Multi-View 3D Face Reconstruction in the Wild Using Siamese Networks

Eduard Ramon
Crisalix SA
eduard.ramon@crisalix.com

Janna Escur
Crisalix SA
janna.escur@crisalix.com

Xavier Giró-i-Nieto
Universitat Politècnica de Catalunya
xavier.giro@upc.edu

Abstract

In this work, we present a novel learning based approach to reconstruct 3D faces from a single or multiple images. Our method uses a simple yet powerful architecture based on siamese neural networks that helps to extract relevant features from each view while keeping the models small. Instead of minimizing multiple objectives, we propose to simultaneously learn the 3D shape and the individual camera poses by using a single term loss based on the reprojection error, which generalizes from one to multiple views. This allows to globally optimize the whole scene without having to tune any hyperparameters and to achieve low reprojection errors, which are important for further texture generation. Finally, we train our model on a large scale dataset with more than 6,000 facial scans. We report competitive results in 3DFAW 2019 challenge, showing the effectiveness of our method.

1. Introduction

3D technology is present in many different fields nowadays. We can use it to reconstruct body limbs and create personalised prosthesis, to autonomously navigate in indoor and outdoor environments, or to unlock our phones using our facial anatomy. However, most of the applications require specific hardware to obtain the 3D information about the scene, for instance laser scanners or structured light sensors, which are rarely present in most of the devices used by the mainstream users. Being able to understand the environment that we are surrounded by using only RGB data from ubiquitous video cameras is a challenging problem that could open a whole new range of possibilities.

Approaches based on deep neural networks [12, 6] have been proposed for solving the task of single and multi-view 3D reconstruction. Despite being capable to encode much more prior knowledge than classical techniques, and thus reduce the number of images required, learning 3D reconstruction from one or multiple images is a challenging problem. The scarcity of annotated 3D data is one of the main concerns and it is usually addressed by learning from synthetic data [14, 15, 16] or defining self-supervised losses in the image domain [19, 15, 23]. Another common issue in deep 3D reconstruction, is deciding which 3D data representation is more suitable for a certain problem. Point clouds [4], meshes [22], voxel grids [2] and 3DMM [20] are some of the most used representations and they are a key design criterion. Finally, it is not trivial how to combine information from multiple views in order to satisfy the geometric constrains of the scene and generate better reconstructions as the number of views increases. Recent approaches introduce geometric inductive biases about the scene into the architectures [10] and the losses [13], which constrain the number of possible solutions and ease the learning process.

In this work, we describe a method that participated in the 3DFAW 2019 challenge [8]. We propose an architecture based on siamese neural networks for the task of 3D face reconstruction from one or multiple images, with focus on building a simple, modular and geometrically grounded learning system. Our contributions are:

- A simple and modular architecture based on siamese neural networks that allows learning both single view and multi-view 3D reconstruction.
- A single-term reprojection loss that introduces multi-view geometry to enforce consistency across multiple views.
- The training of 3D reconstruction deep learning models completely supervised by a large scale dataset with more than 6,000 ground truth scans, which allows the comparison between supervised and self-supervised methods.

2. State of the art

2.1. Single view

Methods that aim to predict 3D shapes from a single image usually require stronger inductive biases than multi-view ones. For this reason, it is common to combine deep learning methods and 3D Morphable Models (3DMM) [3], which embed the sub-space of possible solutions into a lower dimensional one. In [14] and [15], a model is trained on synthetic data to regress the shape parameters of a 3DMM. To generalize to real data, Iterative Error Feedback (IEF) [1] is applied in the image domain, which is slow. In order to speed up the process, [9] performs IEF in the latent space. Other methods directly learn 3D reconstruction by defining losses in the image domain [19, 18, 15]. This greatly improves generalization and avoids the need of using IEF. Nevertheless, since no 3D information is available, these methods require strong regularization in their losses, penalizing large norms of the vectors that contain the 3DMM parameters [20]. An alternative regularization technique is the one proposed by [9], which uses an adversarial loss to keep the distribution of the 3DMM parameters plausible. Finally, [5] proposes an unsupervised method to learn to regress 3DMM parameters by enforcing cycle consistency, similarly to CycleGAN [25], and using a differentiable renderer.

2.2. Multi-view

In contrast to single view methods, the multi-view ones can leverage epipolar geometry to introduce more complex biases into the architecture and into the losses. In [10], deep image features are projected into a 3D volume, processed using 3D convolution, and similarly to [24], a multi-view loss is defined in the image domain by projecting the reconstructed 3D geometry and comparing it against masks or depth maps. In [23], the authors propose a simpler way to combine 2D image features by concatenating them. Then, a photometric consistency loss is defined across all views, which is based on multi-view geometry and uses the same differentiable renderer as [5].

Our work resembles to [23]. However, our architecture is grounded on a single view model used as a siamese neural network, making it more flexible in case that frame by frame predictions are required, for instance in augmented reality (AR) applications. Moreover, we do not restrict the multi-view features fusion to concatenation, but study other ways to merge these cues. Finally, we define a single term loss that has no hyperparameters and allows to obtain competitive models faster, since tuning is not necessary.

3. Methodology

We formalize the problem of learning 3D reconstruction as finding the unknown mappings from a set of input images $\{\mathcal{I}_n\}_{n=1}^N$ to a 3D shape $\boldsymbol{s} \in \mathbb{R}^{3P}$, P being the number of points, and to a set of camera poses $\{c_n\}_{n=1}^N$, each one associated to an input image. We express each camera pose as a 3x4 matrix $c_n = [R|\boldsymbol{t}]$, R being the rotation and $\boldsymbol{t}_n = (t_x, t_y, t_z) \in \mathbb{R}^3$ the spatial translation of each camera. We model R as a unit quaternion $\boldsymbol{q} = (q_0, q_1, q_2, q_3) \in \mathbb{H}_1$ to avoid the Gimbal lock effect, which is the loss of one degree of freedom in a three-dimensional mechanism.

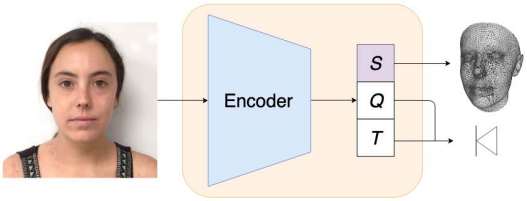

Figure 1: Single view architecture.

3.1. Single view setup

In the single view setup ($N = 1$), we define the three mappings to be learnt as $\mathcal{S}$, $\mathcal{Q}$ and $\mathcal{T}$, which represent three generic functions that map an input image towards a 3D shape, a quaternion and a 3D point respectively. In order to learn them, we make use of a simple architecture formed by an encoder, responsible for extracting image features, and three multilayer perceptrons that act as regressors for $\hat{\boldsymbol{s}}$, $\hat{\boldsymbol{q}}$ and $\hat{\boldsymbol{t}}$, which are the outputs of the network. Figure 1 shows a block diagram of the single view setup.

Note that, since we are using a linear model to represent the 3D shape $\hat{\boldsymbol{s}}$, the mapping $\mathcal{S}$ can be decomposed into two sub-mappings: one that transforms the image to the shape parameters $\hat{\boldsymbol{\alpha}}_{id}$ of the 3DMM, and a second sub-mapping that back-projects the shape parameters to the 3D shape $\hat{\boldsymbol{s}}$. This second mapping is linear and deterministic, and can be expressed as:

$$\hat{\boldsymbol{s}} = \boldsymbol{m} + \Phi_{id}\hat{\boldsymbol{\alpha}}_{id}, \tag{1}$$

where $\boldsymbol{m}$ represents the mean of the 3DMM, and Φ_{id} and $\hat{\boldsymbol{\alpha}}_{id}$ are the identity basis and the predicted identity parameters respectively. So, effectively, $\mathcal{S}$ will only learn the parameters necessary to map $\mathcal{I}$ to $\hat{\boldsymbol{\alpha}}_{id}$.

Learning deep models for single view 3D reconstruction requires strong regularization, since no 3D information is fed into the network. This is often translated on appending multiple terms into the loss together with hyperparameters that help to keep the norm of $\hat{\boldsymbol{\alpha}}_{id}$ small [19]. In order to avoid the use of hyperparameters, we make use of the loss proposed in [13], which allows to simultaneously learn the 3D shape and the camera pose using a sole term expression.

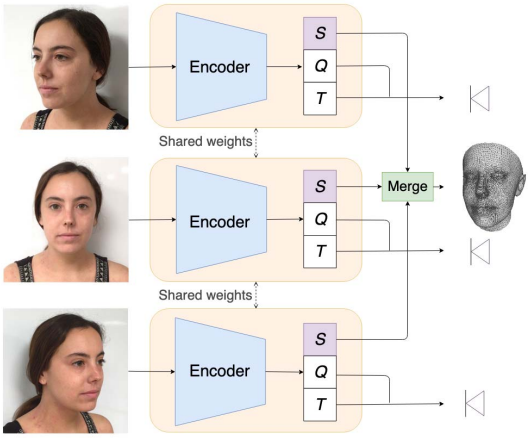

Figure 2: Multi-view architecture.

3.2. Multi-view setup

Our multi-view architecture is composed of two main blocks. The first is the previously described single view architecture that works as a siamese neural network to predict individual camera poses c_n and shape parameters $\hat{\boldsymbol{\alpha}}_{\boldsymbol{id}_n}$ for each view. Then, the N outputs of the shape parameters are fed into a second block that combines them to obtain a global 3D shape, which we call merge block $\mathcal{M}$. The merge block is generic and can be implemented with any operation that aggregates information. Finally, a MLP is used to regress the shape parameters of the 3DMM that will be linearly transformed into the 3D shape using the mapping from Equation 1. We describe this architecture in Figure 2.

Using the single view architecture as the main building block of the multi-view one has several advantages. First, we can train the single view model and use the weights to better initialize the training of the multi-view. The single view model can also be used to predict the target poses that later will be fed into the multi-view architecture. Finally, most of the code can be re-used, avoiding potential bugs.

In order to enforce a global scene consistency in the predictions, we define an objective that uses all the camera poses and the predicted 3D shape within a single term, which does not include any hyperparameter and is easy to minimize, as proposed by [13]. Our loss is defined as the sum of the reprojection errors across all the input views, as in Bundle Adjustment [21], which is the Maximum Likelihood estimator when the image error is zero-mean. Thus, we aim to minimize the following term:

$$\mathcal{L} = \sum_{v=1}^{V} ||\mathcal{P}(\boldsymbol{q_v}, \boldsymbol{t_v})(\boldsymbol{s}_H) - \mathcal{P}(\hat{\boldsymbol{q_v}}, \hat{\boldsymbol{t_v}})(\hat{\boldsymbol{s}}_H)||_2^2, \quad (2)$$

where $\boldsymbol{s}_H$ is the 3D shape in homogeneous coordinates and $\mathcal{P}$ projects any 3D shape $\boldsymbol{s}$ to the 2D image plane, obtaining $\boldsymbol{s}_{2D}$ defined by:

$$\boldsymbol{s}_{2D} = \begin{pmatrix} u'/w' \\ v'/w' \end{pmatrix}, \quad (3)$$

with

$$\left(u'v'w'\right)^T = K[R(\boldsymbol{q})|\boldsymbol{t}]\boldsymbol{s}_H, \quad (4)$$

K being the calibration matrix.

4. Experiments

In this section, we evaluate the performance of the single view and the multi-view models presented in Section 3. We start by describing the dataset we built for training the models. Then, we detail our implementations and, finally, we present the results obtained in the 3DFAW challenge 2019 [8].

4.1. Dataset

Current state of the art methods overcome the scarcity of 3D data by learning from synthetic data [14, 15, 7] or by defining losses in the image domain [19, 18, 15]. Unfortunately, the former ones suffer from poor generalization and the later require strong regularization. In order to avoid these issues, we built a large scale dataset with real images and 3D facial scans.

Our dataset is formed by 6,528 individuals from different gender, age and ethnicity. From each individual, we capture the 3D facial geometry with neutral expression together with a set of images from different angles and the corresponding camera poses. In average, we collect five images per subject. The 3D geometry is acquired using the Structure Sensor scanner from Occipital.

We normalize the data such that all the 3D heads are aligned toward a reference template, which is centered at $\vec{0}$ and facing towards -$\hat{z}$. We split the whole dataset into 70%, 10% and 20% for training, validation and testing respectively. For data augmentation purposes, all the scenes are fully symmetrized.

Finally, we create a 3DMM using the 3D data from the training dataset. First of all, we register a template to each scan in order to have the same topology. Then, the registered templates are aligned using Procrustes Analysis and we apply Principal Component Analysis (PCA) to obtain the identity basis Φ_{id} and the associated eigenvalues Λ, which is the standard procedure.

4.2. Implementation details

As described in Section 3, our single view and multi-view architectures are grounded on a module that processes individual frames, as shown in Figure 1. We implement this module using a VGG-16 image encoder [17] followed by three multilayer perceptrons (MLP) with one hidden layer

of 256 units that regress $\hat{\boldsymbol{\alpha}}_{id}$, $\hat{\boldsymbol{q}}$ and $\hat{\boldsymbol{t}}$. We use 64 shape model modes, which cover the 99% of the 3DMM variance. This module is used for single view inference, an referred as *SV* in the tables.

In order to create the multi-view architecture, we use the single view model as a siamese neural network and we implement the merge block $\mathcal{M}$ using addition and concatenation operations. The aggregated information is then processed by another MLP with also one hidden layer and 256 units. We name these models *MV Add* and *MV Concat*, respectively. Although our architecture could generalizes to any N views, we implement a multi-view model by setting $N = 3$: a frontal one and two laterals views.

All the models have been trained until convergence using Adam optimizer [11] with a learning rate of 10^{-4} and batch size of 32 samples on a NVIDIA RTX 2080 Ti. The training process lasted 13h approximately.

4.3. Evaluation on the 3DFAW 2019 challenge

The proposed single view and multi-view models are assessed on the 3DFAW 2019 challenge [8]. The data is provided in two formats: videos taken with an iPhone, in which the camera moves around the head of the subject, and videos taken with a high resolution camera, in which the camera is static and the subject moves the head to both sides. The metric used to evaluate the 3D accuracy is the Average Root Mean Square Error (ARMSE). We refer the reader to 3DFAW for further details [8].

We use as baseline a model that always predicts the mean of the 3DMM, which we call *3DMM Mean*. In order to evaluate the single view model, we perform two different experiments. The first one consists of predicting the 3D shape using the most frontal frame. The second one performs inference on all the frames and then the predictions are averaged. These two models appear as *SV Frontal* and *SV Mean* in Table 1. Regarding the multi-view setup, we evaluate the two models described in sub-section 3.2, which are named *MV Add* and *MV Concat*. The selection of the frontal and the lateral views was automatically obtained by performing inference with the single view model and keeping only those frames with estimated cameras closer to $\{-45, 0, 45\}$ degrees in the Y axis.

Finally, since our models are designed to estimate shape and pose, we provide a fine-tuned architecture for the task of only 3D shape prediction. We modify the *MV Concat* model by removing the camera pose regressor. Moreover, we add average pooling a the end of the VGG-16 encoder and minimize the MSE error of the 3D shape $\hat{\boldsymbol{s}}$ against the ground truth $\boldsymbol{s}$. We name this last model as *MV Shape-Concat*. Similarly to *SV Mean*, we can boost the performance by averaging multiple predictions of the multi-view model. The best performance was achieved by computing the mean on five predictions, which we name *MV Shape-Concat Mean*.

Model	ARMSE (mm)
3DMM Mean	3.02
SV Frontal	2.62
SV Mean	2.51
MV Add	2.43
MV Concat	**2.33**
MV Shape-Concat	2.23
MV Shape-Concat Mean	**2.14**

Table 1: Performance comparison of the different models in the 3DFAW 2019 challenge [8].

As it can be observed in Table 1, the accuracy of both SV and MV models can be improved by averaging the individuals predictions. In the multi-view setup, using concatenation instead of addition provides better results. Finally, using a specific network for the task of 3D shape regression slightly improves the results, probably because the filters of the encoder can specialize on those features that are more relevant to 3D shape.

5. Conclusions

In this work, we presented a method for learning 3D face reconstruction from a single or multiple images based on siamese neural networks. Our models are simple, modular and, at the same time, capable to obtain highly accurate models. The proposed optimization, based on a single term loss, generates models that are both geometrically consistent across all the 3D scene and does not require fine-tuning any hyperparameter. However, it is not clear whether or not unsupervised losses applied to 3D reconstruction could outperform supervised ones. This is an open question that we leave for future work. Moreover, we empirically showed how merging 3D information can be achieved by simply concatenating the feature vectors extracted by standard encoders such as VGG-16 and that it provides better results than using addition. Finally, using dedicated architectures for the task of 3D shape prediction also provides small gains in accuracy. The design of architectures capable of processing multiple input views and that efficiently extract and merge the 3D information remains a major challenge which we will continue to explore.

6. Acknowledgements

This work has been developed in the framework of the industrial doctorate 2017-DI-028 funded by the Government of Catalonia, and the project TEC2016-75976-R, funded by the Spanish Ministerio de Economía y Competitividad and the European Regional Development Fund (ERDF).

References

[1] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik. Human pose estimation with iterative error feedback. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4733–4742, 2016.

[2] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European Conference on Computer Vision*, pages 628–644. Springer, 2016.

[3] T. F. Cootes and C. J. Taylor. Active shape modelssmart snakes. In *BMVC92*, pages 266–275. Springer, 1992.

[4] H. Fan, H. Su, and L. J. Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613, 2017.

[5] K. Genova, F. Cole, A. Maschinot, A. Sarna, D. Vlasic, and W. T. Freeman. Unsupervised training for 3d morphable model regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8377–8386, 2018.

[6] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[7] Y. Guo, J. Zhang, J. Cai, B. Jiang, and J. Zheng. Cnn-based real-time dense face reconstruction with inverse-rendered photo-realistic face images. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[8] L. A. Jeni, H. Yang, R. K. Pillai, Z. Zhang, J. Cohn, and L. Yin. 3d dense face reconstruction from video (3dfaw-video) challenge. In *2nd Workshop and Challenge on 3D Face Alignment in the Wild Dense Reconstruction from Video (3DFAW-Video) 2019, in conjunction with IEEE International Conference on Computer Vision (ICCV)*, 2019.

[9] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018.

[10] A. Kar, C. Häne, and J. Malik. Learning a multi-view stereo machine. In *Advances in Neural Information Processing Systems*, pages 364–375, 2017.

[11] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[12] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[13] E. Ramon, G. Ruiz, T. Batard, and X. Giró-i Nieto. Hyperparameter-free losses for model-based monocular reconstruction. *arXiv preprint arXiv:1908.09001*, 2019.

[14] E. Richardson, M. Sela, and R. Kimmel. 3d face reconstruction by learning from synthetic data. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 460–469. IEEE, 2016.

[15] E. Richardson, M. Sela, R. Or-El, and R. Kimmel. Learning detailed face reconstruction from a single image. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5553–5562. IEEE, 2017.

[16] J. Roth, Y. Tong, and X. Liu. Unconstrained 3d face reconstruction. *Trans. Graph*, 33(4):43, 2014.

[17] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.

[18] A. Tewari, M. Zollhöfer, P. Garrido, F. Bernard, H. Kim, P. Pérez, and C. Theobalt. Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2549–2559, 2018.

[19] A. Tewari, M. Zollhöfer, H. Kim, P. Garrido, F. Bernard, P. Perez, and C. Theobalt. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *The IEEE International Conference on Computer Vision (ICCV)*, volume 2, 2017.

[20] A. T. Tran, T. Hassner, I. Masi, and G. Medioni. Regressing robust and discriminative 3d morphable models with a very deep neural network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1493–1502. IEEE, 2017.

[21] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustmenta modern synthesis. In *International workshop on vision algorithms*, pages 298–372. Springer, 1999.

[22] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–67, 2018.

[23] F. Wu, L. Bao, Y. Chen, Y. Ling, Y. Song, S. Li, K. N. Ngan, and W. Liu. Mvf-net: Multi-view 3d face morphable model regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 959–968, 2019.

[24] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In *Advances in Neural Information Processing Systems*, pages 1696–1704, 2016.

[25] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.