.com/dpbenalcazar/irisDepth

D. NETWORK ARCHITECTURE AND TRAINING

In this work we trained several state-of-the-art CNNs to compare their performances in solving the iris depth estimation problem. We compared DenseDepth [18], DORN [19], and T^2Net [25], those that have demonstrated great depth estimation performance in outdoor and indoor settings. We then introduce irisDepth, which combines the GAN of T^2Net with the sophisticated depth prediction architecture of DenseDepth, to increase performance. Figure 6 shows the architectures of T^2Net and irisDepth. The yellow module $G_{S\rightarrow R}$ is a GAN that is shared in both networks. In order to use iris images with these networks, we added lateral illumination (LFVL) of the iris, which enhances shadows produced by iris features [15]. Thus, LFVL illumination allows the networks to relate shadows in RGB images to depth information. The networks were then trained to relate shadows in RGB images to depth information [32]. Both real and synthetic images were illuminated with LFVL in this work.

To make use of synthetic and real data in the training process, Zheng *et al.* described two training schemes, called vanilla and full [25]. In the vanilla approach, the translation component is trained first, and the task component is trained afterwards. In the full approach, both translation and task are trained simultaneously. In the context of iris depth estimation, the translation component performs domain adaptation to the synthetic iris images to look realistic, and the task component estimates the depth value of every pixel in the iris image. We used both vanilla and full approaches to train available state-of-the-art networks for 3D iris scanning with the datasets that were described in the Methodology, subsections B and C.

For the vanilla approach, we trained CycleGAN [38], [39] to perform domain adaptation on synthetic images. We used the synthetic iris images as the input, and the real iris images as the target. We trained the network using the train partition of both datasets, and the stop epoch was determined with the validation set. After that, we used CycleGAN to translate all 72,000 of the synthetic images, and thus formed a photo-realistic iris dataset. This dataset was

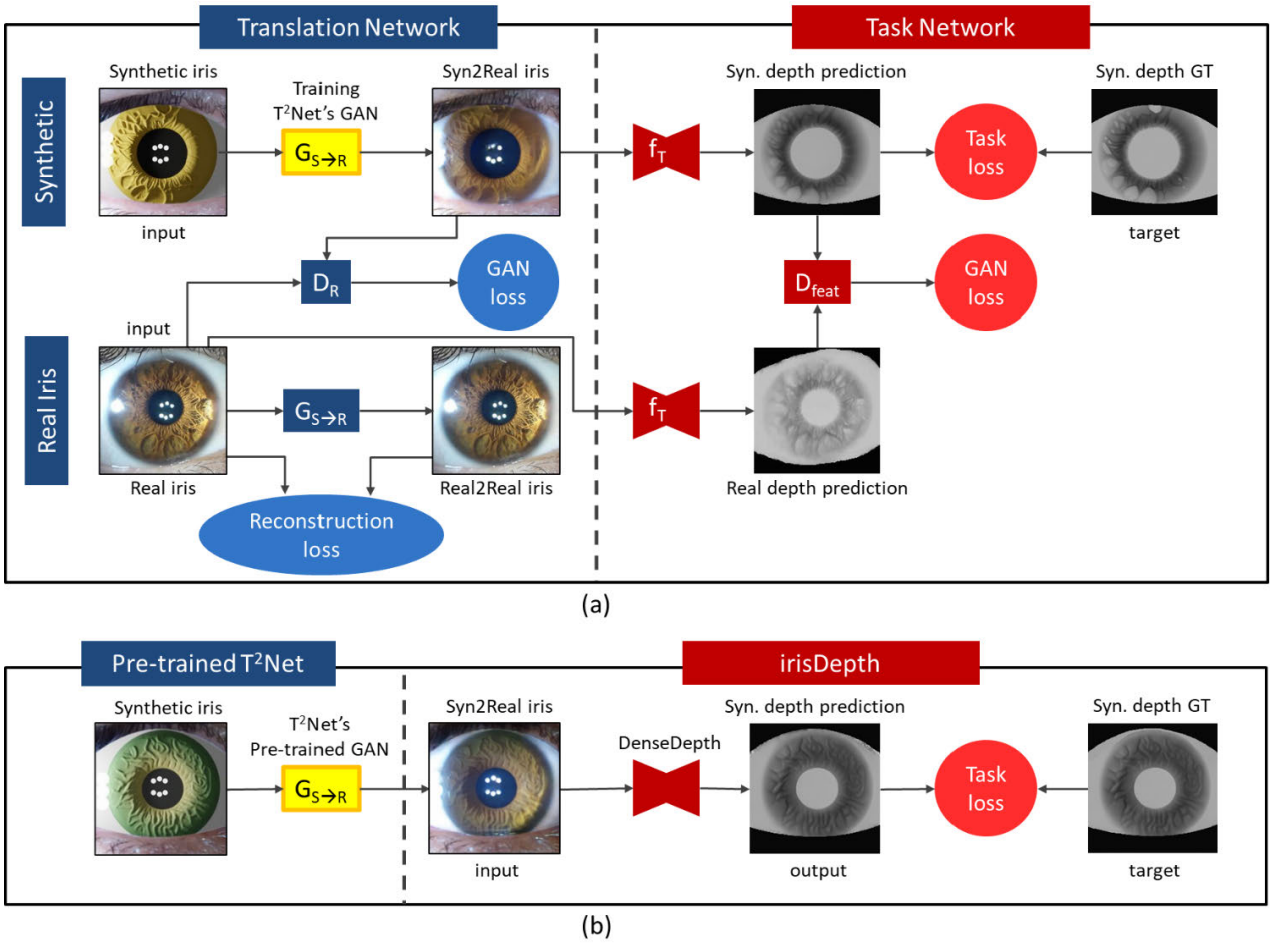


FIGURE 6. Architectures of T^2Net [25] and irisDepth in the context of iris depth estimation. (a) T^2Net consists of two parts translation, in blue, and task, in red. The translation network is comprised of a GAN that enhances the realism of synthetic images. The task part is comprised of an encoder-decoder architecture f_T, which makes depth predictions from real and translated images. (b) irisDepth uses the DenseDepth [18] architecture to improve depth prediction performance. A pre-trained T^2Net GAN enhances the realism of synthetic images while leaving iris features aligned with the corresponding depth features. After training with realistic irises with aligned depthmaps, irisDepth can make depth predictions in real iris images. The yellow module $G_{S\rightarrow R}$ is first trained in (a), and then used in (b) to generate the inputs.

partitioned identically to that of the synthetic iris dataset. Then, with the photo-realistic irises as the input, and the depth ground truth of the synthetic images as the target, we trained DenseDepth [18], DORN [19], pix2pix [24] and T^2Net [25]. In all these cases, we used the same networks available on the original code, with the exception of adjusting image sizes to 256 × 256. We used the train partition of the dataset to train these networks. The validation partition was used to determine the stop criterion for each network.

The full version of T^2Net, shown in Figure 6a, was trained using a similar procedure. We also made no changes in the network architecture other than adjusting input and output image sizes. The GAN part of T^2Net ($G_{S\rightarrow R}$) is based on SimGAN in the generator and PatchGAN in the discriminator [25]. The task network (f_T) uses ResNet-50 in the encoder and up-sampling layers in the decoder [25]. Due to GPU constraints, we had to reduce image resolution to 192 × 192 only for this network. Then, we used the train partitions of both real iris and synthetic iris datasets as the input, and the depth ground truth of the synthetic images as the target. Using the validation partition, we determined the stop epoch.

We propose a method to increase performance by merging Densedepth and T^2Netn. As Zheng *et al.* described in their paper [25], the problem with the vanilla approach is that while the GAN could morph image features in favor of better appearance, those image features might no longer be aligned with depth features in the corresponding depthmap [25]. We experienced this phenomenon with CycleGAN. As a solution to this problem, we propose using the GAN prediction of a pre-trained T^2Net along with the auto-encoder of DenseDepth, instead of using a GAN that is blind to depth information. We call this approach irisDepth, and it makes use of the precision of DenseDepth while solving the main problem of the vanilla approach. Figure 6b illustrates irisDepth's architecture.

The following steps were performed for the purpose of using irisDepth in our problem: First, we changed the configuration of T^2Net to handle images with a resolution

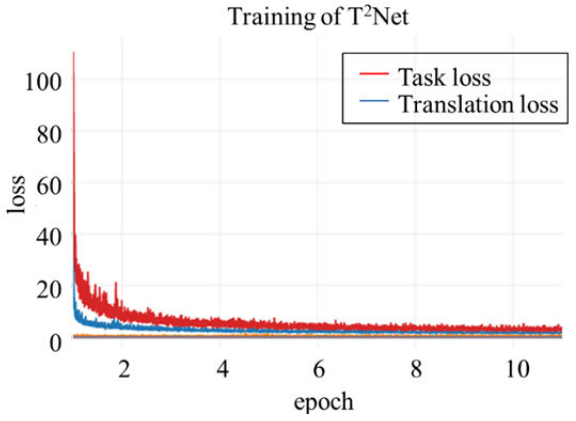


FIGURE 7. Evolution of the loss functions in the training process of T^2Net, as an example of convergence.

of 256×256. We used 6 down-sample layers in the transform network, 3 down-sample layers in the task network, 3 down-sample layers in the discriminator, and kept the rest of the parameters of the original configuration of T^2Net. Then, we trained T^2Net (Figure 6a) with our datasets, and used the validation set to find the stop point. Figure 7 shows the evolution of the translation and task loss functions. This illustrates an example of convergence with the proposed method. We then discarded the task part of this T^2Net, and used only its GAN at the best epoch for the next steps. This is the yellow $G_{S\rightarrow R}$ module in Figure 6. After that, we translated all the images in the synthetic iris dataset to obtain a realistic dataset. We partitioned this dataset to be identical to the synthetic iris dataset. Finally, we trained the standard version of DenseDepth using the train partition of the realistic dataset as the input, and the corresponding depthmaps of the original synthetic images as targets, as illustrated in Figure 6b. In this way, our irisDepth uses a GAN with information about depth data and a robust auto-encoder for the task part.

E. DEPTH EVALUATION WITH SYNTHETIC IMAGES

As one performance evaluation, we compared each network depth estimation capacity using the test set of 7,200 synthetic images. The goal of this test is to evaluate the depth estimation part of each network rather than the photo-realism of the translated images. The results of this test do not generalize to the performance on a real iris, but give a good indication of the precision of each network in the depth estimation task. First, the synthetic images were translated to the realistic domain using CycleGAN for the vanilla networks, as well as their respective GAN for the full networks. Both T^2Net and irisDepth have loss functions for the translation, as well as for the task part. Therefore, the networks perform domain adaptation instead of leaving synthetic images unchanged. Depthmaps were then predicted from the translated images using each network. Finally, we evaluated how similar the depthmaps that were predicted from the translated images were to the ground-truth depthmaps of the synthetic images. For this purpose, we used the standard metrics: Absolute Relative Difference (abs_rel), Squared Relative Difference (sq_rel), Root Mean Square Error (rmse), Logaritmic Root Mean Square Error (rmse_log), and the Accuracy Metrics (a_1, a_2 and a_3) [18]–[23], [25]. The accuracy metrics a_1, a_2 and a_3 are computed using:

$$\text{th}\,(u, v) = \max\left(\frac{\text{depth}(u, v)}{\text{GT}(u, v)}, \frac{\text{GT}(u, v)}{\text{depth}(u, v)}\right) \text{ [25]}, \quad (2)$$

$$a_n = (W \cdot H)^{-1} \cdot \sum\nolimits_{u,v} \left(\text{th}\,(u, v) < 1.25^n\right) \text{ [25]}, \quad (3)$$

where u and v are the coordinates of a pixel, depth(u,v) is the intensity of the predicted depthmap at the (u,v) coordinate, GT(u,v) is the intensity of the ground truth depthmap at the same coordinate, and $n = \{1, 2, 3\}$.

F. 3D RECONSTRUCTION OF HUMAN IRISES

After all the networks are trained and tested, they can be used to generate depth estimates on human iris images. With an iris image and the predicted depthmap we can construct a 3D model of the iris. The 3D pointcloud model consists of a list of (x,y,z) coordinates of each 3D point. The x and y coordinates come directly from scaling the position of the pixels in the image, while the z coordinate is related to the depth value. If we use u and v to describe the horizontal and vertical position of a pixel in the image, and x, y and z to describe the 3D position of a point in the point-cloud model, the coordinates of such a point in millimeters are obtained by:

$$x(u, v) = \frac{13.47}{W}\left(u - \frac{W}{2}\right), \quad (4)$$

$$y\,(u, v) = -\frac{13.47}{H}\left(v - \frac{H}{2}\right), \quad (5)$$

$$z\,(u, v) = \frac{1,936}{255}\left(255 - \text{depth}\,(u, v) + \min\left(\text{depth}\right)\right), \quad (6)$$

where W is the image width, and depth(u,v) is the intensity value of the predicted depthmap at the (u,v) coordinate. The constants in (4)–(6) depend on the size of the virtual iris and the distance to the camera. The constant 13.47 in the xy plane is computed assuming a design criterion where a virtual iris of 12.1 mm in diameter uses 230 pixels in the rendered image. Therefore, 256 pixels are 13.47 mm. The constant 1.936 mm is the maximum depth size equivalent of a variation of 255 levels in the depth map. Then, a 3D mesh model is formed by connecting neighboring points in the pointcloud. As a result, two 3D model representations are formed, and they are compatible with our previous SfM approach [10]. These models can easily be sliced and compared with OCT scans.

G. DEPTH EVALUATION WITH OCT SCANS

For one subject in the test dataset, we acquired four Anterior-Segment OCT slices of each eye, using the Visante^{TM} OCT system [40]. These 8 OCT slices provide a ground truth for the evaluation of depth estimation from real iris images. First, we normalized the scale of the OCTs and rotated them

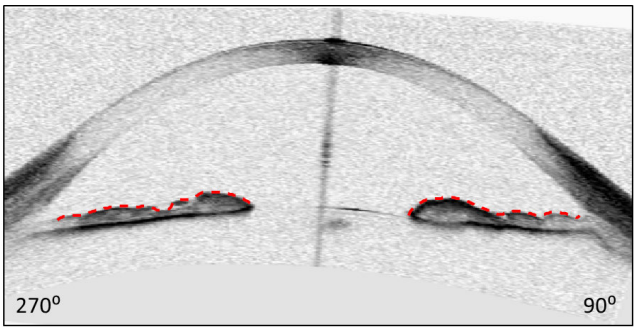


FIGURE 8. OCT edge detection of one subject in the test dataset.

so that the intersections of the cornea and the iris lay in a horizontal line. Then we used Canny edge detection to obtain the positions of the points on the iris surface. Figure 8 shows an example of the OCT with its corresponding iris surface in red. After that, one 3D model was estimated for each iris using real images of the same subject, and using the trained CNNs. We also produced one 3D model for each iris using the SfM 3D-iris-scanning method described in [10]. Then we sliced each 3D model using the same angles as in the available OCTs: 0°, 45°, 90° and 145°. To compensate for dilation differences between the OCTs and the iris images, we transformed the 3D model slices linearly to match the beginning and ending points of the irises. Finally, we compared each 3D model slice with the corresponding OCT slice, and measured the mean absolute error (MAE). The scale information on the OCT scans allowed us to calculate MAE in micrometers.

H. RESOLUTION ASSESSMENT

We assessed the minimum depth that we could detect with our method, as well as the amount of error on all three axes. For the analysis, we manufactured and scanned 3D patterns of known dimensions. We printed them in 3D real truncated pyramids of various heights, as shown in Figure 9a. The x and y dimensions of every step are fixed, and the step height ΔZ varies from 25 μm to 500 μm in increments of 25 μm. In total, we manufactured 20 pyramids for training and 5 for testing, using the FORMLABS FORM-2 stereolithography 3D printer. We set the 3D printer for the best resolution, which is 25μm per layer. Then we trained our irisDepth network with images of the 3D patterns. We used the same architecture and the same training scheme described in the Methodology, in subsection D. In this way, we used real images, as well as synthetic images with aligned depthmaps.

For the real pyramid image dataset, we used the same device and setup that was used for the iris images to assess the depth performance of our method. Figure 9b shows one image captured under these conditions as an example. We captured 360 images of the 20 real step pyramid printed 3D models, which included 6 different angles on the z axis and 3 angles on the y axis. We augmented the data using translation and scaling to produce a total of 7,200 images.

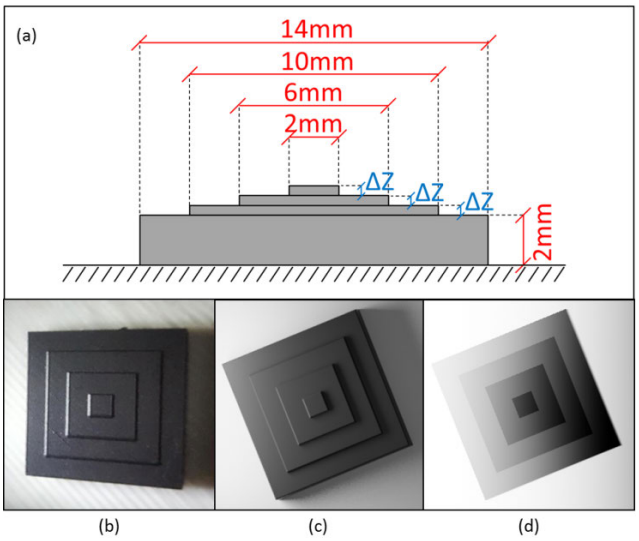


FIGURE 9. Truncated 3D pyramids for resolution assessment. (a) General shape of all 3D patterns. Red dimensions are fixed for every pyramid, while ΔZ, blue, change from model to model. Within a 3D model, all 3 steps have the same ΔZ value. (b) Example of a real image of the 3D patterns captured under LFVL illumination. (c) Example of a synthetic image of the 3D pattern, produced in Blender. (d) Depthmap of the synthetic image in (c).

For the synthetic dataset, we used Blender to create 20 virtual pyramids with similar characteristics to those of the 3D printed ones. Then, we simulated the same LFVL illumination as was used in the synthetic iris dataset. Figure 9c shows an example of a synthetic image, and Figure 9d shows its corresponding depthmap. Using 3D data augmentation, we included 45 different angles from the z axis, and 6 angles from the y axis, rendering 5,400 synthetic images with aligned depthmaps. Then, using 2D data augmentation of 6 random translations and scales, we obtained 32,400 synthetic images. Finally, we partitioned the image dataset into 80% (25,920) for training, 10% (3,240) for validation, and 10% (3,240) for testing.

We followed the same procedure for training our irisDepth network with the real and synthetic pyramid dataset as was used for real and synthetic irises. Using the trained irisDepth network, we reconstructed five 3D models from images of the real truncated pyramids, one for each of the five different heights (from 25 μm to 500 μm in increments of 25 μm). We then measured the height of each step in the reconstructed pyramids along the x and y axes. Figure 10a shows a reconstructed 3D pattern, and Figure 10b and Figure 10c show the segmented version of the 3D pattern in Figure 10a. After that, we measured the average z value, as well as the standard deviation (STD), of the 3D points that form each step. Figure 10c shows the height of each step, and the mean step size ΔZ of the 3D model. We determined the measurement errors on each axis, using the absolute difference between the measured step on the 3D model and the measured step on the ground truth. The ground truth values (ΔZ_{GT}) were measured using a Mitutoyo 293-330 micrometer on the real truncated

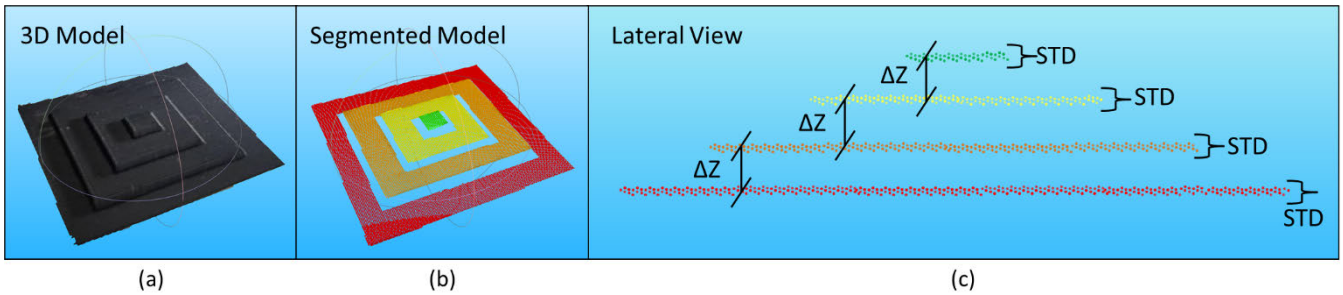


FIGURE 10. Analysis of the 3D reconstructed truncated pyramid model. (a) Example of one 3D truncated pyramid model reconstructed by irisDepth. (b) Segmentation of each step of the truncated pyramid model. (c) Description of the height estimation of each step (ΔZ) and the standard deviation of the 3D points (STD) on the truncated pyramid model.

pyramids. The precision of the ground truth measurements is given by the micrometer precision, which is $\pm 1\mu$m.

I. 3D RUBBER SHEET MODEL PROOF OF CONCEPT

As indicated in the Introduction, a 3D model of the human iris could be used in the future to improve accuracy in iris recognition. In this paper we explore a proof of concept of constructing a 3D rubber sheet from the 3D iris. Additionally, we evaluate iris recognition performance in the test set of 12 subjects.

With the purpose of building the 3D rubber sheet model, we applied a slicing procedure at regular intervals as described in the Methodology section, in sub-section F. Each slice is a 2D curve that represents the relief of the iris in a radial manner. If the radial axis of the slices is normalized between 0 and 1, the 3D rubber sheet is resilient to dilation within certain ranges, as is the case with 2D rubber sheet models. The slices, then, obtained at different angles, are concatenated linearly to form a 3D structure. We built the 3D rubber sheet of the same subject used in the OCT test. We tested the similarity of a regular rubber sheet obtained from a 2D image [3] with the flattened version of the 3D rubber sheet. We tested separately the similarity using MAE, the zero crossing normalized cross correlation (ZNCC) [41], as well as with the Hamming Distance (HD) [2] of the iris codes from both rubber sheets. A close similarity would indicate that our 3D models contain the same information on the xy plane as a 2D iris image; but, we would have additional information available on the z axis to be exploited.

A preliminary 3D iris recognition method was implemented using a 3D rubber sheet model to extract 3D keypoints and descriptors, and to compare their distances. For this purpose, we constructed 480 3D rubber sheet models using 20 images per eye of the 12 subjects in the test set. We enrolled the 20 images with the dilation level closest to the median value of the subject, as recommended by Ortiz *et al.* [42]. We constructed the 3D rubber sheet models using 75 samples on the radial axis, and 360 slices on the angular axis. Our 3D rubber sheets, therefore, contain $75 \times 360 = 27,000$ 3D points. Our proposed method for iris recognition in 3D has the following steps: First, we sample the 3D rubber sheet model with a 4×15 grid to find 60 keypoints. Then, we obtain the Spin Image descriptor [43], [44] for each keypoint. Finally, we assess the similarity of two 3D rubber sheet models as the average ZNCC [41] between corresponding Spin Images on the sampling grid. As with the 2D iris code, we account for small angular displacements by translating the 3D rubber sheet $\pm 5°$ and storing the best result [2].

TABLE 1. Designed and measured step sizes as well as estimated errors on the x, y and z axes, for the 3D truncated pyramids.

ΔZ_{Design} [μm]	ΔZ_{GT} [μm]	ΔZ [μm]	STD [μm]	Z_{err} [μm]	X_{err} [μm]	Y_{err} [μm]
25	29.6	31.9	18.4	2.3	36.7	33.3
50	55.9	55.9	16.6	3.0	24.3	46.8
100	101.9	98.0	17.6	3.9	59.9	49.8
200	206.2	202.2	17.3	4.0	46.3	47.9
400	402.4	395.3	18.6	7.1	40.4	42.8
		Average:	**17.7**	**4.1**	**42.0**	**43.6**

We compared the iris recognition performance of our 3D proposed method with that of the 2D iris code. For this purpose, we obtained the 2D rubber sheets and iris codes of the 480 images in the test set using Osiris V4.1 [45]. We then used the d' index to score iris recognition performance [2]. This index shows how well we can separate intra-class from inter-class comparisons, and it is computed using:

$$d' = \frac{|\mu_1 - \mu_2|}{\sqrt{0.5 * (\sigma_1^2 + \sigma_2^2)}} [2], \tag{7}$$

where μ_1 and μ_2 are the mean values of the intra-class and inter-class distributions, respectively, and σ_1 and σ_2 are the standard deviations (STD) of both distributions. The higher the d' value, the easier it is to separate intra-class from inter-class distributions.

IV. RESULTS AND ANALYSIS

A. RESOLUTION ASSESSMENT

The results on the 3D real truncated pyramids of different step sizes are as follows: Table 1 shows the five sizes for the 3D printed pyramids of the test set with a designed step

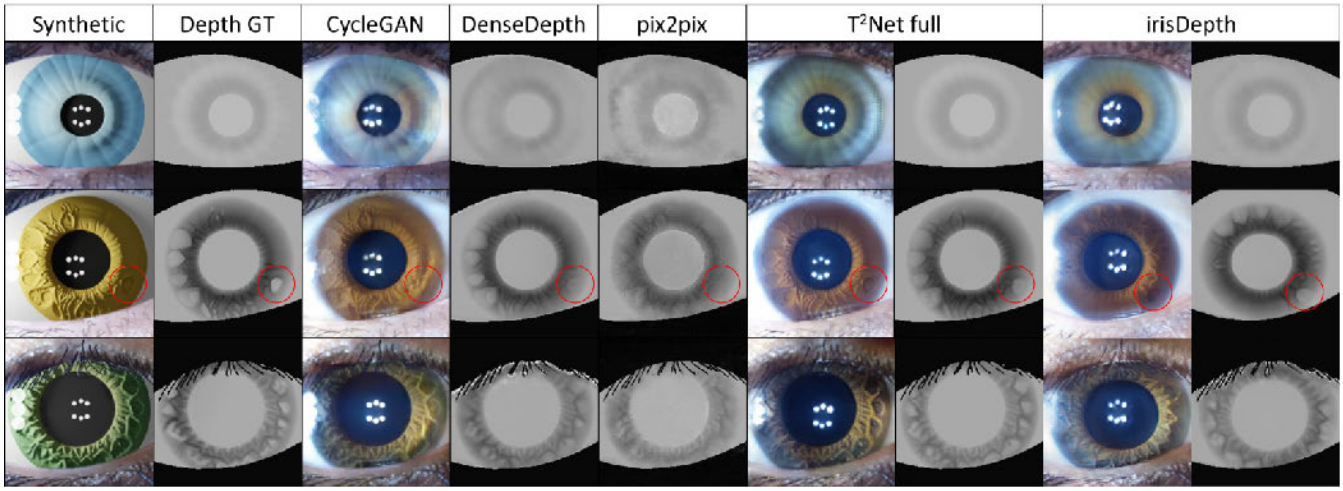


FIGURE 11. Examples of depth estimation using synthetic images. Each row is a different example. The first two columns are synthetic images and their corresponding ground truth depthmap (Depth GT). The succeeding columns show the outputs of each network. DenseDepth and pix2pix make depth predictions from the synthetic images translated by CycleGAN. T^{2}Net and irisDepth make depth predictions from the results of their own GANs. The second row shows a red circle highlighting an iris feature that can be followed into the corresponding depthmaps.

size (ΔZ_{Design}) of 25 μm, 50 μm, 100 μm, 200 μm, and 400 μm. The values of the step size measured with the micrometer are the ground truth for our depth measurements (ΔZ_{GT}); the mean step sizes measured in the 3D reconstructions (ΔZ); the standard deviation of the 3D points that form each step (STD); as well as the absolute errors measured along each z (Z_{err}), x (X_{err}), and y (Y_{err}) axis.

The results of Table 1 show that the measured step size ΔZ is close to the ground truth value (ΔZ_{GT}) for all five 3D patterns. The average absolute error on the z axis is 4.1 μm. The standard deviation represents how much the 3D points deviate from a perfect plane [10]. Its average value is 17.7 μm. This means that a feature on the z axis that is smaller than 17.7 μm is within the noise level of the 3D points. Features larger than 17.7 μm, however, can be detected by our system. Therefore, the resolution limit of our method is 17.7 μm. This figure is about 1/30th of the iris thickness [37]. Additionally, the resolution limit of 17.7 μm is almost twice as high as the 10μm of conventional OCT scans, as well as the 11 μm reported in [10] for SfM. Our results show a reasonable level of precision from a single 256 $\times$ 256 image.

The scale values on the OCT scans, as well as equations (2), (3) and (4) allow estimating the theoretical resolution of our method. According to (2) and (3), a variation of 1 mm on the x or y axis produces a variation of 19 pixels for the 256 $\times$ 256 images. Therefore, the resolution of the 3D model on the xy plane is 52.6 μm/px. This figure is around 1/230th of the iris diameter [37] and can be improved by increasing image resolution. For instance, if we used 800 $\times$ 800 images, equations (2) and (3) yield a resolution of 16.8 μm/px. A variation of 1mm on the z axis produces a depth change of 132 on the depth scale between 0 and 255. Therefore, the resolution on the z axis is 7.56 μm. Measurements are therefore 7 times more precise along the z axis than on the xy plane. These figures roughly match those shown in the experimental results of Table 1, where there is almost 10 times more error along the xy plane than on the z axis.

B. DEPTH EVALUATION WITH SYNTHETIC IMAGES

This test illustrates the precision of each network in the depth estimation task. The ground truth in this experiment comes from the depthmaps in the synthetic iris dataset, while the inputs are translated images. Figure 11 shows examples of ground truth synthetic images in the test set, results of translated images, and the network predicted depthmaps. The vanilla networks, such as DenseDepth, DORN, and pix2pix, make up a depth estimation from the photorealistic images produced by CycleGAN. T^{2}Net and irisDepth make depth estimations from the output of their own GANs. Figure 11 also illustrates the problem of training a GAN blindly from depth estimation. The ground truth example in the second row has a concave feature highlighted with a red circle. Since this feature is not reproduced by CycleGAN, neither DenseDepth nor pix2pix can estimate its depth. However, the GANs trained in the full approach learn to reproduce this feature. Both T^{2}Net and irisDepth were able to estimate the depth of this concave feature correctly.

The results of the depth evaluation with the 7,200 synthetic images in the test set are presented in Table 2. For abs_rel, sq_rel, rmse, and rmse_log metrics, a lower value means a better result, while for a_1, a_2 and a_3, a higher value is better [18]–[20]. The accuracy metrics a_n are computed using (2)-(3). The best result of each column was highlighted in bold. Table 2 shows that irisDepth produced the best results on almost all the tests. DenseDepth and DORN also produced good results due to their specialized architectures in depth prediction tasks. IrisDepth produced the best overall results since it combines a GAN that has information on

TABLE 2. Similarity using standard metrics between depthmaps predicted from the translated images and depthmaps of the synthetic images in the test dataset of 7,200 images.

Method	abs_rel	sq_rel	rmse	rmse_log	a_1	a_2	a_3
DenseDepth [18]	0.0520	1.468	9.876	0.1150	0.9651	0.9952	0.9976
DORN [19]	0.0525	1.775	11.893	0.1023	0.9714	0.9895	0.9931
pix2pix [24]	0.2219	9.498	16.039	0.4360	0.7302	0.8630	0.9174
T^2Net_vanilla [25]	0.0715	2.972	8.649	0.1849	0.9567	0.9847	0.9881
T^2Net_full [25]	0.0576	1.863	**7.350**	0.1440	0.9619	0.9903	0.9931
irisDepth (ours)	**0.0475**	**1.161**	8.878	**0.1105**	**0.9728**	**0.9958**	**0.9978**
	Lower is better				Higher is better		

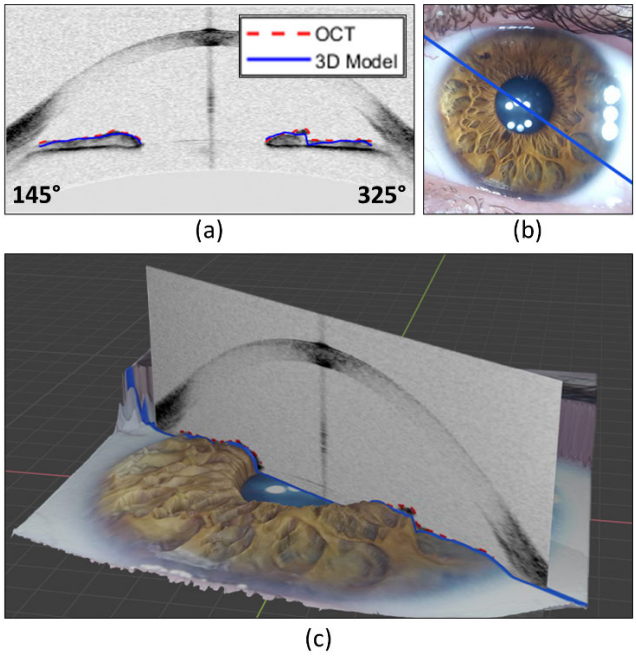


FIGURE 12. Example for the comparison between the OCT slice with the corresponding iris 3D model slice. (a) OCT slice with the ground-truth iris surface in red, and the 3D model slice in blue. (b) An iris image showing the slice angle in blue. (c) The 3D iris model with the OCT superimposed at the same angle.

depth data, and the powerful depth prediction architecture of DenseDepth.

C. DEPTH EVALUATION WITH OCT SCANS

We also assessed the performance of our method by comparing the generated 3D models against the depth ground truth provided by iris OCT slices. Figure 12 shows the comparison between an iris 3D model slice and the corresponding OCT. Figure 12a shows the OCT image with markings of the ground truth iris surface, and the slice of the iris 3D model. Figure 12b illustrates the angle of the slice and the iris features that are present along this line. Figure 12c shows a spatial comparison of the 3D model with the OCT. This visual comparison illustrates the changes in the 3D model across the profile, and shows how they closely match the OCT.

We then compared the difference quantitatively between the ground-truth iris surface in the OCT slices and the corresponding slices of the 3D models produced by

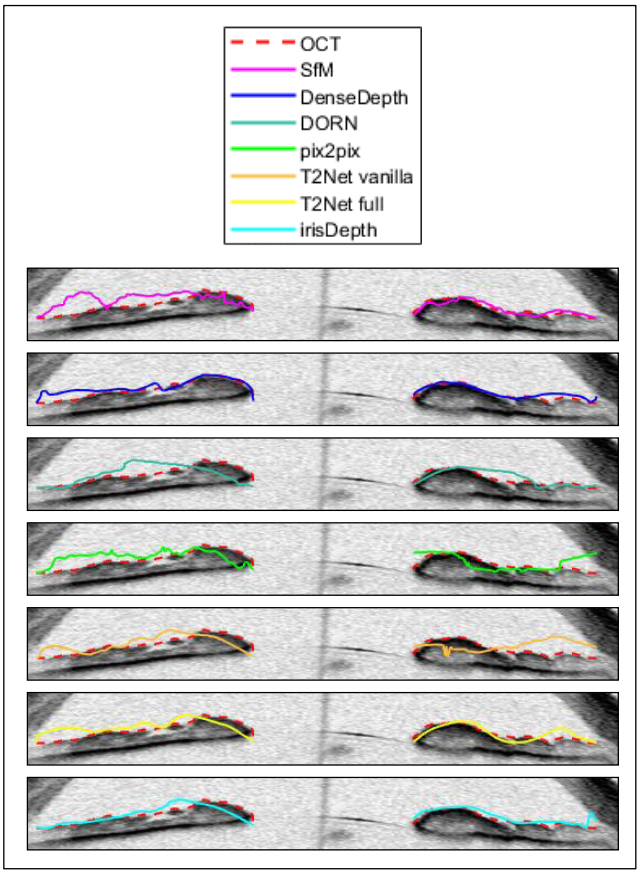


FIGURE 13. Comparison of one OCT slice of the iris with all the 3D models produced by SfM, and all the trained CNNs.

both SfM, and the different CNNs trained in this work. Figure 13 shows close-up comparisons between OCT slices and all the various 3D models produced by the different methods. Figure 13 shows that the models produced by DenseDepth, T^2Net_full, and irisDepth follow the depth ground truth of the OCT closely. The model produced by SfM has a great resemblance on the left side, but a significant difference on the right side of the iris. For each method, we have the curve of the OCT ground truth, and that of the 3D model slice. We computed the mean absolute error to quantify the error between both curves. We compared the 3D iris models that are produced by each method for the left and right eyes of the subject to the total of 8 available OCT slices for the right and left eyes (4 for each eye). Table 3 shows

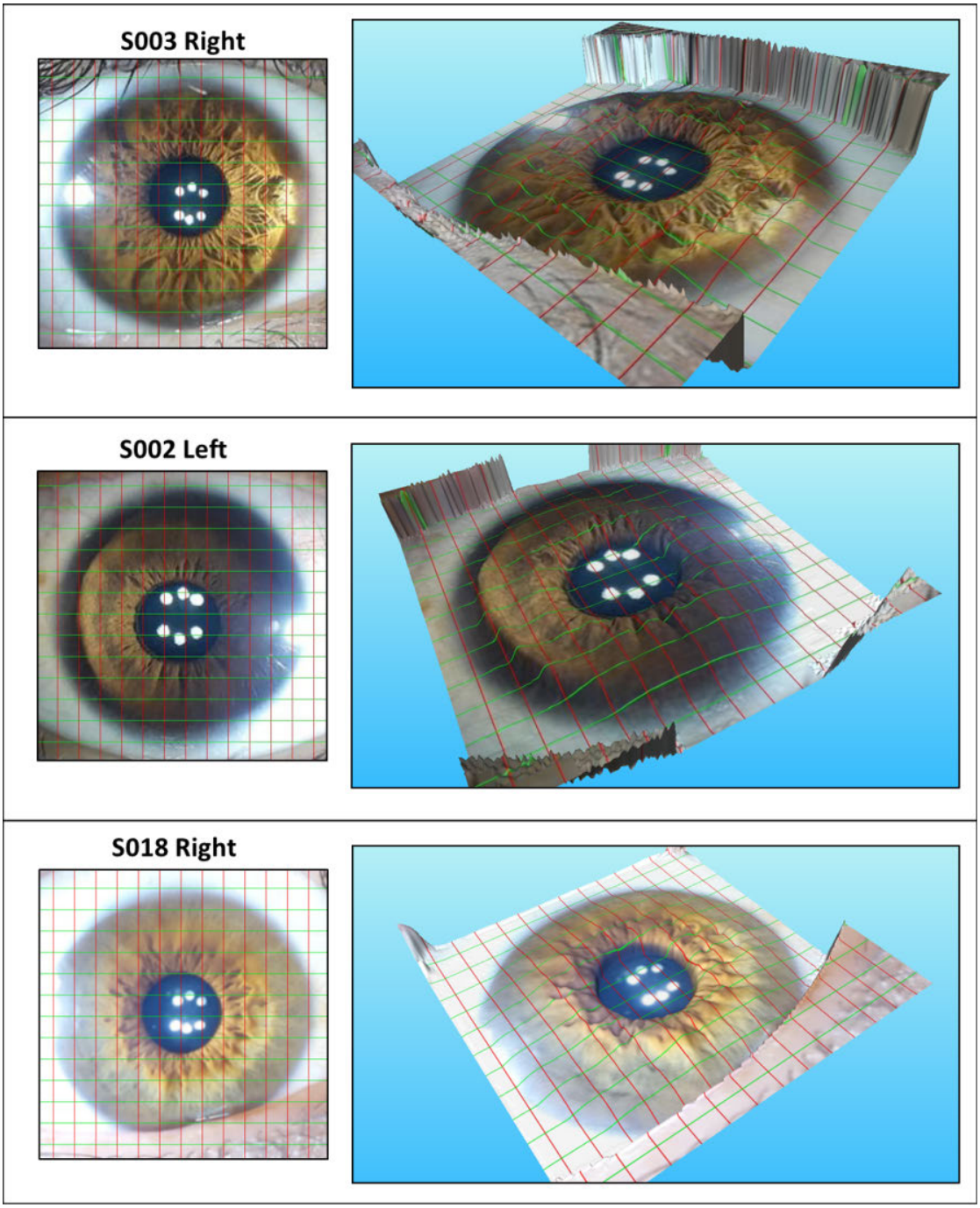


FIGURE 14. Example of human 3D iris models. A red-green grid was drawn on the surface of the models for a better visualization of 3D features. Each row shows the iris image of the subject on the left, and the iris 3D model on the right.

the results of the mean absolute error in micrometers when comparing each 3D iris model to the ground truth (OCT). The minimum average error of 77 μm was obtained with our model irisDepth. The typical thickness of the iris is around 500 μm [37], and therefore, the error achieved with the irisDepth method is within 15% of the thickness. Figure 13 also shows that irisDepth is the method that follows the ground truth the most closely. SfM produced the second to last good performance, and the error of SfM is 60% greater than the best CNN method (irisDepth). This indicates that the CNN irisDepth produces a more accurate 3D model from a single image than was achieved with SfM from multiple images.

D. 3D RECONSTRUCTION OF HUMAN IRISES

We produced pointcloud and mesh 3D models of the subjects in the test set using irisDepth. Figure 14 shows examples

TABLE 3. Depth estimation errors in μm between 3D model slices and OCT scans.

Method	Left Eye MAE [µm]	Right Eye MAE [µm]	Average [µm]
SfM [10]	111	134	123
DenseDepth [18]	74	98	86
DORN [19]	71	115	93
pix2pix [24]	92	129	111
T^2Net_vanilla [25]	119	160	140
T^2Net_full [25]	92	92	92
irisDepth (ours)	**69**	**86**	**77**

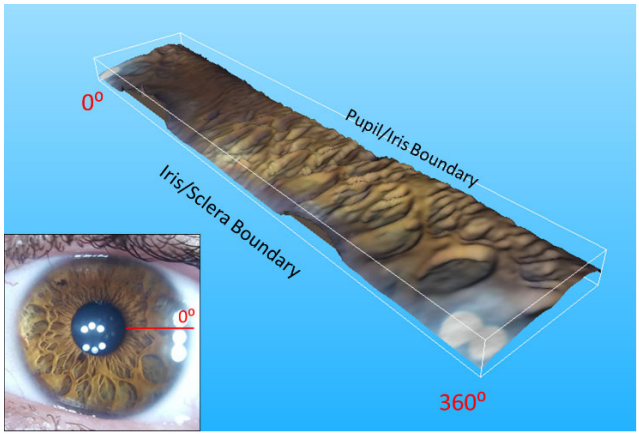


FIGURE 15. 3D Rubber Sheet obtained from 360 slices of the 3D model in Figure 12c. The iris image of the subject is shown on the bottom left corner along with the 0° line of the slicing process.

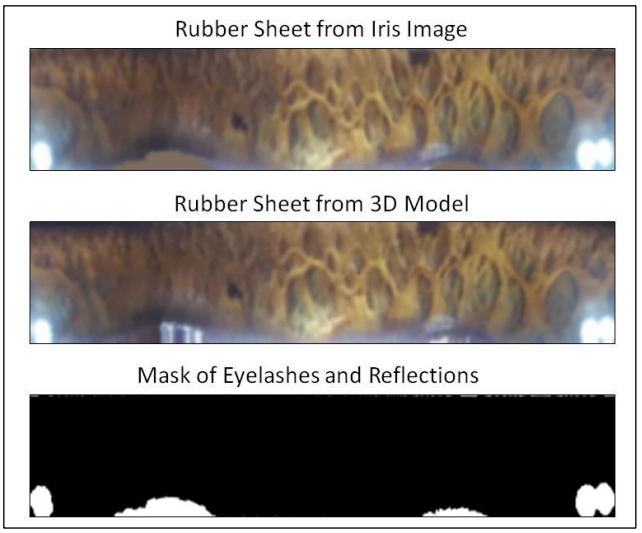


FIGURE 16. Comparison between the rubber sheet of an iris image in Figure 12b and the flattened version of the 3D rubber sheet in Figure 15. A mask was used in the comparison to avoid the effects of eyelids, eyelashes, and reflections [2].

of 3D mesh models for five different subjects. For the purpose of appreciating the 3D information in a 2D image, a red-green grid was drawn on the surface of the 3D model. In this way, deformations in the grid illustrate depth variations across the iris surface. This figure also shows the estimation of the 3D information performed by irisDepth from a single image of the human iris. The pointcloud models produce depth predictions from every pixel in the image. At a resolution of 192 × 192, the models have 36,864 3D points, and at 256 × 256 pixels, there are 65,536 3D points. In contrast, the SfM method reported an average production of 11,005 3D points [10]. Therefore, our CNN approach has more information available for producing the 3D model of the iris compared to that of the SfM approach.

Our results show that there are advantages to using CNNs over SfM for 3D iris model generation. Besides using multiple images at a greater resolution, SfM has problems producing 3D points in areas of the iris that have no significant texture. In contrast, the CNN models produce a uniform distribution of points regardless of iris texture. The number of 3D points obtained by CNNs is always constant, and it can be 6 times greater than those of SfM. Additionally, artifacts such as lateral reflections produced noisy points in the SfM model. One of the main advantages of our proposed method is that CNNs require only a single image for the 3D model estimation. This saves acquisition and processing time, as well as storage space. The acquisition time is relevant for subjects in the use of biometric applications. The SfM approach [10] requires capturing a burst of about 10 images per camera position for the 3D model construction. A set of one-hundred 16Mpx images, therefore, is typically used to reconstruct a single 3D model. Consequently, obtaining a 3D model from a single image is a significant improvement.

E. RUBBER SHEET MODEL AND 3D IRIS RECOGNITION PROOF OF CONCEPT

We reconstructed the 3D rubber sheet from the 3D model in Figure 12c by obtaining one 2D slice every 1°. The 3D rubber sheet is shown in Figure 15. The 3D rubber sheet captures the color information of the 2D image, as well as the depth of the iris. Just like a 2D rubber sheet, this is a representation of the human iris that normalizes dilation changes in a linear manner [2].

We then compared the rubber sheet from the iris image of Figure 12b with the projection of the 3D rubber sheet of Figure 15 onto the *xy* plane. Figure 16 shows the resulting rubber sheets, as well as the mask of eyelids, eyelashes, and reflections. This mask was used to ensure that those artifacts would not affect the comparison. The resulting MAE value for the comparison is 0.0313; ZNCC is 0.9385; and HD is 0.226. These values indicate a small error and a large correlation between the two images. This means that the reconstructions of the 3D model and the 3D rubber sheet preserve the information along the *xy* plane with a small error. Additionally, the low HD ensures a true positive in biometric tests. For context, in a previous work, we analyzed that the mean intraclass HD of LFVL images is 0.243, while

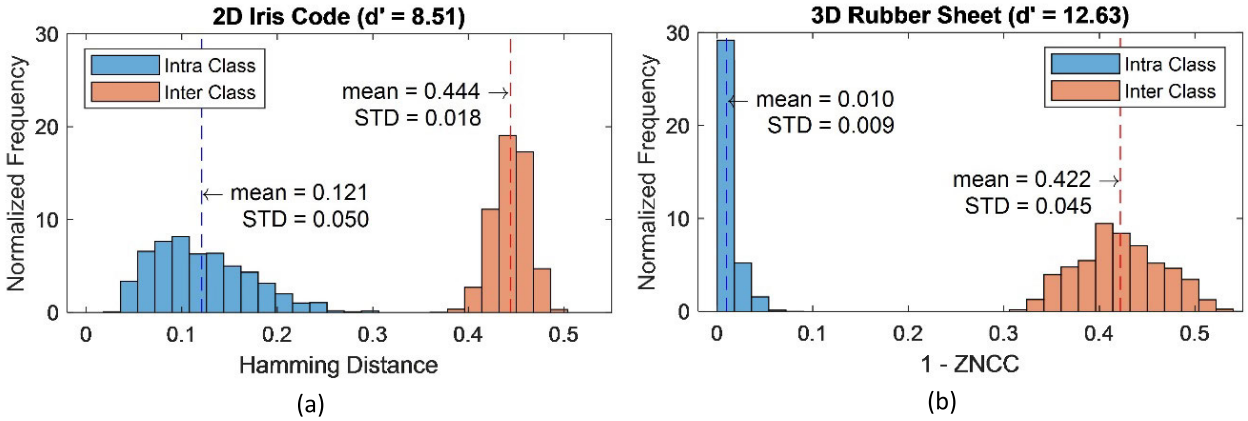


FIGURE 17. Iris Recognition performance in our test set of 12 subjects and 480 images. (a) Using the 2D Iris Code in Osiris V4.1 1 [2], [45]. (b) Using the proposed 3D rubber sheet model, with Spin Image Descriptors [43], [44] and ZNCC [41].

that of the interclass distribution is 0.48 [15]. Therefore, the HD value of 0.226 falls in the range of two different images from the same individual.

The results of the 3D iris recognition are presented in Figure 17 which shows the iris recognition performance of the 3D rubber sheet compared to that of the 2D iris code in our test set of 12 subjects and 480 images. The distributions in Figure 17 are normalized so that they have an area of 1. The results with the 2D iris code yielded a d' of 8.51, using Osiris V4.1. The 3D rubber sheet achieved a d' of 12.63, which is 48% higher. The mean value of the intra-class distribution is similar for both methods, with a value of approximately 0.45. However, the mean value of the intra-class distribution is 0.111 units less for the proposed 3D method. The results of this preliminary test show that the 3D characteristics extracted from the human iris are more discriminative than the 2D iris code.

The preliminary results of iris recognition in the test set of 12 subjects, along with the proof of concept of the Rubber Sheet model, and the depth evaluation tests with 8 ground-truth OCT slices of one subject illustrate the capabilities of the proposed method to reconstruct the surface of the human iris, and its applications in iris recognition. The tests with stepped pyramids of known dimensions demonstrate the smallest resolution our method can measure. All these evaluations show that our method can reconstruct a 3D model of the human iris with good performance.

V. CONCLUSIONS

Our proposed method for 3D iris model estimation from a single image produced complete 3D representations of the human iris using CNNs. Our method, irisDepth, uses the GAN part of a pre-trained T^2Net with the depth prediction of DenseDepth. Therefore, the GAN is not blind to depth information during training, and the depth prediction is more powerful than T^2Net alone. IrisDepth produced the best performance among the trained networks in both the synthetic and real iris tests. We used a dataset of 96 subjects randomly selected for training, 12 for validation and 12 for testing. There are 20,940 training images, 2,700 validation images and 2,880 testing images. We also used synthetic irises with 72,000 images. Both datasets used lateral illumination of the iris (LFVL) to enhance the shadows produced by iris features [15]. Thus, lateral illumination allowed the networks to relate shadows in RGB images to depth information.

We validated the results of our method for modeling the human iris by comparing slices of the 3D models with corresponding OCT slices of both eyes of one subject. The overall shape of the 3D models matches that of the OCT. Our method produced 65,536 3D points, with an absolute error of 77 μm on average. These numbers represent 6 times more 3D points and a 60% increase in accuracy with respect to previous 3D iris models based on SfM [10]. We proposed a 3D rubber sheet model proof of concept, which had a 0.9385 correlation with a 2D rubber sheet on the xy plane, and additional information on the z axis to be exploited. On a preliminary test with 480 images, the proposed 3D rubber sheet model increased iris recognition performance by 48% with respect to the standard 2D iris code [2]. Finally, the resolution of our method is 17.7 μm, as was measured by scanning 3D pyramids of known dimensions. This is roughly 1/30$^{\text{th}}$ of the iris thickness.

A 3D model of the iris may open research lines in iris recognition and ophthalmology. In addition to increasing accuracy in iris recognition [11], obtaining 3D information of the iris could help in extreme pose detection [46]–[50]. Additionally, a 3D model of the iris could produce information similar to that of an OCT, which could help ophthalmologists in the detection of closure angle glaucoma [10], [14].

Future improvements could increase the precision of our method. First, modifying the architecture to train with OCT slices or OCT based 3D models would produce 3D iris models that correlate more closely with actual OCT scans. Also, although CNN and SfM are traditionally used separately, a combination of them could yield a more robust method [28].

The CNN prediction could be the starting point for SfM, which could output more 3D points from several views at a higher resolution, thus improving the 3D model [12], [13].

ACKNOWLEDGMENT

The authors would like to thank Prof. Javier Ruiz-del-Solar and Dr. Patricio Loncomilla from AMTC for their insights on training CNNs. The acquisition of the OCTs for this research was possible thanks to the help of Dr. Claudio I. Perez from Fundación Oftalmológica Los Andes. They would also like to thank the students at the School of Engineering, Universidad de Chile, who participated enthusiastically as volunteers for iris image acquisition.

REFERENCES

[1] K. Hollingsworth, K. W. Bowyer, and P. J. Flynn, "Pupil dilation degrades iris biometric performance," *Comput. Vis. Image Understand.*, vol. 113, no. 1, pp. 150–157, Jan. 2009.
[2] J. Daugman, "How iris recognition works," *Essent. Guid. Image Process.*, vol. 14, no. 1, pp. 715–739, 2004.
[3] L. Masek, "Recognition of human iris patterns for biometric identification," M.S. thesis, School Comput. Sci. Softw. Eng., Univ. Western Australia, Perth, WA, Australia, 2003.
[4] K. W. Bowyer, K. Hollingsworth, and P. J. Flynn, "Image understanding for iris biometrics: A survey," *Comput. Vis. Image Understand.*, vol. 110, no. 2, pp. 281–307, May 2008.
[5] A. Gangwar and A. Joshi, "DeepIrisNet: Deep iris representation with applications in iris recognition and cross-sensor iris recognition," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 2301–2305.
[6] H. Proenca, "Iris recognition in the visible wavelength," in *Handbook of Iris Recognition*. London, U.K.: Springer, 2013, pp. 151–169.
[7] A. Clark, S. Kulp, I. Herron, and A. A. Ross, "A theoretical model for describing iris dynamics," in *Handbook of Iris Recognition*. London, U.K.: Springer, 2016, pp. 417–438.
[8] J. E. Tapia, C. A. Perez, and K. W. Bowyer, "Gender classification from the same iris code used for recognition," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 8, pp. 1760–1770, Aug. 2016.
[9] D. Bastias, C. A. Perez, D. P. Benalcazar, and K. W. Bowyer, "A method for 3D iris reconstruction from multiple 2D near-infrared images," in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, Oct. 2017, pp. 503–509.
[10] D. P. Benalcazar, D. Bastias, C. A. Perez, and K. W. Bowyer, "A 3D iris scanner from multiple 2D visible light images," *IEEE Access*, vol. 7, pp. 61461–61472, 2019.
[11] F. Cohen, S. Sowmithran, and C. Li, "Iris Identification in 3D," in *Proc. Scand. Conf. Image Anal.*, Jun. 2019, pp. 324–335.
[12] C. Wu, "Towards linear-time incremental structure from motion," in *Proc. Int. Conf. 3D Vis.*, Jun. 2013, pp. 127–134.
[13] Y. Furukawa and C. Hernández, "Multi-view stereo: A tutorial," *Found. Trends Comput. Graph. Vis.*, vol. 9, nos. 1–2, pp. 1–148, 2013.
[14] I. I. Bussel, G. Wollstein, and J. S. Schuman, "OCT for glaucoma diagnosis, screening and detection of glaucoma progression," *Brit. J. Ophthalmology*, vol. 98, no. 2, Jul. 2014, pp. ii15–ii19.
[15] D. Benalcazar, C. Perez, D. Bastias, and K. Bowyer, "Iris recognition: Comparing visible-light lateral and frontal illumination to NIR frontal illumination," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 867–876.
[16] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 1994, pp. 593–600.
[17] M. Kazhdan and H. Hoppe, "Screened Poisson surface reconstruction," *ACM Trans. Graph.*, vol. 32, no. 3, pp. 1–13, Jun. 2013.
[18] I. Alhashim and P. Wonka, "High quality monocular depth estimation via transfer learning," 2018, *arXiv:1812.11941*. [Online]. Available: http://arxiv.org/abs/1812.11941
[19] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2002–2011.
[20] D. Wofk, F. Ma, T.-J. Yang, S. Karaman, and V. Sze, "FastDepth: Fast monocular depth estimation on embedded systems," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 6101–6108.
[21] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 3, Jan. 2014, pp. 2366–2374.
[22] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2650–2658.
[23] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 239–248.
[24] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1125–1134.
[25] C. Zheng, T. J. Cham, and J. Cai, "T2Net: Synthetic-to-realistic translation for solving single-image depth estimation tasks," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 767–783.
[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[27] D. Xu, W. Wang, H. Tang, H. Liu, N. Sebe, and E. Ricci, "Structured attention guided convolutional neural fields for monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3917–3925.
[28] K. Tateno, F. Tombari, I. Laina, and N. Navab, "CNN-SLAM: Real-time dense monocular SLAM with learned depth prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6565–6574.
[29] C. Godard, O. M. Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6602–6611.
[30] Y. Kuznietsov, J. Stuckler, and B. Leibe, "Semi-supervised deep learning for monocular depth map prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6647–6655.
[31] Y. Tian, X. Li, K. Wang, and F.-Y. Wang, "Training and testing object detectors with virtual images," *IEEE/CAA J. Automatica Sinica*, vol. 5, no. 2, pp. 539–546, Mar. 2018.
[32] J. E. Cutting and P. M. Vishton, "Perceiving layout and knowing distances: The integration, relative potency, and contextual use of different information about depth," in *Perception of Space and Motion*. New York, NY, USA: Academic, 1995, pp. 69–117.
[33] P. Corke, "Vision," in *Robotics, Vision and Control*, 2nd ed. Brisbane, Australia: Springer, 2013, pp. 251–283.
[34] X. Yuan and P. Shi, "A non-linear normalization model for iris recognition," in *Proc. Int. Work. Biometric Pers. Authentication*, 2005, pp. 135–141.
[35] F. Behar-Cohen, C. Martinsons, F. Viénot, G. Zissis, A. Barlier-Salsi, J. P. Cesarini, O. Enouf, M. Garcia, S. Picaud, and D. Attia, "Light-emitting diodes (LED) for domestic lighting: Any risks for the eye?" *Prog. Retinal Eye Res.*, vol. 30, no. 4, pp. 239–257, Jul. 2011.
[36] Blender 2.8. (2019). *Stichting Blender Foundation*. [Online]. Available: https://www.blender.org/
[37] M. He, D. Wang, J. W. Console, J. Zhang, Y. Zheng, and W. Huang, "Distribution and heritability of iris thickness and pupil size in chinese: The guangzhou twin eye study," *Investigative Opthalmology Vis. Sci.*, vol. 50, no. 4, p. 1593, Apr. 2009.
[38] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2242–2251.
[39] C. Chu, A. Zhmoginov, and M. Sandler, "CycleGAN, a master of steganography," 2017, *arXiv:1712.02950*. [Online]. Available: http://arxiv.org/abs/1712.02950
[40] L. Sorbara, J. Maram, D. Fonn, C. Woods, and T. Simpson, "Metrics of the normal cornea: Anterior segment imaging with the visante OCT," *Clin. Experim. Optometry*, vol. 93, no. 3, pp. 150–156, Apr. 2010.
[41] S. Sadek, M. G. Iskander, and J. Liu, "Accuracy of digital image correlation for measuring deformations in transparent media," *J. Comput. Civil Eng.*, vol. 17, no. 2, pp. 88–96, Apr. 2003.
[42] E. Ortiz, K. W. Bowyer, and P. J. Flynn, "Dilation-aware enrolment for iris recognition," *IET Biometrics*, vol. 5, no. 2, pp. 92–99, Jun. 2016, doi: 10.1049/iet-bmt.2015.0005.
[43] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999, doi: 10.1109/34.765655.

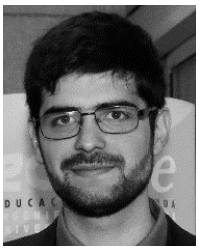
[44] A. Eleliemy, A. Mohammed, and F. M. Ciorba, "Efficient generation of parallel spin-images using dynamic loop scheduling," in *Proc. IEEE 19th Int. Conf. High Perform. Comput. Commun. Workshops (HPCCWS)*, Dec. 2017, pp. 34–41, doi: 10.1109/HPCCWS.2017.00012.
[45] N. Othman, B. Dorizzi, and S. Garcia-Salicetti, "OSIRIS: An open source iris recognition software," *Pattern Recognit. Lett.*, vol. 82, pp. 124–131, Oct. 2016, doi: 10.1016/J.PATREC.2015.09.002.
[46] E. T. Celik and M. Karakaya, "Pupil dilation at synthetic off-angle iris images," in *Geselischaft Fur Inform. e.V.* Darmstadt, Germany, 2016, pp. 6–10, doi: 10.1109/BIOSIG.2016.7736934.
[47] A. Jourabloo and X. Liu, "Large-pose face alignment via CNN-based dense 3D model fitting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4188–4196.
[48] P. Dou, S. K. Shah, and I. A. Kakadiaris, "End-to-end 3D face reconstruction with deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1503–1512.
[49] A. S. Jackson, A. Bulat, V. Argyriou, and G. Tzimiropoulos, "Large pose 3D face reconstruction from a single image via direct volumetric CNN regression," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1031–1039.
[50] A. Tewari, M. Zollhofer, H. Kim, P. Garrido, F. Bernard, P. Perez, and C. Theobalt, "MoFA: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 3735–3744.

DANIEL P. BENALCAZAR (Graduate Student Member, IEEE) was born in Quito, Ecuador, in 1987. He received the B.S. degree in electronics and control engineer from Escuela Politecnica Nacional, Quito, in 2012, and the M.S. degree in electrical engineering from The University of Queensland, Brisbane, QLD, Australia, in 2014, with a minor in biomedical engineering. He is currently pursuing the Ph.D. degree with the Universidad de Chile, Santiago, Chile.

From 2015 to 2016, he worked as a Professor at the Central University of Ecuador. Ever since, he has participated in various research projects in biomedical engineering and biometrics.

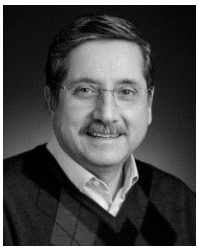
JORGE E. ZAMBRANO was born in Latacunga, Ecuador, in 1991. He received the B.S. degree in electronics and instrumentation engineer from Escuela Politecnica del Ejercito-ESPE, in 2015. He is currently pursuing the Ph.D. degree with the Universidad de Chile, Santiago.

His research interests are in biometrics with image processing and machine learning.



DIEGO BASTIAS received the B.S. degree in electrical engineering and the P.E. degree (Hons.) in electrical engineering from the Universidad de Chile, in 2014 and 2016, respectively.

He is currently working as a Research Assistant at the Department of Electrical Engineering, Universidad de Chile. His current research interests include biometrics and computer vision.



CLAUDIO A. PEREZ (Senior Member, IEEE) received the B.S. and P.E. degrees in electrical engineering and the M.S. degree in biomedical engineering from the Universidad de Chile, in 1980 and 1985, respectively, and the Ph.D. degree from The Ohio State University, in 1991.

He was a Fulbright Student at The Ohio State University, where he received a Presidential Fellow, in 1990. He was a Visiting Scholar with UC Berkeley, Berkeley, in 2002, through the Alumni Initiatives Award Program from the Fulbright Foundation. He was the Department Chairman, from 2003 to 2006, and the Director of the Office of Academic and Research Affairs with the School of Engineering, Universidad de Chile, from 2014 to 2018. He is currently a Professor with the Department of Electrical Engineering, Universidad de Chile. His research interests include biometrics, image processing applications, and pattern recognition. He is a Senior Member of the IEEE Systems, Man and Cybernetics and IEEE-CIS societies.

KEVIN W. BOWYER (Fellow, IEEE) is currently the Schubmehl-Prein Family Professor with the Department of Computer Science and Engineering, University of Notre Dame, and the Director of the College of Engineering Summer International Programs. His main research interests are in computer vision and pattern recognition, including biometrics, data mining, object recognition, and medical image analysis. He is a Fellow of the IEEE for contributions to algorithms for recognizing objects in images and the IAPR for contributions to computer vision, pattern recognition, and biometrics. He received the IEEE Computer Society Technical Achievement Award for pioneering contributions to the science and engineering of biometrics and the inaugural IEEE Biometrics Council Meritorious Service Award. He is serving as the inaugural Editor-in-Chief of the new IEEE TRANSACTIONS ON BIOMETRICS, BEHAVIOR, AND IDENTITY SCIENCE (T-BIOM).