3D Face Recognition using Inception Networks for Service Robots

Sérgio Baixo
Industrial Electronics Dept.
University of Minho
Guimarães, Portugal
a76513@alunos.uminho.pt

Tiago Ribeiro
Industrial Electronics Dept.
ALGORITMI Center
University of Minho
Guimarães, Portugal
id9402@alunos.uminho.pt

Gil Lopes
Communication Sciences and
Information Technologies Dept.
University Institute of Maia
Maia, Portugal
alopes@ismai.pt

A. Fernando Ribeiro
Industrial Electronics Dept.
ALGORITMI Center
University of Minho
Guimarães, Portugal
fernando@dei.uminho.pt

***Abstract* — The field of face recognition has significantly advanced as deep learning methods, such as those using CNNs, continuously show improvements. However, despite face recognition's promising potential, there are still many concerns regarding privacy and safety. Moreover, the first 2D algorithms, besides having good performance, turned out to be influenced by several factors like the environment's lighting conditions, pose, and facial expression of the subjects, compromising the model's accuracy. This work describes the development of a computer vision system using Deep Learning methods to detect and recognise human faces in 3D in real-time. The RGB images and depth maps from several subjects were captured using an Intel RealSense D455, processed, and consequently provided into two independent CNNs, an Inception-Resnet V1 to deal with the RGB images and an Inception V3 to deal with depth maps. The final algorithm was implemented on the anthropomorphic domestic and healthcare service robot CHARMIE (Collaborative Home Assistant Robot by Minho Industrial Electronics) to perform its tasks according to the recognised user.**

Keywords – Face Recognition, 3D Image, Deep Learning, Convolutional Neural Network, Inception Resnet

I. INTRODUCTION

According to [1], computer vision and robot vision are distinct concepts. The first targets the understanding of a scene mostly from single images or from a fixed camera position. Its methods are designed for specific applications, and the research is focused on individual problems and algorithms. On the other hand, the second requires looking at the system level perspective, where computer vision is one of several sensory components that work together to fulfil specific tasks, such as in anthropomorphic robots. Regarding biometric analysis, face recognition is a natural biometric technique embedded in the everyday lives of human beings. Amongst all biometric technologies used so far, face recognition is one of the most widely outspread biometric technologies [2]. Concerning 3D face recognition, this is a very well-known technology that is broadly used by a continuously increasing large number of people since a significant part of the world's population uses a smartphone equipped with some biometric recognition capability. Although the accuracy of previous 2D face recognition algorithms was considerably high, in [3], some factors such as the influence of lighting around the subject, orientation, and the subject's expression compromised the model´s overall effectiveness. Another great disadvantage of 2D methods concerns safety, where a photograph of a person yields the same outcome as the person in real life. To fill in these gaps, 3D face recognition introduces depth maps, a one-channel image that contains information about the distance between the surfaces of objects from a given standpoint. The purpose of depth information and 3D recognition overall is to effectively minimize the influence of illumination, facial posture, and expression commonly associated with 2D face recognition. Besides depth maps, depth information can also be represented in the form of point clouds and face meshes. The main goal of this work is to apply the resulting trained models to the service robot CHARMIE [4]. Since this is a collaborative anthropomorphic robot, its purpose is to aid and collaborate with humans by helping them to perform tasks. By developing face recognition, it is possible to make the robot perform such tasks adjusted to the person it interacts with.

II. RELATED WORK

The performance of 2D face recognition systems has been substantially improved with the adoption of Convolutional Neural Networks (CNN). Using CNN feature extraction trained on massive datasets outperforms traditional methods, which use hand-crafted feature extraction methods. Deep learning methods demand large datasets to learn face representations, depending on multiple factors like postures and expressions. Large scale datasets of 2D face images are available throughout the internet. Facenet[5] uses roughly 200 million face images from 8 million people, and the network is based on Inception modules. VGG Face[6] proposes a procedure to assemble a large dataset comprising 2.6 million face images from over 2700 individuals trained on VGG-16 networks[7]. Szegedy[8] analyses CNN facial recognition performance of various Inception-Resnet and Inception network versions, and the results look promising. Inception Resnet combined with a Multi-task Cascaded Convolutional Networks (MTCNN) and an SVM classifier were used in the previous version implemented on CHARMIE [9], only detecting in 2D. Alongside the development of active 3D sensing techniques, it is now possible to acquire 3D face models without concerns regarding pose, expression, and others. There are two different types of 3D sensors. 3D

978-1-6654-8217-2/22/$31.00 ©2022 IEEE

scanners such as Minolta produce high-quality results but lacks performance and is quite expensive. The second kind consists of depth cameras like the Microsoft Kinect [10], which, even though it is compact and affordable, has low resolution, and weak reliability. In contrast, the Intel RealSense depth cameras have a higher resolution and are more reliable and affordable. In [11], low-quality RGB-D data from an affordable 3D sensor is used to deal with face recognition problems under several poses. For similarity calculation, both texture images and face depth maps were transformed into frontal views. Besides this, symmetric filling on texture images may degrade the matching performance. As opposed to frontalization on the 2D face image, in [12], multiple face images were generated under some predefined poses from 3D face models in the gallery. Tests showed that the recognition rate was higher than the frontal method. The texture images were deformed according to rotation from depth maps to deal with pose variation achieved a 69.1% rank-1 recognition rate on the Bosphorus database[10].

III. Methodologies

The system's prototype development was segmented into five sequential stages: image acquisition, image processing, dataset preparation, dual model training, and testing. These sequential phases are illustrated in Figure 1. Inside the execution cycle, two Haar Cascade classifiers are operated (despite being developed outside the five stages), one of them specifically designed to detect faces in RGB images, while the other was developed to detect faces in depth maps.

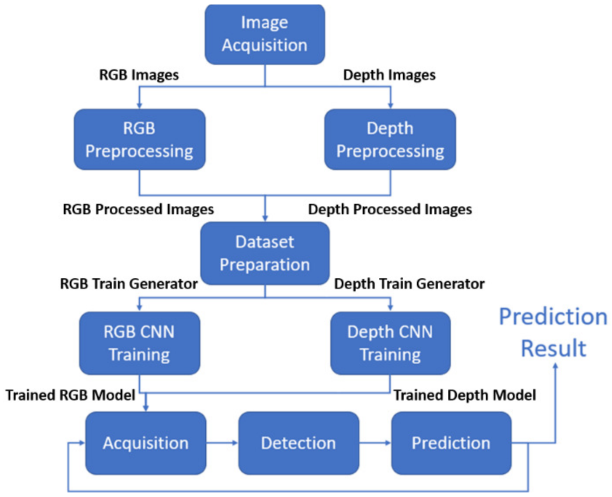

Figure 1- Layout of the system's stages

A. Hardware and Software

The camera used was an Intel RealSense D455, which simultaneously provides regular RGB images and depth map information. This 3D sensor also enables the distance extraction to a given target. The image acquisition and processing took place in Pycharm IDE using Python language. The training of both CNNs took place using Google Colab Pro. The networks were also implemented in Google Colab Pro using Tensorflow and Keras frameworks.

B. Image Acquisition

Several sets of images of various subjects were acquired to build the dataset. Each set of images was captured with the subjects facing different directions: forwards, left, right, up, and down. This image capture process resulted in 551 RGB images (Figure 2) and 551 depth images (Figure 3) per subject from multiple poses. Both figures 2 and 3 contain the raw captured images. It is necessary to apply some image processing to remove some noise.

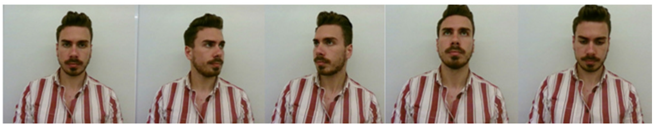

Figure 2- Samples of RGB images from RGB dataset with the subject facing different directions

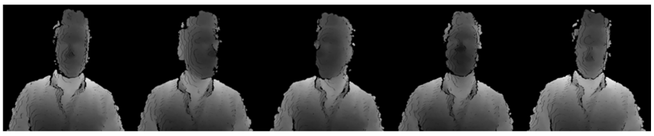

Figure 3- Samples of depth images from depth dataset with the subject facing different directions

C. Image Processing

Initially, a Haar Cascade classifier extracted the faces from the captured images. The implemented algorithm detects a face on a given frame, draws a square surrounding it based on the coordinates of the face detected, and crops it, leaving behind the outside of the square area. Figure 4 illustrates this process. This is carried out for all RGB images per subject, overriding the raw images. Regarding depth images, these experienced more processing than RGB ones. To reduce noise in the images, a dilation operation, having an ellipse shape as a structuring element, was implemented using OpenCV. Also, the previously extracted coordinates in the respective RGB image were used to crop the depth images. The final result is portrayed in Figure 5.

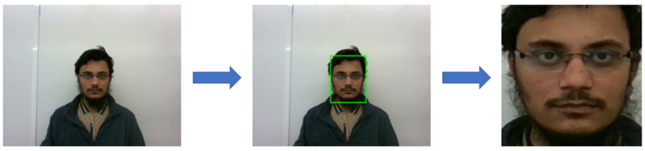

Figure 4- Processing of an RGB image

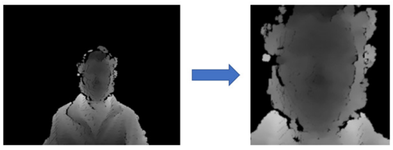

Figure 5- Processing of a depth image

D. Dataset Preparation

After the image processing, it is essential to prepare the dataset before feeding the images to the CNNs. This type of data preparation is critical to ensure the data training process is carried out effectively, smoothly, and without impacting memory resources. The preparation is carried out by creating proper image generators for image directories. An image data generator is a class available in the Keras API that allows the loading of images according to stipulated batch size. Image augmentation is applied and fed directly to the neural networks during the withdrawal process to a specified directory. This technique saves a considerable amount of system memory since the images are loaded one by one into the networks rather than all at once.

E. Training

Both networks were implemented using transfer learning, ensuring both of them produce the best possible outcome in fewer epochs. This method allows the reuse of a trained network, including its respective weights and features, and apply those to the desired problem. VGG-16, Inception-Resnet V1 and Inception V3 were assessed to conclude which one had the best performance both on the RGB dataset and the depth dataset. Deep Learning models need lots of data to accurately learn and make predictions, and 551 images per model, all with similar poses, might likely be a reduced number to build a reliable and robust model. Image augmentation virtually creates new training images from existing training images by applying transformations to copies of some of the samples from the training data, creating new and different training examples, increasing the size of the datasets and their data diversity. Besides, this also allows the models to generalise better on new or unseen data, thus making them more robust. These transformations include a wide range of image manipulations, like shifts, horizontal and vertical flips, zooms, rotations, changes in brightness, and so on. Figure 6 portrays some of the transformations applied to the training set images using image augmentation.

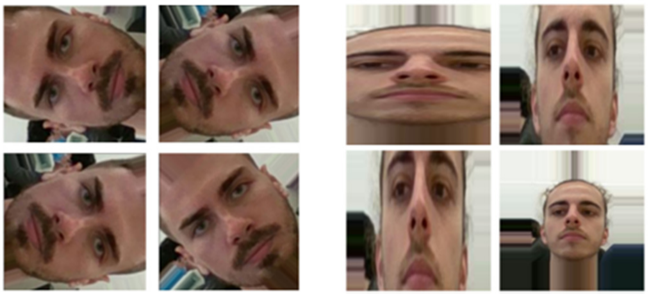

Figure 6- Random Rotation (left) and Zoom (right) Transformations

F. *Haar Cascade Classifiers*

Two Haar Cascade classifiers were integrated into this work. The RGB classifier was imported from OpenCV since it is integrated with the library. This example can be seen in Figure 4. The other classifier was entirely developed to recognise only depth faces. To do this, the classifier learns what to recognise with positive images while learning what not to recognise using negative images. Positive images correspond to depth faces, while negative images correspond to everything that is not a depth face but can appear in the same frame as a positive image. For example, the camera can capture the depth of a shoulder in the same frame as the depth face. Since the shoulder should not be detected as a face, it is considered a negative image. Other important negative images are noisy images since the depth sensor has some noise associated with the depth image. Figure 7 shows examples of positive and negative images.

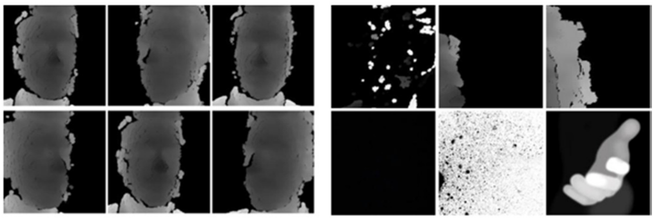

Figure 7- Positive (left) and Negative (right) Images

After training the classifier, a .h5 file is generated and imported to the project's directory. The result obtained from the created file is similar to Figure 8.

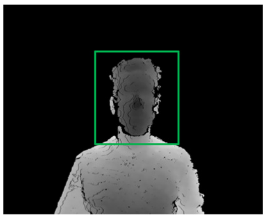

Figure 8- Face depth map detection

IV. TESTS

In this section, multiple CNN architectures and different learning rates are assessed to compare how different CNNs perform when dealing with the same data. The CNNs evaluated are VGG-16, Inception V3, and Inception-Resnet V1, which are amongst the most used CNN architectures[13]. The optimiser selected is the Adam optimiser. Also, for all these tests, an Early Stopping function was used to halt the training once the model's performance stopped evolving on the validation set, in this case, in a span of 20 epochs. If there was no improvement whatsoever on the validation set, the training stops. This technique is helpful to prevent overfitting. A static batch size of 32 has been set for all tests performed.

A. *RGB Dataset*

In Figure 9, it is possible to see the comparison between the results (training and validation) across different learning rates using the VGG-16 architecture trained using RGB images. It is noticeable that despite all the training having converged, the higher learning rate ones converged quicker. The lowest learning rate took 44 minutes and 49 seconds. The learning rate of 0.0004 took 19 minutes and 20 seconds, while the largest learning rate took 11 minutes and 35 seconds to train. On the other hand, only the lowest learning rate has a relatively smoother learning curve in the validation set.

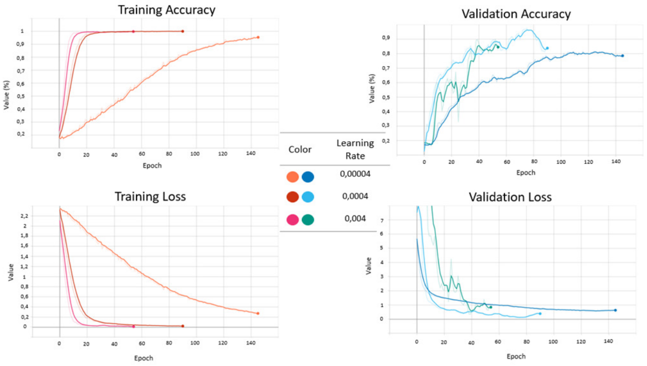

Figure 9- Training (left) and validation (right) results of VGG-16 on RGB dataset

The following architecture used is the Inception V3. Similar to the previous test, this one generated very similar results, at least when comparing the training record as shown in Figure 10. Again, all the models (with different learning rates) have converged, indicating that the models have learned from the dataset.

Figure 11 shows the theoretical results of the test made with the Inception-Resnet V1 across the three different training, each with three different learning rates, the ones used previously. The results show great performance across all training, with all results converging rather quickly.

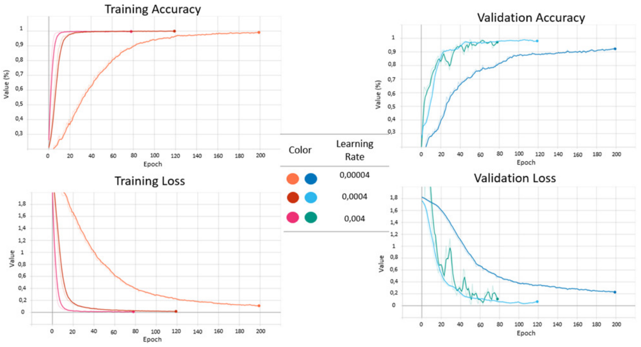

Figure 10- Training (left) and validation (right) results of Inception V3 on RGB dataset

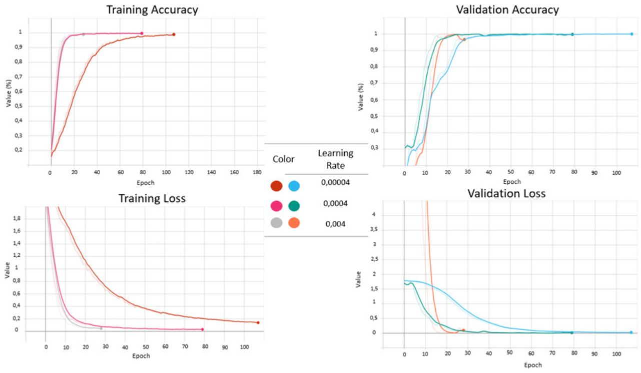

Figure 11- Training (left) and validation (right) results of Inception-Resnet V1 on RGB dataset

B. *Depth Dataset*

The test methods are similar to the previous ones using the depth dataset. The architecture used here is the Inception V3. As said previously, the pre-loaded weights were extracted from ImageNet, since there was no weights file resulting from the training of depth faces. Judging by the results in Figure 12, it can be assumed that the model with a learning rate of 0.00004 has a more stable learning process. Overall, the lowest learning rate has always the smoothest learning curve with all the tests performed. If the learning rate is lowered even more, it will end up with a longer training process since it takes more time to converge, and there could be a risk of the model getting "stuck" in the training process.

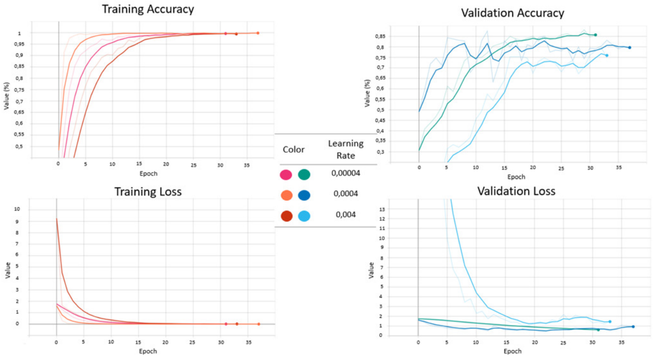

Figure 12- Training (left) and validation (right) results of Inception V3 on depth dataset

V. Results

A. *RGB Model*

To test the models with novel and unseen data outside the training process, an additional function was developed to assess the results of the trained models. However, the following first tests were not performed in real-time. Instead, these were performed by fetching each new image from a given directory, making the required processing and normalisation for each image and then making a prediction and returning its result. Only the final tests further described portray a real-time model prediction. For reference, Figure 13 (Top) shows the correct name (class) for each of the subjects present in the dataset. Figure 13 (Bottom) has three unknown subjects to the trained models. The unknown subjects were only used to analyse how the model reacted when predicting unknown people.

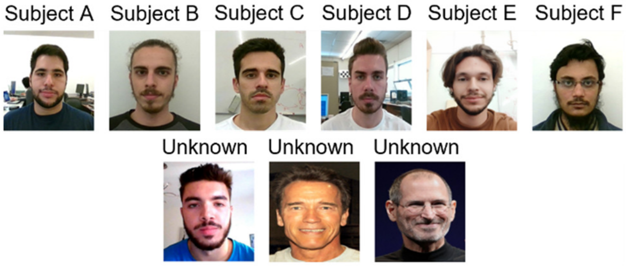

Figure 13- Illustration of the six classes of the RGB dataset (Top). Three unknown subjects to the model (Bottom)

By performing predictions with the VGG-16 model trained previously, Figure 14 shows that the model made the correct prediction in some images, while it got them wrong in others. This set of predictions used a learning rate of 0.00004. The worst-case scenario is observable here when the model "thinks" that an unknown person is a subject of the dataset. This occurred in three predictions, all of which had very high confidence scores. Besides that, the model also failed to make the correct prediction in several other images, which makes this model with a learning rate of 0.00004 unreliable and unusable in real-time image prediction. The results were very similar by changing the learning rate to either 0.00004 or 0.004, the two other learning rates tested in the previous test set. It only some changed results in one image or another but, the model still made several mistakes. This led to the conclusion that this model is unsuitable for dealing with RealSense's dataset. Therefore, another model architecture was tested to see if it uncovers more acceptable results.

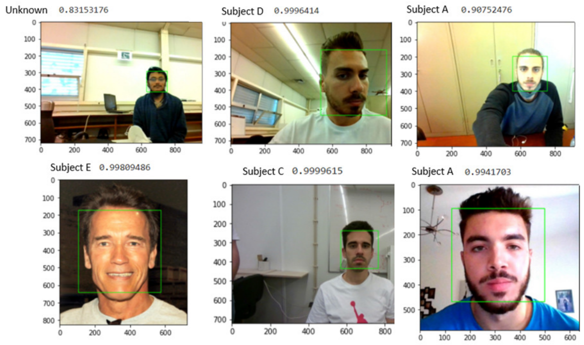

Figure 14- Prediction results of VGG-16 model with a learning rate of 0.00004

The following architecture used was the Inception V3 using 0.004, 0.0004 and 0.00004 learning rates. By looking at some of the predictions outputted by this architecture (Figure 15), it is perceived that neither prediction had a perfect score on all images while using an Inception V3 architecture, regardless of the learning rate used. Likewise, the VGG-16 can indicate that the dataset is unsuitable for this architecture or needs more time to learn.

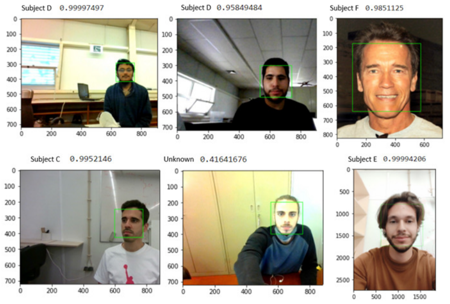

Figure 15- Prediction results of the Inception V3 model using several learning rates

Figure 16 shows the predictions made by the Inception-Resnet V1 model, including the results on extreme conditions, like having the subjects far away from the camera and or wearing a mask. The model could still recognise Subject A and Subject F even though they were farther away and despite Subject F looking slightly to the side. The model also provides low confidence predictions for unknown subjects (around 50%). Besides all of these, the model's most remarkable feat is correctly predicting when the subject is wearing a mask, and with relatively high accuracy. Subject B's confidence percentages wearing a mask are similar to their corresponding image without a mask. Considering that the model trained on an Inception-Resnet V1 having a learning rate of 0.00004 correctly predicted all the tests performed, this is the model selected for the final real-time implementation of the system.

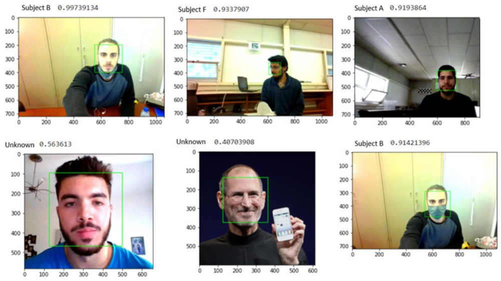

Figure 16- Correctly predicted images from the Inception-Resnet V1 model

B. Depth Model

The process used in previous models was repeated to test the depth model capability. Figure 17 shows the results of the predictions carried out with this model. The model correctly predicts most of the faces, while others mistake them either for another face or none at all. Inception V3 was the choice to be used in the final real-time implementation of the system since it was the architecture that obtained the best overall results.

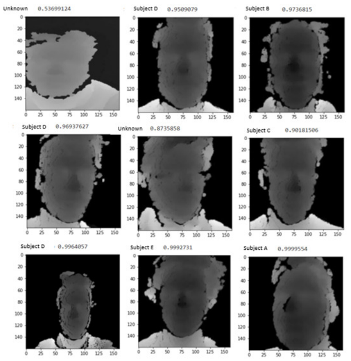

Figure 17- Prediction results of Inception V3 with depth dataset model with a learning rate of 0.00004

C. Real-Time Results

In this section, the final prototype is tested. It is the collection of all the previous results, from the image capturing and processing to the CNNs, that performed better. Both Haar Cascade classifiers (one for RGB images and the other for depth images) are also functioning to validate a person's presence in front of the camera. The program starts by loading both CNN models previously generated into the source code. The Haar Cascade classifier file created is also loaded into the project. Also, the Haar Cascade classifier for detecting regular non-depth faces is loaded, with the intent of detecting a face and highlighting it. The prediction result is made of the subject's predicted name and the respective percentage of confidence. The values next to the subject's name correspond to the confidence percentage of each model. The top prediction regards the one made by the RGB model, while the bottom one corresponds to the one made by the depth model. Figure 18 has the detection of a subject and its respective prediction. A decrease in confidence percentage in both models when the subject is wearing a mask can be noticed. This behaviour is expected since, with a mask on, there is a lot less information that the model can use to make predictions. It is important to note that, during training, all images were taken without a mask, meaning the model had never seen a subject with a mask on until these predictions.

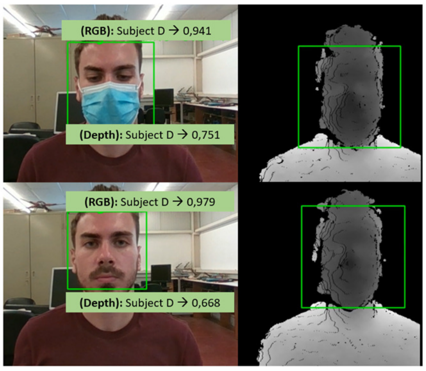

Figure 18- Detection and correct prediction with and without a mask

Figure 19 shows Subject C in two irregular poses, looking slightly downwards wearing a mask and another one with the eyes almost closed in a dark environment. Even under challenging conditions, the RGB model does not struggle when predicting. In contrast, the depth model struggles a little more, resulting in a lower percentage of confidence compared to the RGB model.

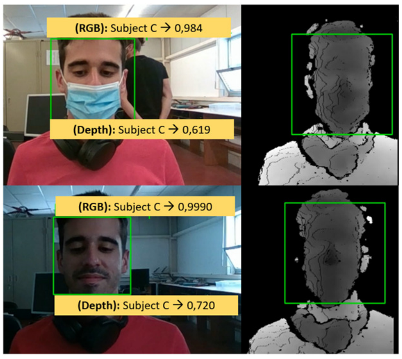
(RGB): Subject C → 0,984
(Depth): Subject C → 0,619
(RGB): Subject C → 0,9990
(Depth): Subject C → 0,720

Figure 19- Detection and correct prediction of Subject C

In an attempt to mislead the system, a subject from the dataset was shown, but in a mobile phone. As stated before, one of the primary purposes of 3D face recognition was to implement additional security measures to prevent the system from being fooled, making it more reliable and robust. These security measures were implemented in this environment through the depth image Haar Cascade classifier. The classifier was trained to detect depth faces and discard noisy images or any other irrelevant object or surface present in the camera's line of sight at a short distance. If a depth face is detected, that means for the system that a "real" person is standing in front of the camera, allowing it to be possible to validate a legitimate person and not a printed image of a face.

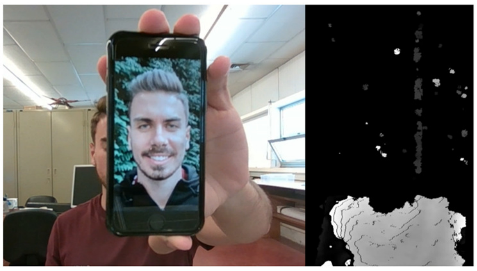

Figure 20- Attempt to fool the algorithm

Hypothetically if someone gained access to a picture of a known person included in the dataset, should that stranger show that same picture to the camera, what would happen is similar to what is portrayed in Figure 20. Even though the subject on the image is present on the dataset, the CNN's do not predict since the depth classifier does not detect a face in the first place. Since the classifier is seeing noise rather than a face, it prevents the system to continue with the prediction. By removing the smartphone from the camera's viewing angle, the algorithm immediately detects a depth face and an RGB face as before, automatically does the predictions, and outputs the prediction result on the screen. This proves the additional security in using depth information in a face recognition system making depth a complement to RGB images.

VI. Conclusions and Future Work

Regarding RGB images, excellent results in prediction were obtained, thanks mainly to the Inception-Resnet V1 architecture already pre-trained using face images from other people. As for depth images, the conclusion reached when training Deep Learning models with depth maps, particularly face depth maps, is that the results can easily differ since these images can be quite ambiguous. Even for humans, it is not obvious at first sight which depth face image belongs to which subject, therefore, one should not rely exclusively on depth face recognition especially if there are matters of security on the line, as these can be at risk. Bearing in mind the overall inferior confidence percentage in the depth model, the outcome is only shown when both models correctly predict the name of the person. As proven, the system developed is capable of precisely recognise subjects within the dataset, and classify, as unknown subjects, people that do not belong to the dataset. Besides this classification, the system has demonstrated the ability to detect when it is being fooled by a smartphone or a printed sheet of paper containing the face of a person, and thus, one can acknowledge this as a reliable 3D face recognition system.

The future of this work aims to perfect the accuracy associated with depth face prediction, since, in comparison with RGB model prediction, one can see that the depth model always predicts with inferior accuracy values, having sometimes 30% inferior percentage of confidence. This may be solved by adding even more images to the dataset and by fine-tuning the network's hyper parameters.

Acknowledgments

This work has been supported by FCT—Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020. In addition, this work has also been funded through a doctoral scholarship from the Portuguese Foundation for Science and Technology (Fundação para a Ciência e a Tecnologia) [grant number SFRH/BD/06944/2020], with funds from the Portuguese Ministry of Science, Technology and Higher Education and the European Social Fund through the Programa Operacional do Capital Humano (POCH).

VII. References

[1] D. Kragic and M. Vincze, "Vision for Robotics," *Foundations and Trends in Robotics*, 2009.

[2] N. Zaeri, "3D Face Recognition," in *New Approaches to Characterization and Recognition of Faces*, InTech, 2011. doi: 10.5772/18696.

[3] J. Luo, F. Hu, and R. Wang, "3D Face Recognition Based on Deep Learning," in *Proceedings of 2019 IEEE International Conference on Mechatronics and Automation, ICMA 2019*, 2019, pp. 1576–1581.

[4] T. Ribeiro, F. Gonçalves, I. S. Garcia, G. Lopes, and A. F. Ribeiro, "CHARMIE: A collaborative healthcare and home service and assistant robot for elderly care," *Applied Sciences (Switzerland)*, vol. 11, no. 16, 2021, doi: 10.3390/app11167248.

[5] F. Schroff and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering."

[6] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition."

[7] D. Kim, M. Hernandez, J. Choi, and G. Medioni, "Deep 3D Face Identification."

[8] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning."

[9] N. Pereira, T. Ribeiro, G. Lopes, and A. F. Ribeiro, "Real-Time Multi-Stage Deep Learning Pipeline for Facial Recognition by Service Robots," *RoboCup Asia-Pacific 2021 Symposium, Aichi, Japan*, Nov. 2021.

[10] G. Sang, J. Li, and Q. Zhao, "Pose-Invariant Face Recognition via RGB-D Images," 2016.

[11] B. Y. L. Li, A. S. Mian, W. Liu, and A. Krishna, "Using Kinect for face recognition under varying poses, expressions, illumination and disguise," *Proceedings of IEEE Workshop on Applications of Computer Vision*, pp. 186–192, 2013.

[12] C. Ciaccio, L. Wen, and G. Guo, "Face recognition robust to head pose changes based on the RGB-D sensor," *IEEE 6th International Conference on Biometrics: Theory, Applications and Systems, BTAS 2013*, 2013.

[13] A. Khan, A.Sohail, *et.al.* "A Survey of the Recent Architectures of Deep Convolutional Neural Networks"