EFFICIENT AND ACCURATE FACE SHAPE RECONSTRUCTION BY FUSION OF MULTIPLE LANDMARK DATABASES

Pengrui Wang[1,2] *Yi Tian*[1,3] *Wujun Che*[1] *Bo Xu*[1]

[1]Institute of Automation, Chinese Academy of Sciences, China
[2]University of Chinese Academy of Sciences, China
[3] Hunan Normal University, China

wangpengrui2015@ia.ac.cn, tianyi@smail.hunnu.edu.cn, {wujun.che, xubo}@ia.ac.cn

ABSTRACT

We propose an efficient and accurate regression-based 3D face shape reconstruction method. We use an encoder based on MobileNet to estimate parameters including face pose and coefficients of a parametric face model from a single face image. The encoder is trained only by 2D landmarks. Faces can be reconstructed by these parameters. Three contributions of our method are: 1) we propose a databases fusion method to train our network which can easily utilize multiple 2D landmark databases which have different landmark numbers and positions; 2) with the fusion method, we propose a simple MobileNet based network which is efficient, accurate and robust for face reconstruction even without complex training strategies; 3) we add an additional deformation field for shape correction to further improve our network's performance. Experiments demonstrate our method can bring about great performance improvement on most test databases and also compare favorably to some state-of-the-art methods in performance and speed.

***Index Terms*—** 3D Face Reconstruction, Multi Database Fusion, Face Alignment, Deep Learning

1. INTRODUCTION

3D face reconstruction especially using single photograph has drawn a lot of attention in computer vision and graphics in the last decades [1]. It serves as a fundamental task for a wide range of important applications, such as face recognition [2], content creation for games and movies, virtual and augmented reality, communication, and teleconferencing scenarios.

Single image facial reconstruction methods broadly fall into two categories: optimization-based and regression-based. Optimization-based approaches fit a parametric 3D Morphable Models (3DMM) by minimizing the distances between the projected model and the image landmarks detected by face alignment methods [3, 4, 5, 6, 7]. These methods usually have low running speed and rely heavily on alignment methods. Only recently, regression-based approaches are proposed by the power of deep learning [8, 9, 10, 11, 12]. Because ground-truth 3D face scans are costly to acquire and most of them are near-frontal views, researchers usually train neural networks by synthetic image-shape pairs data or only by 2D landmarks. Synthetic 3D face databases are generated either through 3DMM with random textures and lightings [13, 14] or from optimization-based approaches [10, 15]. However, these databases may be inconsistent with actual face images, which leads to their robustness. Besides, some end-to-end methods usually generate noise or even crash shapes [14, 15]. So, post processing is expected to generate realistic face meshes.

For those regression-based methods trained only by 2D landmarks (also called weak-supervised), they generally estimate coefficients of 3D parametric face models from neural networks and use the coefficients to generate 3D face shape [16, 11, 17]. Because the shapes are created from 3DMM, these meshes are realistic and almost impossible to crash when trained with proper parameter regularizations. However, 3DMM constrains faces lying in a low-dimensional subspace and are usually trained from a small number of scans data, which limit the networks' expressivity for modeling accuracy facial shapes. To overcome the problem, some works adopt a trainable shape or add a trainable shape corrections [9, 11]. However, these increase the networks' size and complexity, and more regularization terms need to be designed to train them. In addition, to the best of our knowledge, we find that all of these weak-supervised methods use only one type of landmark database and they usually use face alignment methods to detect new faces to extend training sets. Hence, they miss to exploit the advantages of other existent landmark databases. Although existent manually tagged landmark databases have landmarks varying in numbers and positions, they are more accurate and high-quality than the detected ones. Furthermore, if there is a way to fuse the diverse databases, it may be equivalent to create a larger database with more images samples and landmark positions which is helpful to estimate more accurate and robust 3DMM coefficients.

Responding to these concerns, we propose an encoder-decoder liked structure, trained in a weak-supervised manner, for 3D face shape reconstruction. We propose a way to fuse the various existent 2D landmark databases and use the fusion database to train our network. The main idea is to find the correspondences between all the landmarks in the multiple databases with fixed positions and the vertices on 3DMM's average face. To compensate for 3DMM's expressivity, we also propose adding a shape correction model which is a medium-scale 3D deformation field. To let our model run fast, we choose to use MobileNet [18] based network as our encoder whose model size is a really small. Hence our reconstruction method is efficient and accurate. Our encoder first estimates pose, identity and expression coefficients of a parametric face model and coefficients of a shape correction model. Then, our no training required (model-based) decoder reconstructs the final face shape. So, our method deserves to generate more realistic 3D face than the end-to-end methods PRN [15] and N3DMM [9] which have blur and grid-like meshes and may have crashes on mesh boundaries and nose areas.

In summary, this paper has three main contributions:

This work is supported by National Key R&D Plan of China(No. 2017YFB1002804) and National Natural Science Foundation of China (No. 61471359).

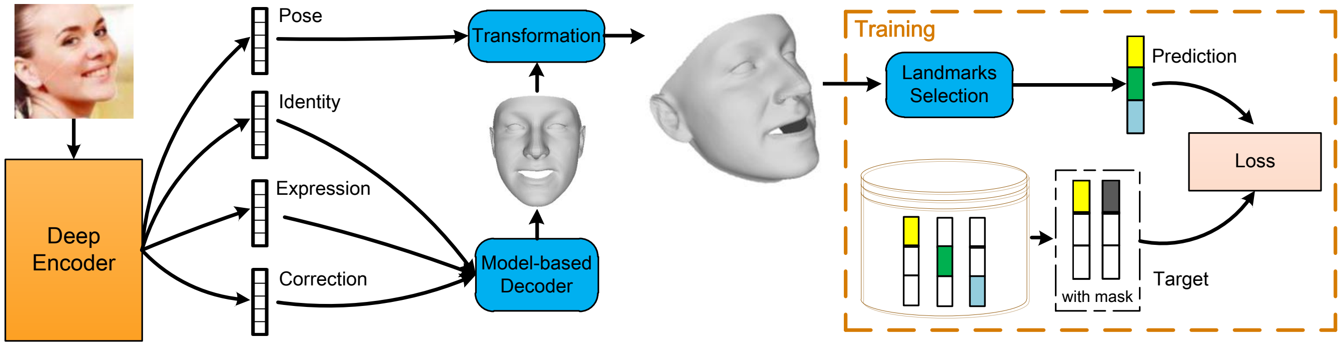

Fig. 1. Pipeline overview. Given an input image, an encoder first forwards it to get the facial pose, coefficients of 3DMM and coefficients of our correction model and then uses these parameters to generate the final 3D face shape. In training process, the corresponding 2D landmarks according to all the used databases need to be selected from the 3D shape firstly. A training loss is to minimize the differences between the ground-truth landmarks from the fusion database and the selected landmarks where the unused landmarks are masked.

- A 2D landmark database fusion method which can utilize the various existent 2D landmark databases to train a 3D face reconstruction neural network.
- A MobileNet based neural network for face reconstruction which runs fast and has high performance after training by the fusion database.
- An additional shape correction module based on a deformation field to rich the shape expressivity and get better performance.

2. VARIETY OF DATABASES IN EXISTENCE

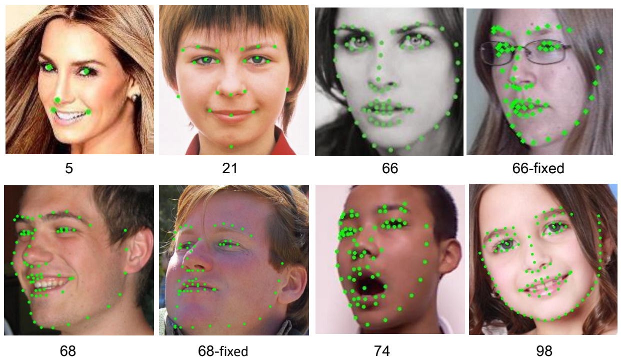

Fig. 2. Examples of different types of landmark database.

There are a lot of 2D landmark databases for face alignment and face reconstruction tasks. We list parts of them in Fig. 2. We can see that these databases have landmarks varying in numbers and positions. What's more, databases without marking with "fixed" also have changeful positions on the face, e.g. landmarks on cheeks. These problems lead to the databases are too difficult to be merged directly and be used together for training. However, a single database, especially the manually tagged high-quality one having fewer samples, is hard to train accurate deep neural networks. So, if these databases are merged in a feasible way, the resultant fusion database will have more samples with more landmarks and should get better performance for the face reconstruction approaches trained only by 2D landmarks.

Table 1. Types, training and test sample numbers and sources of the used databases in this work.

Database	Type	Train	Test	Source
AFLW	(up to) 21	-	21080	[19]
300W	68	3837	-	[20]
300W-LP	68-fixed	122450	-	[21]
AFLW2000-3D	68-fixed	-	2000	[21]
LS3D-W Balanced	68-fixed	-	7200	[22]
LS3D-W 300VW-3D	68-fixed	95192	-	[22]
FaceWarehouse	74	5904	-	[23]
LFW-74	74	5258	2000	[24], web
WFLW	98	7500	2500	[25]

We introduce the used databases for training and test in this work. Their main information is shown in Tab. 1, then we roughly describe them:

AFLW: AFLW [19] contains in-the-wild faces with large pose variations. Each image is annotated up to 21 visible landmarks.

300W: It is a standard multiple face alignment databases including AFW, LFPW, HELEN, and IBUG [20].

300W-LP: A synthesized database generated by face profiling [21]. In this work, we use its augmented database in [26] and also split it into the training set (636252) and the validation set (51602).

LS3D-W Balanced: It has an equal number of images in yaw angles $[0° - 30°]$, $[30° - 60°]$ and $[60° - 90°]$ [22].

LS3D-W 300VW-3D: It comes from [22]. In this paper, we only use its training set which has 50 videos with 95192 frames.

FaceWarehouse: Facewarehouse is a facial expression database and aims at high-performance facial image animation [23].

LFW-74: LFW-74 is collected from the web to add in-the-wild faces for 74 landmark databases. Its faces are from the LFW database [24].

3. METHODS

This section describes our reconstruction method. An overview of our approach is shown in Fig. 1, which also has an overall introduction. Our deep encoder which is an efficient model encodes the facial pose, identity, expression and correction parameters at the same time. Its architecture is described in Sec. 3.3. The encoded parameters are utilized in a model-based decoder to generate the final face shape later. The model-based decoder includes the 3DMM and correction model described in Sec. 3.1 and Sec. 3.2 respectively. The way to train our network with fusion databases is shown in Sec. 3.4.

3.1. Basic Decoder Model and Pose Transformation

We choose the same Basel Face Model used by Tran et al. [12] as our 3DMM. 3DMM represents an individual's 3D face as follow:

$$S = S_0 + A_{\text{id}}\mathbf{x}_{\text{id}} + A_{\text{exp}}\mathbf{x}_{\text{exp}},$$

where $S_0 \in \mathbb{R}^{3N}$ is an average face shape, N is the number of vertices, A_{id} and A_{exp} are Principal Component Analysis (PCA) bases for identity and expression, and $\mathbf{x}_{\text{id}} \in \mathbb{R}^{100}$ and $\mathbf{x}_{\text{exp}} \in \mathbb{R}^{29}$ are control coefficients. We assume a face in 2D image plane is projected from a 3D face by Weak Perspective Projection just as many works adopted [9, 21, 12]. Hence, we define the pose parameters as $\mathbf{x_{pose}} = [s, R, \mathbf{t}_{2\text{d}}] \in \mathbb{R}^6$ where s is the scale, $R \in SO(3)$ is the rotation matrix represented by Lie algebra and $\mathbf{t}_{2\text{d}} \in \mathbb{R}^2$ is the 2D translation vector. The final face shape transformed by pose is,

$$S_T = sS_{3d}R^T + [\mathbf{t}_{2\text{d}}, 0],$$

where $S_{3d} \in \mathbb{R}^{N\times 3}$ is reshaped by S.

3.2. Corrective Shapes

The 3DMM restricts the facial identity and expression to a small dimensional linear subspace. The total dimension is 129 in this work. Thus variations face shapes falling outside of this low subspace cannot readily be expressed. In the spirit of Garrido et al. [5], we adopt a medium-scale 3D deformation filed as our corrective shape model:

$$S_{\text{c}} = E_c\mathbf{x}_{\text{c}}.$$

Here, $E_c = [H_1 \otimes I_{3\times3}, ..., H_k \otimes I_{3\times3}] \in \mathbb{R}^{3N\times3K_c}$ contains three copies the K_c linear Manifold Harmonics basis functions $H_k \in \mathbb{R}^N$ as columns [5] and $\mathbf{x}_{\text{c}} \in \mathbb{R}^{3K_c}$ is the control coefficients. H is the K_c lowest frequency eigenvectors of the cotan-weights Laplace Beltrami operator matrix on the average face S_0. So the individual shape after correction is $S' = S + S_c$. In this work, we set $K_c = 20$.

3.3. Architecture of Encoder

To have a fast runtime speed, our deep encoder adopts an architecture based on MobileNet-V1 [18] which has a small model size (about 13 MB) and less computing workload owe to the deep-wise convolutions. In more detail, our encoder first uses 13 deep-wise based blocks after a convolution layer, which is the same as in [26]. Next, above the blocks, a fully connection layer with output dimensions $|\mathbf{x}| = 64 + 100 + 29 + 20 * 3$ is used to output most of the required parameters. Then, the first 64 parameters are input to another fully connection layer to output the pose parameters. Finally, these parameters are used to generate the final face shape through the model-based decoder. In this work, the input image size is 128×128.

3.4. Database Fusion and Training Strategy

Our method of merging the diverse landmarks databases for training is intuitive. The idea is to make unified predictions and unified targets to avoid considering the training samples' sources and diversity. To achieve this, we first determine a landmark set in which each landmark appears in any of the multiple databases. For validity and simplicity, the selected landmarks must satisfy the following three rules: (1) the landmarks in "fixed" type databases are all reserved; (2) the landmarks which are varying on the 3D face model are removed, e.g. landmarks on cheeks of "68" and "98"; (3) the landmarks which are hard to be represented on the 3D face model are removes, e.g. landmarks on the center of mouths or eyes. Next, because each of the determined landmarks can find a corresponding vertice on 3D face, prediction landmarks can be selected from the first two columns of prediction shape S_T, we denote them as $\mathbf{y}_{\text{lm}} \in \mathbb{R}^{2N_{\text{lm}}}$. Then, a unified vector $\mathbf{t_{lm}} \in \mathbb{R}^{2N_{\text{lm}}}$ corresponding with $\mathbf{y}_{\text{lm}}$ is defined as the new target. In each sample of each database, the values in the new target whose corresponding landmarks also appear in the original target should be set with the original target values. A mask index vector $\mathbf{t_{mask}} \in \mathbb{R}^{2N_{\text{lm}}}$ is adopted to mask out the unset positions. Finally, we equally get a fusion database and the training loss without considering the differences of databases is,

$$\begin{aligned}\text{Loss} =& \|(\mathbf{y}_{\text{lm}} - \mathbf{t}_{\text{lm}}) \circ \mathbf{t}_{\text{mask}}\|_2^2 / \sum_{i=1}^{N_{\text{lm}}} t_{\text{mask}}(i) + \\ & w_{\text{id}}\|\mathbf{x}_{\text{id}}\|^2_{W_{\text{id}}} + w_{\text{exp}}\|\mathbf{x}_{\text{exp}}\|^2_{W_{\text{exp}}} + w_{\text{c}}\|\mathbf{x}_{\text{c}}\|^2_{W_{\text{c}}},\end{aligned} \tag{1}$$

where $\circ$ represents the element-wise multiplication, w is weight value and the notation $\|\mathbf{x}\|_W$ is a shorthand for $\sqrt{\mathbf{x}^TW\mathbf{x}}$. The last three items in Equ. 1 are regularizations where for example, $W_{\text{id}} = diag(\delta^2_{\text{id}})^{-1}$, the δ^2 are the eigenvalues from the PCA model or the Eigenvalue decomposition.

4. EXPERIMENTS

In this part, we first describe experiment protocols in Sec. 4.1. Next, we evaluate the performances of our proposed method in different experimental conditions in Sec. 4.2. Then, we compare our method with some state-of-the-art methods in Sec. 4.3. Finally, we make a discussion.

4.1. Protocols

Datasets: We first define all fusion databases used for training: DB-A (300W-LP Train); DB-B (DB-A, FaceWarehouse and LFW-74 Train); DB-C (DB-B and WFLW Train); DB-D (DB-C and LS3D-W 300VW-3D). 300W-LP's validation set is the only one used for super parameter adjustment in all of our experiments. Besides, considering the rules in Sec. 3.4, not all the landmarks in the databases are used. We remove the unsuitable landmarks, and then database type "68" has 52 landmarks, "74" has 59 landmarks, "98" has 66 landmarks and "68-fixed" still has 68 landmarks.
Experiment setup: Here, we describe some training details. Experiments on DB-B, DB-C or DB-D are all pre-trained with DB-A. Furthermore, the sample number of each sub-database in the same fusion set has great differences, for example, the sample number of 300W-LP Train is nearly 90 times the sample number of WFLW Train. To balance the sub-databases, we randomly select samples on the larger sub-databases and thus make the sub-databases balanced in each training epoch. Besides, we augment the samples by image rotations, flips and face bounding box perturbations. We set $w_{\text{id}} = 0.1$, $w_{\text{exp}} = 1$ and $w_{\text{c}} = 10$ in this work.
Evaluation Criteria: Although our work is about face reconstruction, the reconstructions are coarse because our method is based on statistical models and only uses the 2D landmarks as supervision. Therefore, a more appropriate way to evaluate our work is through the performance of face alignment. The alignment accuracy is evaluated by the Normalized Mean Error (NME), which is the average

Table 2. Evaluation results (NME %) on different databases with different training fusion databases and model types.

Database	Use	AFLW				AFLW	LFW-74	WFLW	Balanced				AVG
	Correction	0-30	30-60	60-90		2000-3D	Test	Test	0-30	30-60	60-90		(mean)
A	N (No)	4.28	5.01	5.67	4.99	3.73	3.57	4.69	2.44	2.64	4.00	3.03	4.00
	Y (Yes)	4.21	4.95	5.71	4.95	3.80	3.51	4.58	2.36	2.59	4.11	3.02	3.97
B	N	4.12	4.94	5.85	4.97	3.95	2.66	4.18	2.35	**2.50**	4.07	2.98	3.75
	Y	4.07	4.88	5.70	4.89	3.84	2.72	4.17	2.42	2.57	4.03	3.01	3.72
C	N	4.03	4.80	5.80	4.88	3.95	2.57	3.15	2.54	2.72	4.05	3.10	3.53
	Y	4.03	4.69	5.60	4.77	3.85	2.55	3.15	2.57	2.71	3.96	3.08	3.48
D	N	4.00	4.70	5.64	4.78	3.83	**2.52**	3.06	**2.31**	**2.50**	3.84	**2.88**	3.41
	Y	**3.99**	**4.64**	**5.59**	**4.74**	**3.72**	2.53	**2.97**	2.34	2.55	**3.79**	2.89	**3.37**

of landmarks error normalized by face size d:

$$\text{NME} = \frac{1}{N}\sum_{i=1}^{N}\frac{\|t_i^* - y_i\|_2}{d},$$

where N is the number of valid landmarks, t is the ground-truth landmarks and y is the estimated landmarks. The face size is defined as the $\sqrt{\text{width} \times \text{height}}$ of the bounding box.

4.2. Ablation Study

To verify the effects of the databases fusion and the correction model, we design a lot of experiments and evaluate them on different test sets. The results are in Tab. 2, where the best result in each category is highlighted in bold, the lower is the better. Moreover, for some databases, faces with different yaw angles are also reported. At the last column, the average of these mean results are also shown.

From the Tab. 2, we can firstly see that with the increases of databases the performances become better on most of the test sets. Secondly, we find that compared with the no correction models, models with the corrections can further improve the performances. Thus most of the best result on each test set occurs in DB-D,Y. Finally, from the results of WFLW Test ("98"), we see that training datasets including "98" sets, like DB-C and DB-D, have much better performances than those without, like DB-A and DB-B. A similar phenomenon also appears in the LFW-74 Test. These demonstrate that different types of databases fusion can not only improve the performances but also make the deep encoder more robust and have a better generalization. So, with database fusion and correction model, our method is accurate and robust.

4.3. Comparison Experiments

Table 3. Performance comparison with state-of-the-art methods 3DDFA [26], [21] and PRN [15].

		3DDFA	3DDFA	PRN	DB	DB
		[26]	[21]	[15]	-A,N	-D,N
AFLW	0-30	4.38	4.11	-	4.28	**3.99**
	30-60	5.26	**4.38**	-	5.01	4.64
	60-90	6.28	**5.16**	-	5.67	5.59
	Mean	5.30	**4.55**	-	4.99	4.74
AFLW2000-3D		4.09	3.79	-	3.73	**3.72**
Balanced	0-30	2.46	-	2.77	2.44	**2.34**
	30-60	2.66	-	2.78	2.64	**2.55**
	60-90	4.14	-	4.30	4.00	**3.79**
	Mean	3.08	-	3.28	3.03	**2.89**

We compare our method with state-of-the-art face alignment / reconstruction methods 3DDFA [26], [21] and PRN [15], see Tab. 3. Here, "DB-A,N" and "DB-D,Y"" means our method. "DB-A,N" has nearly the same network structure and training sets with 3DDFA [26]. However, "Ours-A,N" trained with data augments in a weak-supervised manner. 3DDFA is trained using their Weighted Parameter Distance Cost (WPDC) cost function in a supervised manner [21]. 3DDFA [26] and PRN are tested using their open source codes and models. Results of 3DDFA [21] come from their paper.

The results in Tab. 3 shows that "DB-D,Y" can match the performances of these state-of-the-art methods on the three test sets and on different yaw angles, but "Ours-A,N" does not show the advantages. This demonstrates that it is our database fusion method and correction model that brings the high performances. It is worthwhile pointing out that 3DDFA [21] is a cascaded method for large pose face alignment. It has elaborate and complicated networks and training strategies while our network is really light and with only simple training strategies. Even though, we get matchable performance.

Besides, our reconstruction method runs fast, nearly over 1000 fps on a GeForce GTX 1080Ti GPU, while PRN runs at 111fps on a GeForce GTX 1080 GPU[15].

4.4. Discussion

We use $K_c = 20$ for our correction model in the above experiments. We also try to use smaller or larger K_c, but it doesn't seem to bring about a performance gain. Because the correction model will add another $3K_c$ output coefficients, a relatively small K_c is sufficient.

Our method reconstructs 3D face by the estimated parameters and the model based decoder. The estimated parameters are predicted according to the landmarks in essence. Hence, positions of landmarks on 3D face need to be manually marked as accurately as possible to generate more realistic 3D face shapes.

5. CONCLUSION

In this paper, we propose a regression-based method for face shape reconstruction, which is efficient, accurate and robust. It uses a deep encoder based on MobileNet to estimate parameters and trained only by 2D landmarks. We propose using a database fusion method which merges many 2D landmark databases with landmarks varying in numbers and positions to train our deep encoder. We also propose a corrective shape model to further improve the reconstruction performance. Experiments tested on many datasets show that our database fusion method and correction model are important for reconstruction performances even using a small deep encoder.

In the future, we will design a more appropriate method to use the landmarks appear in multiple databases as much as possible even though their positions on 3D face are not fixed. Furthermore, we will focus on fine detail face reconstruction based on this work.

6. REFERENCES

[1] M. Zollhöfer, J. Thies, P. Garrido, D. Bradley, T. Beeler, P. Pérez, M. Stamminger, M. Nießner, and C. Theobalt, "State of the art on monocular 3d face reconstruction, tracking, and applications," in *Computer Graphics Forum*, 2018, vol. 37, pp. 523–550.

[2] X. Zhu, Z. Lei, J. Yan, D. Yi, and S.Z. Li, "High-fidelity pose and expression normalization for face recognition in the wild," in *CVPR*, 2015, pp. 787–796.

[3] L. Jiang, J. Zhang, B. Deng, H. Li, and L. Liu, "3d face reconstruction with geometry details from a single image," *ToIP*, vol. 27, no. 10, pp. 4756–4770, 2018.

[4] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," in *CVPR*, 2016, pp. 2387–2395.

[5] P. Garrido, M. Zollhöfer, D. Casas, L. Valgaerts, K. Varanasi, P. Pérez, and C. Theobalt, "Reconstruction of personalized 3d face rigs from monocular video," *ACM ToG*, vol. 35, no. 3, pp. 28, 2016.

[6] C. Cao, D. Bradley, K. Zhou, and T. Beeler, "Real-time high-fidelity facial performance capture," *ACM ToG*, vol. 34, no. 4, pp. 46, 2015.

[7] A. E. Ichim, S. Bouaziz, and M. Pauly, "Dynamic 3d avatar creation from hand-held video input," *ACM ToG*, vol. 34, no. 4, pp. 45, 2015.

[8] S. Yamaguchi, S. Saito, K. Nagano, Y. Zhao, W. Chen, K. Olszewski, S. Morishima, and H. Li, "High-fidelity facial reflectance and geometry inference from an unconstrained image," *ACM ToG*, vol. 37, no. 4, pp. 162, 2018.

[9] L. Tran and X. Liu, "On learning 3d face morphable model from in-the-wild images," *arXiv:1808.09560*, 2018.

[10] N. Chinaev, A. Chigorin, and I. Laptev, "Mobileface: 3d face reconstruction with efficient cnn regression," *arXiv:1809.08809*, 2018.

[11] 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," *arXiv:1712.02859*, vol. 2, 2017.

[12] A. Tran, T. Hassner, I. Masi, E. Paz, Y. Nirkin, and G. Medioni, "Extreme 3d face reconstruction: Seeing through occlusions," in *CVPR*, 2018, pp. 3935–3944.

[13] E. Richardson, M. Sela, R. Or-El, and R. Kimmel, "Learning detailed face reconstruction from a single image," in *CVPR*, 2017, pp. 5553–5562.

[14] M. Sela, E. Richardson, and R. Kimmel, "Unrestricted facial geometry reconstruction using image-to-image translation," in *ICCV*, 2017, pp. 1585–1594.

[15] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou, "Joint 3d face reconstruction and dense alignment with position map regression network," *arXiv:1803.07835*, 2018.

[16] A. Tewari, M. Zollhöfer, H. Kim, P. Garrido, F. Bernard, P. Pérez, and C. Theobalt, "Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction," in *ICCV*, 2017, vol. 2, p. 5.

[17] F.J. Chang, A.T. Tran, T. Hassner, I. Masi, R. Nevatia, and G. Medioni, "Faceposenet: Making a case for landmark-free face alignment," in *ICCVW*. IEEE, 2017, pp. 1599–1608.

[18] A.G Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[19] M. Koestinger, P. Wohlhart, P.M Roth, and H. Bischof, "Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization," in *ICCVW*. IEEE, 2011, pp. 2144–2151.

[20] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in *ICCVW*, 2013, pp. 397–403.

[21] X. Zhu, Liu X., Z. Lei, and S.Z Li, "Face alignment in full pose range: A 3d total solution," *TPAMI*, vol. 41, no. 1, pp. 78–92, 2019.

[22] A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks)," in *ICCV*, 2017, vol. 1, p. 4.

[23] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "Facewarehouse: A 3d facial expression database for visual computing," *TVCG*, vol. 20, no. 3, pp. 413–425, 2014.

[24] G.B Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[25] W. Wu, C. Qian, S. Yang, Q.n Wang, Y. Cai, and Q. Zhou, "Look at boundary: A boundary-aware face alignment algorithm," in *CVPR*, 2018, pp. 2129–2138.

[26] X. Zhu J. Guo and Z. Lei, *3DDFA*, 2018, https://github.com/cleardusk/3DDFA.