

Fusing Human Faces

Xianxiang Wang, and Xuejin Chen

University of Science and Technology of China, 230031 Hefei, China
kowo@mail.ustc.edu.cn, xjchen99@ustc.edu.cn

Abstract. Face fusion technology grows more important nowadays. In this paper, We will introduce a method that could fuse features of source face into target face, and target face would still looks natural with its background. Our method contains two steps. First, aligning two faces according to facial landmarks, for which could give us size, location and distribution of features of two faces. Segmenting two faces into several triangles using Delaunay triangulation, warping the corresponding triangles of two faces into the same shape. Second, reconstructing gradients for new face that we are going to generate, with one face image as target image and reconstructed gradients, we could generate fused face by seamless cloning. There are three ways we tried to reconstruct the gradients, first, composing new gradients by linearly combining gradients of two faces, second, preserving the larger gradients of two faces, third, constructing new gradients by preserving weaker gradients which is proposed by us.

Keywords: Face fusion, Facial landmarks, Delaunay triangulation, Seamless clone

1 Introduction

These years, there is a great need for face editing application. Someone would wonder what would they look like if they have some face features from pop stars, they may make their faces sharper through face editing application. While some other people want to know what would they look like when they grow older. these tasks could be accomplished by face fusion. Progeny appearance prediction is a popular task lately, what if the son of a young couple looks like when he grows up, it is attracting for many people, and this could also be accomplished by face fusion technology.

As we know, different faces have different features, someone has a shaper face while the shape of other faces are closer to square. There are many ways to build correspondence of two faces. In early years, hand marked features of faces are used to build correspondence of two faces [1], because of the lack of the technology to detect face features automatically, when we align two faces using this method, there are a lot of labour works to do. Most people do not want to mark the features manually. Lately, a five feature points based face fusion method [4] was proposed, the five feature points are detected automatically, this could really save a lot of time for us, then with these feature points, two faces

could be warped and fused into one. In fact, five feature points are not enough to represent a face, when we use the sparse correspondence to generate a new face, the result is weird.

Given two faces, the face we want to extract its features and fuse its features into the other face named source face, the face which keeps its background and let source face features fuse into self named target face. There are 68 facial landmarks are detected by using an Ensemble of Regression Trees [15] in our method, thus we know exactly the features and size of the face. Facial landmarks indicate the face structures, in other words, the distribution of face features, while the face gradients indicate the face details. When we fuse two faces, first, we warp two faces to make sure the structures of two faces are the same. Second, we fuse the gradients of two faces, in order to maintain the main features of the source face and illumination information of the target face, we reconstruct the gradients for new face by keeping the larger gradients of the source face and part of lower gradients of the target face, then, we could recover the fused face by seamless cloning [3].

2 Related work

In this section, we summarise existing approaches that is similar to our work.

Face Morphing. Face morphing is commonly referred as the animated transformation of one digital face image to the other, in most case, the background of the face image is pure in order to make us focus on transition of the face itself.

Beier and Neely [1] developed user interface to build the correspondences of two faces by a group of line pairs, the procedure of build correspondences of two faces is tedious and time-consuming, besides, hand-marked features are not precise, after building correspondences, the corresponding pixels are linearly interpolated. Wolberg [2] proposed a mesh-based method to build the correspondences of two faces, the correspondences of two faces are referred as meshes instead of line pairs. Karungaru et al. [4] detect five feature points as control points, then segment face images into several triangles according to control points, next warp the face images according to corresponding triangles. These face morphing methods have a different way in building correspondences, Their aim is creating a seamless transition from one face to the other.

Face Swapping. Face swapping is also known as face replacement, means transferring a face from a source photo onto a face appearing in a target photo in order to generate a new genuine face.

The most early work for face swapping is [5], but the results are not that realistic. Later, the results are better [7][8][9][14] and mainly used for deidentification. The main problem for face swapping is pose variation, most works solve this problem by using 3D models [9][10][17]. Nirkin et al. [17] transfer the face from the source image onto the target image using a 3D face model, they first detect facial landmarks used to establish 3D pose and facial expression for 3D face shape, then use a FCN to segment the visible parts of faces from their context

and occlusions, at last, source face is blended-in with the target context using image seamless cloning [3], this step would fail when blending very different facial hues. Korshunova et al. [16] propose a network to generate a source face that could replace the face in target image, one network could only generate faces of the same type, for example, a net named CageNet could only generate the face of Nicolas Cage. Those two methods mentioned above may greatly change the lightning condition of the face on the target image because they just transfer the source face onto the target image, but not containing any information of the target face. Bitouk et al. present a system for automatic face replacement in images, given a target face image, the system would retrieve a face with similar pose, lightning and color condition, then use such a face to replace the face on the target image, the limitation of the source face is clear, it must have a similar pose and lightning condition with the target face. The main difference between face swapping and our task is that we want keep some features of the target face but not to replace the target face with the source face, how to fuse the features of the source faces into the target faces is our main consideration.

3 Fusing features of the source face into the target face

Our task is to fuse features of source face into target face, there are two steps to follow.

First, aligning two faces, it is necessary to calculate the location and the size of the facial part from each face image. With the help of the facial landmarks [15], it is quick to do so. But only knowing the location and size of two faces is not enough, we need to know where are the eyes, nose and other parts of the face, different faces have different face features, some people have a big nose while others with a small one, some people's nose is closer to eyes while others' nose is farther to eyes. Still, with the help of facial landmarks, we know exactly the distribution of the face features.

Second, by aligning two faces, we know the parts on one face and its corresponding parts on the other face. Then, we could fuse features of one face into other face from part to part.

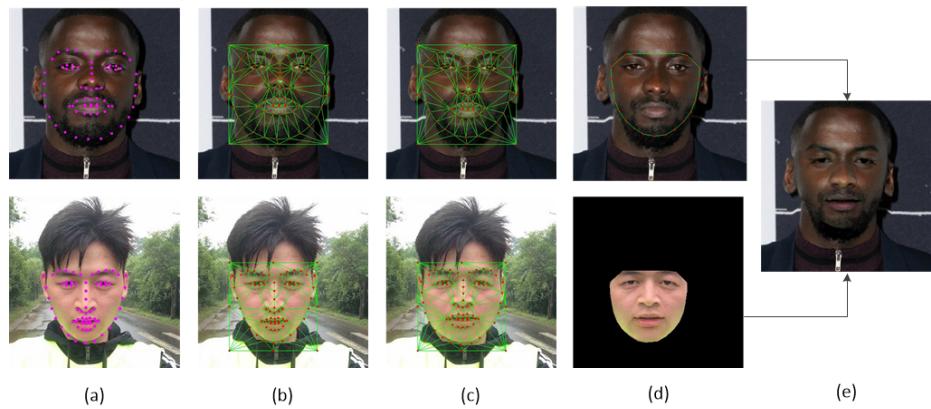


Fig. 1. Pipeline of our work: (a) Perform facial landmark detection on two faces. (b) Triangulate two faces according to the facial landmarks. (c) Calculate new facial landmarks by interpolating two groups of facial landmarks, warp two faces according to the new facial landmarks. (d) Reconstruct gradients on facial part for fused face . (e) Fuse features of source face into target face by seamless cloning.

3.1 Face alignment

There are many ways to find facial landmarks [11][12][13][15], we adopt [15], it is implemented in Dlib.

In Fig. 1(a), with the landmarks of two faces, two faces' bounding box could be calculated, as we talked before, different faces have different facial landmarks, so the source face's bounding box and the target face's bounding box could have different ratio of height and width. In order to align two faces easily, we reshape the bounding boxes of two faces into square by stretching the shorter side of the bounding box, and we will get a bounding square of face. Then, we need to enlarge the bounding squares of two face a little for the consensus of triangulation of two faces as shown in Fig. 1(b). The center point and side length of bounding square of source face are denoted as c_s and l_s , the center point and side length of bounding square of target face are denoted as c_t and l_t . The index of facial landmarks $i \in \{1 \dots N\}$, N is the number of facial landmarks, for each point p_i inside the bounding square of source face, we could calculate its corresponding location on the target face by following mapping function:

$$f(p_i) = (p_i - c_s) \cdot \frac{l_t}{l_s} + c_t \quad (1)$$

Before we triangulate a face according to facial landmarks, we need make sure that the transition from the background to the face is natural, so we add some points into the facial landmarks to help triangulate the faces, these points are four corners of bounding squares and four midpoints on four sides of bounding squares. The facial landmarks we need to triangulate on the source face and the target face are denoted as F_s (the red points on the face image in second row of Fig. 1(b)) and F_t (the red points on the face image in first row of Fig. 1(b)) respectively. The facial landmarks that we want could be calculated by interpolating F_s and F_t and are denoted as F_n , with these new facial landmarks, we could create a new face structure. for each point in F_s , F_t and F_n , they are denoted as F_{s_i} , F_{t_i} and F_{n_i} respectively.

$$F_{n_i} = \alpha \cdot f(F_{s_i}) + (1 - \alpha) \cdot F_{t_i} \quad (2)$$

α controls which face structure we prefer more, if α is small, the new face structure would look more like the target face, in other words, the profile of the new face and the distribution of the new face features would look more like the target face.

After triangulating two faces according to F_s and F_t using Delaunay Triangulation, we get two groups triangles from the source face and the target face,

denoted as T_s and T_t respectively. The two faces corresponding triangles are not of the same shape because of different facial landmarks. what we need to do is to make sure the shape of corresponding triangles of two faces are the same. we could get T_n by triangulating F_n . Warping T_t to the T_n , and warping T_s in a similar way. In Fig. 1(c), the source face and the target face are warped, the size and the location of their bounding squares are different at each face image, but the shape of the corresponding triangles of two faces are the same.

3.2 Face fusion

After warping two faces, we have two faces of the same structures and the same distribution of face features, but the size and the location of their bounding square are different, we extract the bounding square area of the source face, scale and translate this area to the same size and location of the bounding square of the target face, the new source face image is shown in Fig. 2(b).

As we can see from the Fig. 2, the triangles of the source face and the target face are exactly at the same location and of the same shape and size. the purpose of our task is to fuse the source face features into the target face, so we need to know the profile of the source face, then extract the facial part from the source face image. In Fig. 2(c), the yellow outline is generated from the facial landmarks, but we need to shrink the profile a little to make sure that all the pixels are from the face, inside the blue outline is Ω , Ω stands for the facial part we are going to fuse.

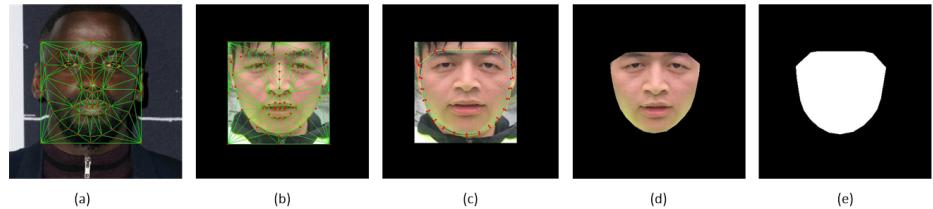


Fig. 2. (a) The target face warped according to F_n . (b) The warped source face after being translated and scaled. (c) Yellow outline is generated from the landmarks of the warped source face, light blue outline is shrank a little from the yellow profile. (d) Facial part of the warped source face that inside Ω . (e) Facial mask.

The target face and the source face are shown in Fig. 2(a) and 2(b), we fuse the gradients of two faces, recover the face from the fused gradients by seamless cloning.

The gradient of the source face is denoted as g_s , the gradient of target face is denoted as g_t , g is the gradient we will use to recover the face, there are three ways we tried to fuse the gradients of two faces.

For the first way, The guidance field g could be generated by linearly combining g_s and g_t :

$$g = \beta \cdot g_s + (1 - \beta) \cdot g_t \quad (3)$$

For the second way, at each point in Ω , retain the stronger gradients in g_s or in g_t :

$$\text{for all } x \in \Omega, g(x) = \begin{cases} g_s(x) & \text{if } |g_s(x)| > |g_t(x)| \\ g_t(x) & \text{else} \end{cases} \quad (4)$$

For the third way, define g by linearly combination of g_d and g_t :

$$g = \beta \cdot g_d + (1 - \beta) \cdot g_t \quad (5)$$

The definition of g_d is:

$$\text{for all } x \in \Omega, g_d(x) = \begin{cases} g_t(x) & \text{if } |g_s(x)| < \theta_1 \text{ or } |g_t(x)| < \theta_2 \\ g_s(x) & \text{else} \end{cases} \quad (6)$$

g_d contains the bigger gradients of the g_s and lower gradients of g_t , if $g_s(x)$ is smaller than the threshold θ_1 and $g_t(x)$ is smaller than the threshold θ_2 , then $g_d(x)$ keeps the gradient from g_t at point x , so it contains main features of the source face and the part of the lightning condition of the target face. Fig. 3(a) and Fig. 3(b) are facial parts of the target face and source face, they share the same profile and distribution of features, Fig. 3(c) shows that which parts of g_d are come from g_t or g_s , for all $x \in \Omega$ and for each channel, if $g_d(x)$ is come from $g_s(x)$, then the corresponding part in Fig. 3(c) is brighter.

Finally, we could use g to recover the face we want on the target face image by seamless cloning, we could control how much features from the source face to fuse into target face by using the first way or the third way.

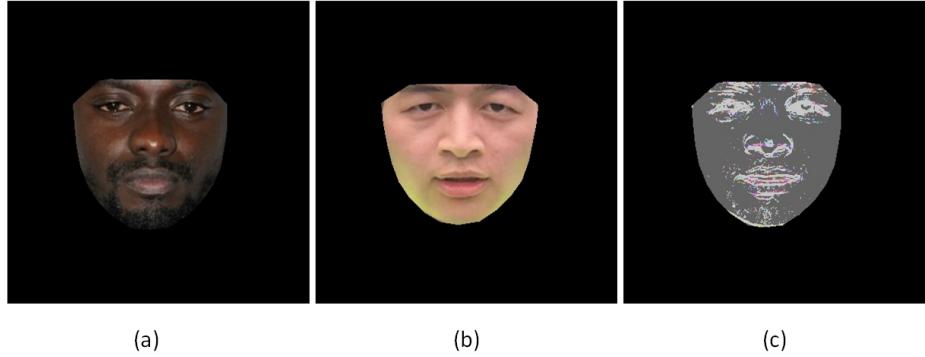


Fig. 3. (a) Facial part of target face. (b) Facial part of the source face. (c) A mask indicate which part of g_d are come from g_s or g_t .

4 Experiments

4.1 Comparison of different gradient reconstruction

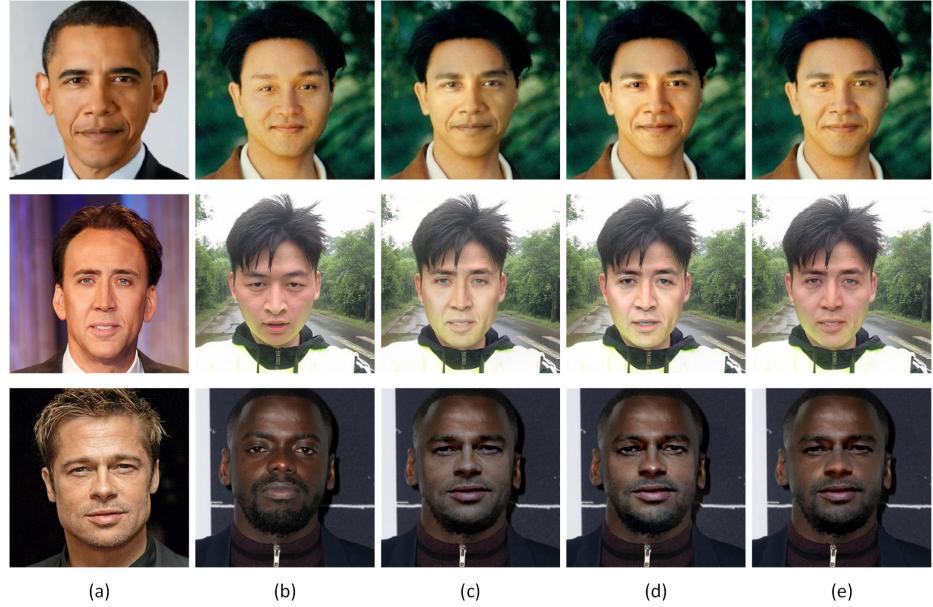


Fig. 4. The effects of different gradient reconstruction: (a) Source face image. (b) Target face image. (c) Fusing face by linearly combining the gradients of two faces. (d) Fusing face by keeping the larger gradients of two faces. (e) Fusing face through weaker gradient preserving method.

Fig. 4 shows the effects of different face gradient reconstruction. From Fig. 4(c), we can see that direct linear combination of two faces gradients could bring too much illumination from source face into target face, which makes face looks very unnatural. If we kept the larger gradients of two faces, the illumination of the target face would be strongly broken, more importantly, we can't control how much features we are going to fuse from source face into target face. In Fig. 4(e), the lightning condition and facial hues is closest to the target face image, when we reconstruct gradients using the third method, we only keep the larger gradients of the source face which means the main features of the source are preserved, and we keep lots of the weaker gradients of the target face, which means we maintain some illumination information and facial hues of the target face.

4.2 How the different parameters effect face fusion

In this paper, there are four parameters we can control, they are α , β , θ_1 , θ_2 . α controls the structure of human faces, in other words, it controls the profile and

the distribution of features of human faces. β controls the details of human faces, like wrinkles and textures. With α and β , we can control how much content we want to fuse from source face into target face. If the gradients of source face is weaker than θ_1 and the gradients of target face is weaker than θ_2 , then we retain the gradients of the target face, otherwise we keep the gradients of the source face.

Fig. 5 shows how α effects the fused results, $\beta = 0.5$, $\theta_1 = 10$, $\theta_2 = 15$. As the α grows bigger, the distribution of face features of fused face is more like the source face.

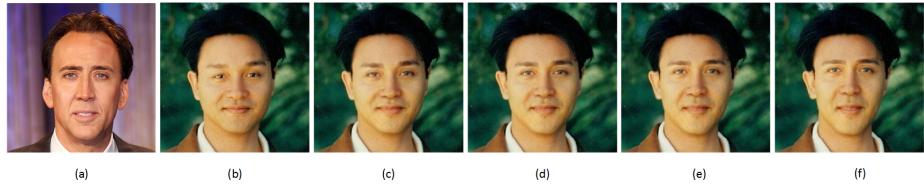


Fig. 5. (a) Source face. (b) Target face. (c) Fused face, $\alpha = 0.2$. (d) Fused face, $\alpha = 0.4$. (e) Fused face, $\alpha = 0.6$. (f) Fused face, $\alpha = 0.8$.

Fig. 6 shows how β effects the fused result, $\alpha = 0.5$, $\theta_1 = 10$, $\theta_2 = 15$. As the β grows bigger, the fused result has more details from source face.

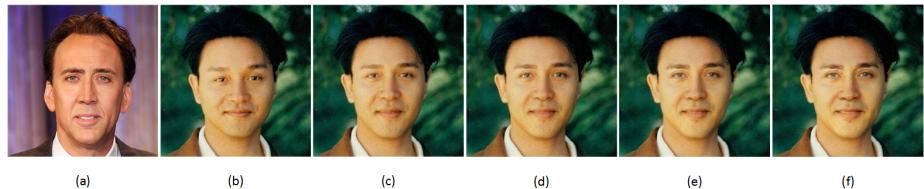


Fig. 6. (a) Source face. (b) Target face. (c) Fused face, $\beta = 0.2$. (d) Fused face, $\beta = 0.4$. (e) Fused face, $\beta = 0.6$. (f) Fused face, $\beta = 0.8$.

If we want to only keep main features of source face, we could only retain stronger gradients of the source face, Fig. 7 shows that the fused results are more natural as more weaker gradients of the source face are been dropped. $\alpha = 0.5$, $\beta = 0.5$, $\theta_2 = 40$. The weaker gradients carry too much illumination information and facial hues of the source face, when these gradients are brought into the fused results, the results are become very unnatural.



Fig. 7. (a) Source face. (b) Target face. (c) Fused face, $\theta_1 = 2$. (d) Fused face, $\theta_1 = 4$. (e) Fused face, $\theta_1 = 6$. (f) Fused face, $\theta_1 = 8$.

When we fuse the main features of source face into target face, somehow, we need to erase the main features of target face. In fact, we want to keep the illumination information of the target face, both the weaker gradients of source face and weaker gradients of target face are being kept. For Fig. 8, it shows the experiment on θ_2 , $\alpha = 0.5$, $\beta = 0.5$, $\theta_1 = 10$, when the θ_2 grows bigger, more stronger gradients are preserved, as we could see, more shadows are preserved.

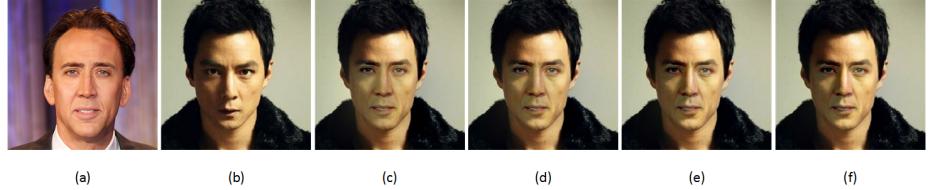


Fig. 8. (a) Source face. (b) Target face. (c) Fused face, $\theta_2 = 10$. (d) Fused face, $\theta_2 = 20$. (e) Fused face, $\theta_2 = 30$. (f) Fused face, $\theta_2 = 40$.

Although our work are not aiming at face swapping task, we still could create similar effects by setting $\beta = 1$, this means we almost preserving all the gradients of source faces, but only keep part of weaker gradients of target faces. We compare our results and the work of Korshunova et al. [16], in Fig. 9, the source face of all the results is from Fig. 8(a), we can see that our results keep more illumination information and facial hues of target faces which makes our results more natural, but in Fig. 9(c), our result fail to catch the lightning condition of the target face from a high level, this is where we need improve.

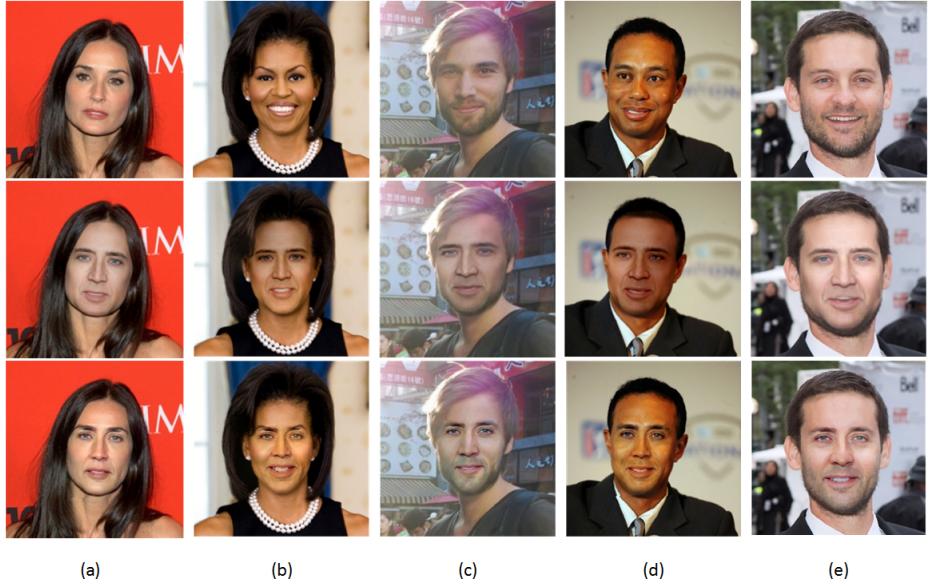


Fig. 9. The first row are the original target face, the second row are the results of Korshunova et al., the third row are our results.

5 Conclusions

We developed a face fusion method that could automatically fusing one face features into the other face. When we are considering fusing human faces, there are two parts we are going to fuse, one is face structures, the other is face details. The 68 facial landmarks are enough to align two faces precisely. The illumination of the source face might pollute the target face, but the method we proposed preserves as much illumination information as possible while decreasing the affection of illumination of the source face. At mean time, we developed a user interface for letting users control how much features they want from the source face to fuse into the target face.

References

1. Beier, T., Neely, S.: Feature-based image metamorphosis. In: ACM SIGGRAPH Computer Graphics, Vol. 26, No. 2, pp. 35-42 (1992)
2. Wolberg, G.: Image morphing: a survey. In: The visual computer, 14(8), 360-372 (1998)
3. Prez, P., Gangnet, M., Blake, A.: Poisson image editing. In: ACM Transactions on Graphics (TOG), Vol. 22, No. 3, pp. 313-318 (2003)
4. Karungaru, S., Fukumi, M., Akamatsu, N.: Morphing Human Faces: Automatic Control Points Selection And Color Transition. In: International Conference on Computational Intelligence, pp. 224-227 (2004)
5. Blanz, V., Scherbaum, K., Vetter, T., Seidel, H. P.: Exchanging faces in images. In: Computer Graphics Forum, Vol. 23, No. 3, pp. 669-676 (2004)
6. Bitouk, D., Kumar, N., Dhillon, S., Belhumeur, P., Nayar, S. K.: Face swapping: automatically replacing faces in photographs. ACM Transactions on Graphics (TOG), 27(3), 39. (2008)
7. De La Hunty, M., Asthana, A., Goecke, R.: Linear facial expression transfer with active appearance models. In: International Conference on Pattern Recognition (ICPR) (2010)
8. Ross, A., Othman, A.: Visual cryptography for biometric privacy. In: IEEE transactions on information forensics and security.(2011)
9. Lin, Y., Wang, S., Lin, Q., Tang, F.: Face swapping under large pose variations: A 3D model based approach. In: International Conference on Multimedia and Expo (ICME), pp. 333-338 (2012)
10. Lin, Y., Lin, Q., Tang, F., Wang, S.: Face replacement with large-pose differences. In: Proceedings of the 20th ACM international conference on Multimedia. (2012)
11. Xiong, X., De la Torre, F.: Supervised descent method and its applications to face alignment. In: Proceedings of the IEEE conference on computer vision and pattern recognition.(2013)
12. Ren, S., Cao, X., Wei, Y., Sun, J.: Face alignment at 3000 fps via regressing local binary features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.(2014).
13. Cao, X., Wei, Y., Wen, F., Sun, J.: Face alignment by explicit shape regression. In: International Journal of Computer Vision. (2014)
14. Mosaddegh, S., Simon, L., Jurie, F.: Photorealistic Face de-Identification by Aggregating Donors Face Components. In: Asian Conference on Computer Vision, pp. 159-174 (2014)

15. Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), pp. 1867-1874 (2014)
16. Korshunova, I., Shi, W., Dambre, J., Theis, L.: Fast face-swap using convolutional neural networks. arXiv preprint arXiv:1611.09577. (2016)
17. Nirkin, Y., Masi, I., Tran, A. T., Hassner, T., Medioni, G.: On Face Segmentation, Face Swapping, and Face Perception. arXiv preprint arXiv:1704.06729. (2017)