





FaceVision-GAN: A 3D Model Face Reconstruction Method from a Single Image Using GANs

Danilo Avola¹^a, Luigi Cinque¹^b, Gian Luca Foresti²^c and Marco Raoul Marini¹^d

¹Sapienza, University of Rome, Department of Computer Science, Via Salaria 113, 00199, Rome, Italy

²University of Udine, Department of Mathematics, Computer Science and Physics, Via delle Scienze 206, 33100 Udine, Italy

Keywords: GAN, 2D to 3D Reconstruction, Face Syntesis, 3D Modelling from Single Image.

Abstract: Generative algorithms have been very successful in recent years. This phenomenon derives from the strong computational power that even consumer computers can provide. Moreover, a huge amount of data is available today for feeding deep learning algorithms. In this context, human 3D face mesh reconstruction is becoming an important but challenging topic in computer vision and computer graphics. It could be exploited in different application areas, from security to avatarization. This paper provides a 3D face reconstruction pipeline based on Generative Adversarial Networks (GANs). It can generate high-quality depth and correspondence maps from 2D images, which are exploited for producing a 3D model of the subject's face.


1 INTRODUCTION


The increasing computational power of recent computers drives the scientific community to invest in heavy computational approaches, e.g., very deep neural networks or systems of equations with millions of parameters. It often provides improvements in lots of application areas, e.g., human-computer interactions (Avola et al., 2019), UAV self-driving (Avola et al., 2023), or manufacturing (Avola et al., 2022). In this context, recreating a 3D face mesh from images is a challenging task (Toshpulatov et al., 2021). It depends on the quality of the images, the face pose, the presence of occlusions, the lighting condition, or the general environment in which pictures are taken. However, it is a popular task in computer vision, computer graphics, medical treatments, security, and human-computer interaction. Given its central role, many approaches have been used in the past years. First, pair-wise stereo reconstruction has been used, as in (Beeler et al., 2010). It consists of taking several photos from different angles and using them to reconstruct a complete 3D mesh of the subject iteratively. Subsequently, such mesh is refined to introduce fine


texture details. Nowadays, neural networks have become one of the top trending techniques to reconstruct 3D mesh from a single image (Sela et al., 2017; Richardson et al., 2017; Richardson et al., 2016; Isola et al., 2017; Zhu et al., 2015). Most studies apply Convolutional Neural Networks (CNNs) or Image-To-Image translation networks to generate depth and correspondence maps, the 2D representations of spatial information. Starting from them, it is easy to recreate the corresponding 3D model. This work provides a 3D face reconstruction module using Generative Adversarial Networks (GANs) (Kuang et al., 2019; Goodfellow et al., 2014). We trained a complete GAN model whose generator can recreate a correct depth and correspondence map from the original RGB face image. We can generate a complete 3D face model, subsequently enriched with fine texture details. The results show appreciable qualitative details, which could validate the method's effectiveness.


2 DATASET

The dataset used is Labeled Faces in the Wild (LFW)(Huang et al., 2008). It contains 13244 images of 246 subjects with a dimension of 250×250 taken in uncontrolled environments. However, we also needed each image's depth and correspondence map to generate the training set. Since LFW does

^a <https://orcid.org/0000-0001-9437-6217>

^b <https://orcid.org/0000-0001-9149-2175>

^c <https://orcid.org/0000-0002-8425-6892>

^d <https://orcid.org/0000-0002-2540-2570>

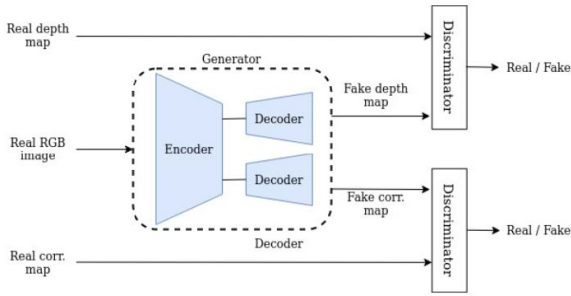


Figure 1: GAN architecture representation. Our generator module comprises a single encoder and a double decoder, while the discriminator is divided according to the two different tasks.

not contain such images, we created them separately from the original dataset. We exploited the method proposed in (Zhu et al., 2017), which, among its features, provides the depth and correspondence map of a given input image¹. This tool is based on a face detection approach, so we could also refine the images by dropping the ones in which zero or more than one face is found. At the end of this process, we obtained a total dataset of 12044 images.

3 PROPOSED GENERATIVE NETWORK

We used a GAN-based architecture to generate a depth and correspondence map given an “into-the-wild” face image. It comprises two models, trained one against the other: the generator tries to generate images that approximate the original training data; the discriminator distinguishes whether the generator creates the given images or belongs to the real ones. An overview of the architecture is shown in Figure 1.

3.1 Architecture of the Generator

Our generator architecture resembles the U-net(Ronneberger et al., 2015) encoder-decoder architecture with some changes. We wanted to simultaneously generate depth and correspondence maps from the same input image. Thus, our network is made up of a single encoder and two different decoders, one for each task. The encoder tries to represent all the useful information to recreate both output images in a lower dimensional latent space. Starting from that encoding, the decoders specialize in recreating a depth or a correspondence map. Following the U-net architectural style, we added skip connections between corresponding encoding

¹<https://github.com/cleardusk/3DDFA>

and decoding layers to share low-level information. It helps the decoder recreate the original pose and some texture details from the face.

3.2 Architecture of the Discriminator

We used a different discriminator for the two tasks. Their architecture is based on PatchGAN (Demir and Unal, 2018), aiming to discriminate $N \times N$ size patches of the output image instead of considering it as a whole. Such an effect is given by a predefined sequence of convolutional kernels, which uses a specific size, padding, and stride. This architecture is powerful because it reduces the number of trainable parameters and speeds up the operation since the input image size is reduced. Our discriminator can consider 70×70 size patches of the generator output image. We also added a skip connection from the original RGB image to the discriminator input to improve its capacity. We used an advanced loss function to speed up the convergence and support the models to store and detect valuable low-level information. The details are provided in Equation 1 and discussed above.

$$\begin{cases} L_G = \min_G \max_D V(D, G) + \lambda_1 L_1(G) + \\ \lambda_g L_1(\nabla(\hat{z}) - \nabla(z)) \\ L_{D_{depth}} = -\min_G \max_{D_{depth}} V(D_{depth}, G) \\ L_{D_{corr}} = -\min_G \max_{D_{corr}} V(D_{corr}, G) \end{cases} \quad (1)$$

For each discriminator, the loss function is given by the original GAN discriminator loss. As regards the generator, its loss is given by the original loss plus the L_1 distance between the original and recreated output image, and it also considers the L_1 distance between the gradients. Each element is replicated to consider depth and correspondence map generation outputs. Some examples of the generated images are provided in Figure 2, where the second row shows the depth images, while the third contains the correspondence maps.

4 3D MESH SYNTHESIS

We could generate a raw 3D triangular mesh from the depth and correspondence maps synthesized by the generator module. Firstly, the X and Y coordinates of each point in the space are extracted from the images. Then, using the Delaunay Triangulation algorithm (Lee and Schachter, 1980), a 2D grid mesh is generated. Finally, every point is raised in 3D space using the Z coordinate encoded in the map.

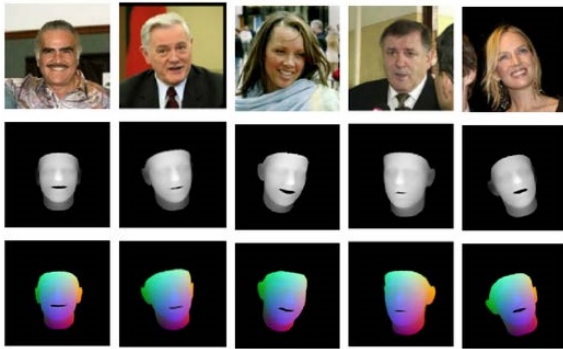


Figure 2: Depth (second row) and correspondence maps (third row) synthesized by the GAN generator from the input RGB face images (first row).

4.1 Mesh Refinement and Smoothing

We perform some refinement operations to improve the 3D mesh quality. Firstly, we removed all the background triangles with a Z coordinate 0. After that, we removed all the triangular spikes, characterized by a bigger area or longer sides. We applied the Laplacian smoothing filter (Badri et al., 2012) once the 3D mesh triangles were refined. This operation consists of slightly moving each vertex position to the centroid of the position of its one-ring neighbors to remove all the fine uneven details.

4.2 Detail Filling

After smoothing the 3D mesh, we performed a detail filling, adding fine detail structures to the actual geometry. Fine details are mainly visible by extracting the high-frequency part of the original face texture. This is executed by applying low-pass filtering and subtracting it from the original image converted to grayscale. Once such features were extracted, each point in the 3D mesh was moved along the direction of its normal small displacement step. The latter is calculated considering the features of the one-ring neighborhood vertices. The final shape of each model is between ~ 10000 and ~ 15000 faces, corresponding to a range of vertices from ~ 5000 to ~ 8000 .

5 EXPERIMENTAL SETUP AND RESULTS

The custom dataset was divided into 11794 images for the training set and 543 images for the test set, according to the suggestions provided by (Joseph, 2022). We trained our model for 50 epochs on depth and correspondence maps, empirically noticing the con-

vergence of the model. We resized the images to a dimension of 128×128 , to fit in our computing resource. The involved hardware for the experiments exploits a high-end CPU Intel Core i7-5930k, 16 GB of DDR4 RAM, and a GTX1070 with 8GB of dedicated RAM. The software is developed in Python, exploiting the PyTorch² framework. In Figure 3, some results of the generated synthetic images by our approach are shown. We noticed that the proposed approach can capture even finer details about the mouth position and the eyes. This quality is due to frequent skipped connections in the network, which maintains low-level features in the decoder layers. Starting from such images, we reconstructed the entire 3D face model and added texture details. Coarse grain face details and model expression resemble the original RGB image, showing that the network correctly synthesized out-depth and correspondence maps. However, fine texture details are not extremely visible. An explanation can rely on our lower 3D mesh tessellation, which is a consequence of the original image scaling operation. The collected results could not be compared with other methods due to the custom dataset exploited in the experimental phase. However, a manual evaluation could be performed (Borji, 2018), mainly based on visual similarity with the original subject. As noticeable in the provided examples, the generated shapes are very close to the original face ones, even if some noise (spikes) and blur are present. To the best of our knowledge, the system could not be directly compared with any other work in the state of the art due to the introduction of the proposed custom dataset generated from LFW. Moreover, the LFW dataset is mainly used for face detection and/or recognition, even when 3D reconstructions are involved (Liu et al., 2018; Jiang et al., 2023). Thus, no metrics in terms of precision, accuracy, or visual appearance have been proposed for this specific task yet.

6 CONCLUSIONS

In this work, we proposed a method for reconstructing 3D models from a single 2D image of a face. First, a generative network synthesizes the depth and correspondence map from the original image. Subsequently, a complete 3D mesh is reconstructed from it, refining the initial raw model, smoothing it out, and restoring some fine details using the original texture. The obtained results seem promising, highlighting an appreciable quality even in a limited-resources environment. Moreover, some future improvements

²<https://pytorch.org/>

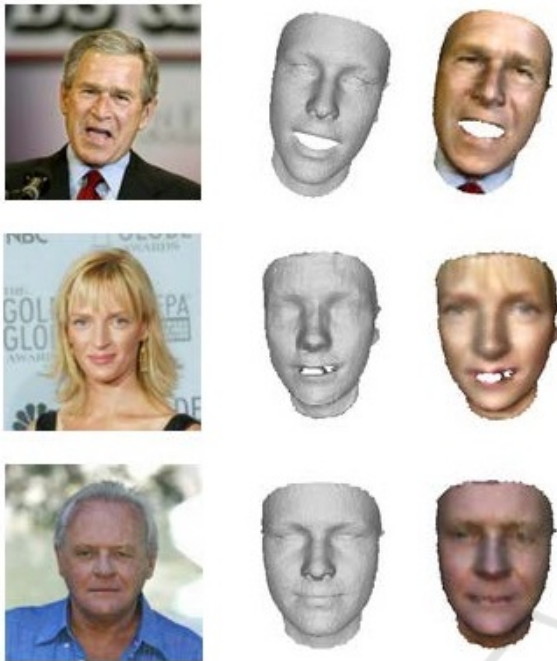


Figure 3: Original images (first column), generated shapes (second column), and refined final output (third column) obtained from the proposed pipeline.

could be assumed. The method seems particularly suitable to provide appreciable results while avoiding heavyweight models, e.g., Transformers (Basak et al., 2022). Also, more powerful hardware could be exploited for testing the method with different hyperparameters and deeper architectures. Furthermore, the details filling phase could be improved by capturing finer texture details in the final geometry to provide a more realistic face model. In conclusion, a comparative analysis will be conducted in the future: exploiting the same procedure to generate the custom dataset, we will collect results after running different well-known approaches on it and compare them with the proposed solution's ones.

ACKNOWLEDGEMENTS

This work was supported by the “Smart unmanned Aerial vehicles for Human like monitoring (SEARCHER)” project of the Italian Ministry of Defence (CIG: Z84333EA0D) and the research leading to these results has received funding from Project “Ecosistema dell’innovazione - Rome Technopole” financed by EU in NextGenerationEU plan through MUR Decree n. 1051 23.06.2022 - CUP H33C22000420001.

REFERENCES

- Avola, D., Cascio, M., Cinque, L., Fagioli, A., Foresti, G. L., Marini, M. R., and Rossi, F. (2022). Real-time deep learning method for automated detection and localization of structural defects in manufactured products. *Computers & Industrial Engineering*, 172:108512.
- Avola, D., Cinque, L., Foresti, G. L., Lanzino, R., Marini, M. R., Mecca, A., and Scarcello, F. (2023). A novel transformer-based imu self-calibration approach through on-board rgb camera for uav flight stabilization. *Sensors*, 23(5).
- Avola, D., Cinque, L., Foresti, G. L., and Marini, M. R. (2019). An interactive and low-cost full body rehabilitation framework based on 3d immersive serious games. *Journal of Biomedical Informatics*, 89:81–100.
- Badri, H., El Hassouni, M., and Aboutajdine, D. (2012). Kernel-based laplacian smoothing method for 3d mesh denoising. In *Image and Signal Processing: 5th International Conference, ICISP 2012, Agadir, Morocco, June 28-30, 2012. Proceedings 5*, pages 77–84. Springer.
- Basak, S., Corcoran, P., McDonnell, R., and Schukat, M. (2022). 3d face-model reconstruction from a single image: A feature aggregation approach using hierarchical transformer with weak supervision. *Neural Networks*, 156:108–122.
- Beeler, T., Bickel, B., Beardsley, P., Sumner, B., and Gross, M. (2010). High-quality single-shot capture of facial geometry. In *ACM SIGGRAPH 2010 papers*, pages 1–9.
- Borji, A. (2018). Pros and cons of gan evaluation measures.
- Demir, U. and Unal, G. (2018). Patch-based image inpainting with generative adversarial networks. *arXiv preprint arXiv:1803.07422*.
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial networks.
- Huang, G. B., Mattar, M., Berg, T., and Learned-Miller, E. (2008). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *Workshop on faces in 'Real-Life' Images: detection, alignment, and recognition*.
- Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134.
- Jiang, D., Jin, Y., Zhang, F.-L., Zhu, Z., Zhang, Y., Tong, R., and Tang, M. (2023). Sphere face model: A 3d morphable model with hypersphere manifold latent space using joint 2d/3d training. *Computational Visual Media*, 9(2):279–296.
- Joseph, V. R. (2022). Optimal ratio for data splitting. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 15(4):531–538.
- Kuang, H., Ding, Y., Ma, X., and Liu, X. (2019). 3d face reconstruction with texture details from a single image

- based on gan. In *2019 11th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pages 385–388. IEEE.
- Lee, D.-T. and Schachter, B. J. (1980). Two algorithms for constructing a delaunay triangulation. *International Journal of Computer & Information Sciences*, 9(3):219–242.
- Liu, F., Zhu, R., Zeng, D., Zhao, Q., and Liu, X. (2018). Disentangling features in 3d face shapes for joint face reconstruction and recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Richardson, E., Sela, M., and Kimmel, R. (2016). 3d face reconstruction by learning from synthetic data. In *2016 fourth international conference on 3D vision (3DV)*, pages 460–469. IEEE.
- Richardson, E., Sela, M., Or-El, R., and Kimmel, R. (2017). Learning detailed face reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1259–1268.
- Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation.
- Sela, M., Richardson, E., and Kimmel, R. (2017). Unrestricted facial geometry reconstruction using image-to-image translation. In *Proceedings of the IEEE international conference on computer vision*, pages 1576–1585.
- Toshpulatov, M., Lee, W., and Lee, S. (2021). Generative adversarial networks and their application to 3d face generation: A survey. *Image and Vision Computing*, 108:104119.
- Zhu, X., Lei, Z., Yan, J., Yi, D., and Li, S. Z. (2015). High-fidelity pose and expression normalization for face recognition in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 787–796.
- Zhu, X., Liu, X., Lei, Z., and Li, S. Z. (2017). Face alignment in full pose range: A 3d total solution. *IEEE transactions on pattern analysis and machine intelligence*, 41(1):78–92.