



# Facial expression synthesis with direction field preservation based mesh deformation and lighting fitting based wrinkle mapping

Weicheng Xie<sup>1</sup> · Linlin Shen<sup>1</sup> · Meng Yang<sup>1</sup> ·  
Jianmin Jiang<sup>2</sup>

Received: 3 October 2016 / Revised: 23 February 2017 / Accepted: 29 March 2017 /

Published online: 8 April 2017

© Springer Science+Business Media New York 2017

**Abstract** Nowadays, facial expression synthesis is widely used in expression simulation, recognition and animation. While a variety of works available in literature are 3D based, these algorithms usually require face matching and trimming, are thus time consuming. In this work, an automatic algorithm for expression synthesis in 2D space is proposed, which mainly consists of three stages. The optimum matching of three sets of feature points on the faces of source neutral ( $F_{sn}$ ), source expression ( $F_{se}$ ) and target neutral ( $F_{tn}$ ) are obtained in the first stage. Different components on the target face are deformed by learning from not only the displacements but also the geometry shape of face  $F_{se}$  in the second stage. In the last stage, the details of the source expression are mapped onto the corresponding positions on the target face by fitting of the lighting differences. Experimental results on expression synthesis with large geometry deformation and lighting difference show that the proposed algorithm is able to accurately preserve the geometry deformation, and the synthesized expressions are visually realistic.

**Keywords** Expression synthesis · Face correspondence · Mesh deformation · Wrinkle fitting

---

✉ Linlin Shen  
llshen@szu.edu.cn

Jianmin Jiang  
jianmin.jiang@szu.edu.cn

<sup>1</sup> Computer Vision Institute, School of Computer Science & Software Engineering, Shenzhen University, Shenzhen, China

<sup>2</sup> Research Institute for Future Media Computing, School of Computer Science & Software Engineering, Shenzhen University, Shenzhen, China

## 1 Introduction

With the development of face recognition and facial animation, expression synthesis has been increasingly important and widely used in these applications. Low-quality 2D images and 3D depth maps were mapped to realistic facial expressions by optimization with geometry and texture registration [29]. Features of 2D geometric deformation were employed for the facial expression recognition [10], where 2D deformable models tracking the greatest facial expression intensity were used for SVM classification. Other algorithms related with facial deformation and synthesis for expression recognition were also reported in [1, 19, 23].

Generally, the algorithms of expression synthesis are mainly divided into the categories of 3D and 2D based approaches. The 3D based algorithms can synthesize realistic expressions [32], however, they often require a matching between 2D image and 3D shape or a database of basic expression set with respect to (w.r.t.) the target person when a 2D non-expression face is provided. The 2D based algorithms learn geometry and texture features of the reference (source) expressions and map these features directly onto the target face. Based on the number of the reference expressions, the 2D based algorithms can be classified into two classes, i.e. statistical learning of multiple reference expressions, and face deformation and wrinkle mapping with a single expression face.

With respect to statistically-based algorithms, [4] divided the expression synthesis into the phases of analysis network training and synthesis network learning. A database of various facial expressions of a single person was learned by the principle component analysis (PCA) for simulating new expressions [33]. This method may not work when the size of the face varies largely as it was based on facial region segmentation and interpolation. Using bilinear kernel regression [8], facial expression synthesis was proposed to preserve subtle person-specific facial characteristics by learning expression variations. Facial animation parameters (FAPs) were used to model the primary expressions in [21]. Vlasic et al. [28] proposed to synthesize a new expression face from the Cartesian product of sub-statistical models of the example's identity, expression and viseme. For statistically-based algorithms, a database of referred expressions w.r.t. the considered person is needed in general, and these methods mainly concentrate on the detail preservation of the generated expression.

The deformation and mapping based algorithms generally require the following steps:

- Location and alignment of the feature points on faces of  $F_{sn}$ ,  $F_{se}$  and  $F_{tn}$ ;
- Deformation of the target face  $F_{tn}$  by learning the movement from  $F_{sn}$  and  $F_{se}$ ;
- Wrinkle mapping on the deformed face of  $F_{tn}$ .

### 1.1 Correspondence matching

For face correspondence matching, Liu et al. [15] and Zhang et al. [34] located the positions and determined the correspondence of the feature points manually. Song et al. [25] made this step semi-automatic by determining the initial positions of the feature points with active appearance model, then these locations are adjusted to be inside of the face contour by a few manual interactions. Actually, this step can be made automatic using state-of-the-art face alignment algorithms. Li et al. [13] proposed to align the faces by the optical flow. Qian et al. [20] and Liao et al. [14] found a better contour matching between two faces by the minimization of an energy function. Cascade of regression trees were adopted in the face alignment in [9, 22]. These algorithms obtain the facial correspondence by a global optimization. While the local features around facial parts were not matched, the genuineness of synthesized expression is greatly compromised. The employed optimization in this paper

is concentrated on matching local features around facial parts by optimizing the feature point location incorporating with the detected boundary pixels.

## 1.2 Face deformation

For mesh deformation, Sumner and Popovic [26] presented a method of deformation transfer by deforming a source triangle mesh with a minimization optimization. However, the feature points were manually located, and the details were not well preserved. The facial expression was generated by simulating the muscle deformation with differential equation model in 3D space [12]. A hierarchical geodesic-based resampling approach was applied to extract the landmarks for deformation modeling [16].

For 2D deformation algorithms, [8, 33] firstly located the target feature points from the expression database of the considered person, then the non-feature points were interpolated with basis functions. However, these interpolation algorithms neglect the geometry features of the target face. The wrapping algorithm in [3] obtained the movement of each pixel by incorporating a few directed line segments and the distances between the pixel and these key segments on  $F_{te}$ , which was also adopted in [15, 25] for mesh deformation. However, this deformation algorithm was time-consuming, and the line segments on  $F_{te}$  were difficult to be determined when the face parts are largely deformed. The feature points on the deformed target face were approximated with the weighted positions of the feature points on the reference face by an elastic model [34]. While the geometry features of the learned face were preserved, the facial part smoothness are not well preserved during the deformation. The work in [18] compared the performance of 2D (e.g. Active Appearance Models) and 3D (e.g. Morphable Models) face models in terms of representation power, construction accuracy and efficiency. A global optimization model for 2D shape deformation was presented in [30], while the geometry features inside the object such as the directional vectors of the part edge were ignored.

## 1.3 Wrinkle mapping

While [2, 11] model wrinkles for the 3D expression synthesis, the 2D wrinkle simulation needs only to learn and map the reference wrinkle features. [15] assumed the expression ratio image (ERI) between  $F_{sn}$  and  $F_{se}$  to be the same as that between  $F_{tn}$  and  $F_{te}$ . The expression features with ERI were mapped to the retrieved and deformed face sequence to obtain the final expression sequence [13]. Shift ERI incorporated with Gaussian damping weighting term was employed to map the wrinkles onto the target face [34]. These ratio based algorithms of lighting mapping are largely influenced by the abnormal lighting ratios when the lighting values of pixels on  $F_{sn}$  is relatively small. To avoid this disadvantage, we adopt the difference of the lighting intensities [6] rather than the ratio between corresponding pixels on  $F_{sn}$  and  $F_{se}$  to simulate the lighting changes. Moreover, a fitting form of the lighting difference [31] is proposed.

Thus, for deformation and mapping based algorithms, three problems still need to be resolved. Firstly, the alignment of the feature points needs to be improved for accurate part matching. Secondly, the direction information of the part edges is not sufficiently learned in the face deformation. Lastly, the existing wrinkle mapping algorithms are mostly based on the ratio of lighting intensity, which is sensitive to the abnormal lighting. The paper aims to propose a new algorithm of expression synthesis to deal with these three problems.

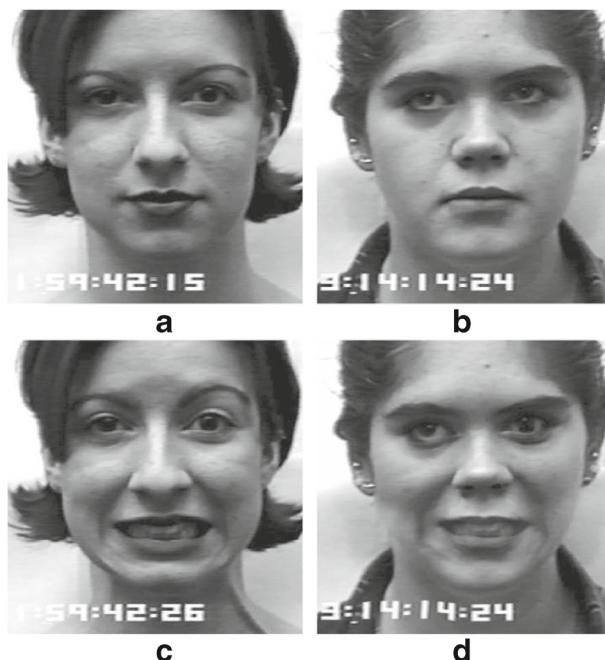
In this work, a mesh deformation algorithm incorporating wrinkle mapping (abbreviated as DaWF) is proposed to synthesize the facial expressions. The main contributions of the

proposed DaWF concentrate on three aspects. Firstly, we locally optimize the correspondences among the feature points on the faces  $F_{sn}$ ,  $F_{se}$  and  $F_{tn}$  by minimizing an energy function, which matches part boundary and the differences between the local pixels around the face part edges. In this way, the lighting intensities on two faces are better matched to reduce abnormal lighting differences. Secondly, accurate geometry deformation of the parts is obtained by a mesh deformation algorithm, which can properly preserve the direction field of part edges. Experimental results of the two deformation algorithms in Section 3.1 demonstrate that DaWF can efficiently simulate the geometry shape changes of different expressions. Thirdly, the wrinkle mapping on the deformed face is performed by solving a system of linear equations to fit lighting differences. Comparative results with several related algorithms in Section 3.2 reveal that the proposed mapping method can generate more complete wrinkle and is both robust and efficient.

This paper is structured as follows. Section 2 gives a description about the proposed algorithm step by step. The experimental results of the proposed algorithm on various expressions are presented in Section 3. Finally, discussions and some conclusions are addressed in Section 4.

## 2 The proposed algorithm

The sketch of expression synthesis is illustrated in Fig. 1, where the faces from Fig. 1a to b are used for learning, the learned information is then applied to Fig. 1c and finally the synthesized expression is showed in Fig. 1d.



**Fig. 1** The sketch of the expression synthesis. **a** Source neutral face  $F_{sn}$ , **b** Source fear expression for learning  $F_{se}$ , **c** Target neutral face  $F_{tn}$ , **d** Target fear expression after learning  $F_{te}$

## 2.1 Framework of the algorithm

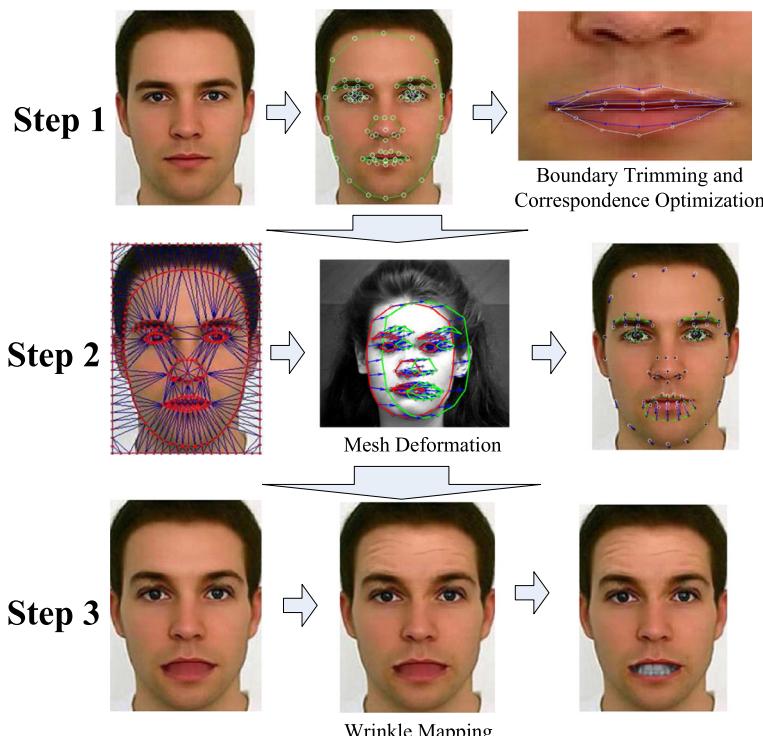
As presented in Fig. 2, the general framework of the proposed DaWF consists of three steps. In the first step, the correspondence between feature points is matched. In the second step, we obtain the deformed geometry by mesh deformation. The wrinkles and the teeth are mapped onto the target face in the last step.

It can be seen from Fig. 2 that the proposed DaWF mainly differs from the other algorithms on three aspects. Firstly, the feature points and the edges of the parts are aligned by a local optimization. Secondly, direction field is incorporated into the mesh deformation to preserve the directional features of the target expression during geometry deformation. Lastly, lighting differences among nine neighboring pixels and fitting of the differences are introduced for the wrinkle mapping.

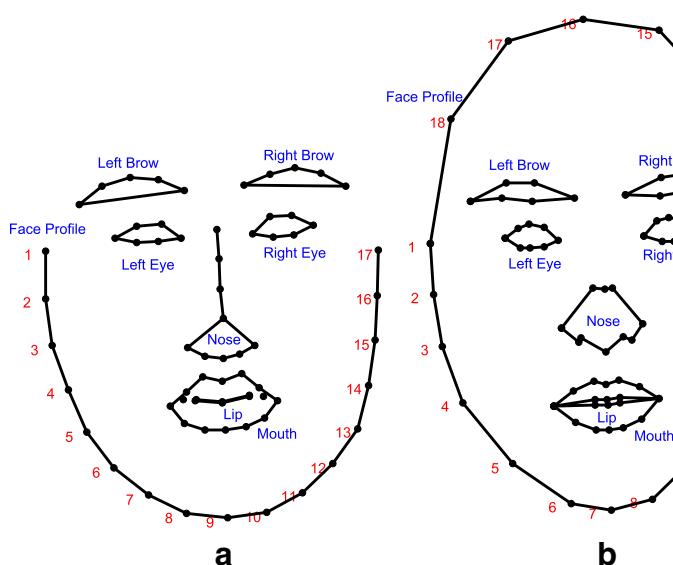
As shown in Fig. 3b, we view the whole face as a structure  $\Gamma = \{\Gamma_1, \dots, \Gamma_n\}$  which consists of eight parts i.e. two eyes, two brows, nose, mouse, lip and profile. Each part encloses a closed region represented with shapes, geometry features, lighting intensities and textures like wrinkles. The proposed algorithm aims to reconstruct the texture and geometry properties of all the considered parts when the expression is synthesized.

## 2.2 Correspondence matching based on local optimization

Before correspondence matching between two sets of feature points, active appearance model (AAM) [27] is first employed to obtain the initial positions of 68 feature points,



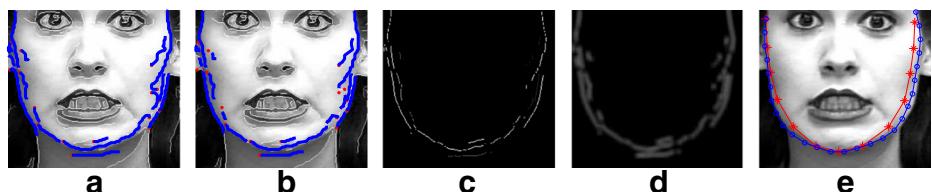
**Fig. 2** The framework of the proposed algorithm



**Fig. 3** Facial parts, profile and the locations of 68 and 79 feature points

which are expanded to 79 points to include the forehead region with Procrustes transformation. Figure 3a, b show the locations of the 68 and 79 feature points, respectively.

In order to map the expression feature onto the accurate position of the target face, edges of the parts on the considered faces are finely matched and adjusted by local optimization. That is, based on the obtained 79 feature points, each part boundary is interpolated with a B-spline curve. Then the positions of the feature points on the interpolated B-spline curve are locally adjusted by considering the lighting intensities of local neighboring pixels on  $F_{sn}$ ,  $F_{se}$  and  $F_{tn}$ , where the lighting intensity is defined as the Y component of the YUV color [15]. The correspondence matching is obtained by integrating the part boundary trimming with correspondence optimization, which is divided into the following five steps. Figure 4 presents the optimization procedure of correspondence between  $F_{sn}$  and  $F_{se}$  with the face profile as an example.



**Fig. 4** Procedure of correspondence optimization of a face profile. **a** demonstrates the continuous edge segments detected by Canny operator in a local region, where edge segments are labeled with blue lines and the starting points are labeled with red dots. **b** records the trimmed boundary pixels. Images of boundary pixels with different degrees and its smoothness are presented in (c) and (d), respectively. **e** presents the initial and optimized edges labeled with red stars and blue circles, respectively

- Canny edge detector is employed on faces  $F_{sn}$  and  $F_{se}$  to obtain the boundary pixels, and continuous boundary edge segments are found by depth first searching (DFS) in a local region around the initial face profile;
- Boundary pixels on edge with length less than 10 or angle larger than  $\pi/3$  are abandoned;
- After trimming the boundary pixels, the strength of each edge pixel is computed by gradually increasing the threshold of Canny operator;
- Each edge pixel is assigned an index to indicate its strength of being boundary and smoothed to facilitate the local searching around the face profile;
- Optimized face profile is obtained by a local optimization formulated in (1).

The first term of optimization (1) denotes the lighting intensities of the smoothed face  $F_{se}$ , which records the distance between the re-located pixel and the strongest edge pixels detected by Canny operator. The second term records the similarity of direction field with the original edge. The third term corresponds to the smoothness of the interpolated boundary edge, which is used to reduce abnormal feature points. The last term reflects the matching degree between the source and target face edges, which is introduced to finely locate the correspondence points on the target face. The matching degree  $M_i$  of the  $i$ -th point is formulated in (2).

$$\begin{aligned} \min E = & -\frac{1}{n} \sum_{i=1}^n l_{P_i}^{(se)} + \frac{\alpha}{n-1} \sum_{i=1}^{n-1} \|v_i - i v_i\|^2 \\ & + \frac{\beta}{n-2} \sum_{i=2}^{n-1} \left\| \frac{v_{i+1} + v_{i-1}}{2} - v_i \right\|^2 + \frac{\gamma}{n} \sum_{i=1}^n M_i. \end{aligned} \quad (1)$$

where  $\{IP_i, i = 1, \dots, n\}$ ,  $\{P_i, i = 1, \dots, n\}$  are the discrete approximations of the initial and current interpolation B-spline curves, respectively.  $v_i = P_{i+1} - P_i$ ,  $iv_i = IP_{i+1} - IP_i$ .

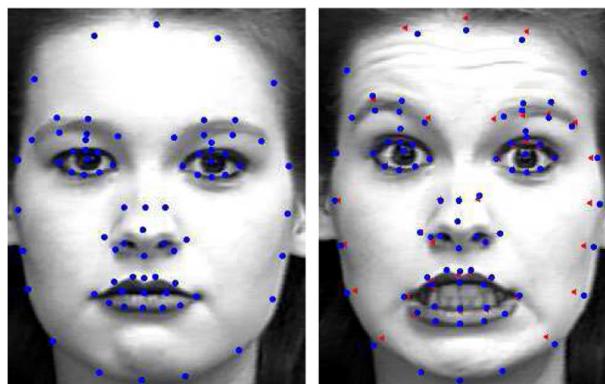
$$M_i = \sum_{1 \leq j \leq w, 1 \leq k \leq h} \left| l_{i,j,k}^{(sn)} - r_i l_{i,j,k}^{(se)} \right|. \quad (2)$$

where  $\{l_{i,j,k}^{(sn)}, j = 1, \dots, w; k = 1, \dots, h\}$  and  $\{l_{i,j,k}^{(se)}\}$  denote the lighting intensities of the local pixels around the  $i$ -th points on  $F_{sn}$  and  $F_{se}$ , respectively.  $w = 14$  and  $h = 7$  denote the numbers of pixels in the tangent direction  $T_d$  and normal direction  $N_d$ .  $r_i = \left| \frac{l_{(i,j,k)+t \cdot N_d}^{(sn)} - l_{(i,j,k)-t \cdot N_d}^{(sn)}}{l_{(i,j,k)+t \cdot N_d}^{(se)} - l_{(i,j,k)-t \cdot N_d}^{(se)}} \right|$  is the ratio of the lighting differences of two pixels in the normal direction  $N_d$ , which is introduced to normalize the lighting intensities of faces  $F_{sn}$  and  $F_{se}$  around the boundary regions.

The minimization problem (1) is optimized iteratively by Golden section method with the parameters  $(\alpha, \beta, \gamma)$  increasing from  $(0.005, 0.005, 0.001)$  to  $(0.2, 0.1, 0.2)$  and the number of discrete points  $n$  increasing from  $n_0$  to  $3 \cdot n_0$ , where  $n_0$  is the number of initial feature points on the part boundary. Similar optimization is also applied to  $F_{tn}$  to adjust the feature points located by AAM. The fourth term of optimization (1) is omitted for the part edge optimization of the reference face  $F_{sn}$ .

Figure 5 presents the initial and adjusted feature points on the face  $F_{se}$ . One can observe that most of the feature points on the face profile of  $F_{se}$  have been adjusted to match the corresponding points on  $F_{sn}$ .

The optimization model (1) is newly proposed for the correspondence matching of the feature points on the faces  $F_{sn}$ ,  $F_{se}$  and  $F_{tn}$ . Different from current algorithms which manually refine the feature points based on coarse point correspondence, the proposed algorithm



**Fig. 5** The left image depicts the feature points on  $F_{sn}$  located by AAM and optimization (1), the right image demonstrates the feature points on  $F_{se}$ , where the initial feature points are labeled with red triangles and the adjusted feature points are labeled with blue dots

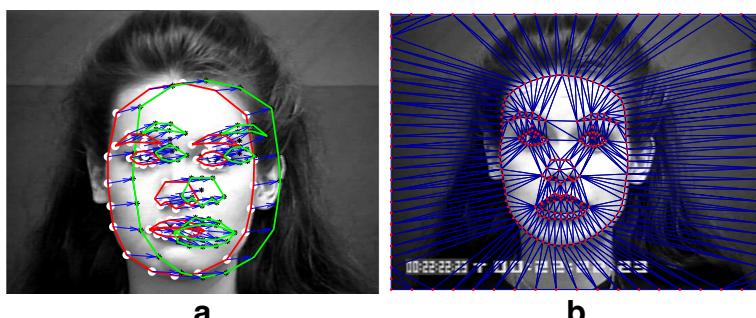
automatically adjusts the feature points to match the local texture features of the reference points and the part boundary edges while preserving the geometry property of these points.

### 2.3 Mesh deformation based on edge direction preservation

The procedure of generating the deformed feature points on  $F_{te}$  is divided into three steps.

In the first step, all the feature points are rotated to an unified coordinate system with reference to the positions of the two eyes and the nose on  $F_{sn}$  and  $F_{tn}$ . In the second step, initial positions of feature points on  $F_{te}$  are obtained according to the scaled movements of feature points on  $F_{sn}$  and  $F_{se}$ , as demonstrated in Fig. 6a.

In the third step, accurate feature points and the corresponding deformed non-feature points on  $F_{te}$  are obtained using a global triangulation mesh deformation based on the feature points of  $F_{te}$ . The mesh employed in this work is generated by a triangulation with boundary constraints of the considered parts, where the boundary points are uniformly and densely sampled from the interpolated B-spline curve. Figure 6b shows the triangulation of an example face.



**Fig. 6** The movements and triangulation of feature points. **a:** The movements of all the feature points from  $F_{sn}$  to  $F_{se}$ . The points in the white circles are from  $F_{sn}$ . **b:** The constrained triangulation of an example set of feature points

Motivated by the work in [30], we propose to deform the mesh by the minimization optimization using (3). The optimization approach in [30] can flexibly deform source exterior boundary into the target shape, while the deformation of the interior parts and the fine geometry feature such as edge direction field are not considered. Thus, for expression synthesis application, we proposed an improved algorithm of [30] as below:

$$\begin{aligned} \min & ||MV||_2^2 + \lambda_1 ||HV - e(V)||_2^2 + \lambda_2 ||DV - \delta(V)||_2^2 + \lambda_3 ||CV - U||_2^2 \\ \text{s.t. } & \Lambda(V_i) - \tilde{\Lambda}_i = 0, 1 \leq i \leq N. \end{aligned} \quad (3)$$

where  $V$  records the vector form of vertex coordinates of the mesh,  $D, M, H, C$  correspond to the coefficient matrixes,  $\delta(V), e(V), \tilde{\Lambda}_k$  are the corresponding terms for learning and  $N = 8$  is the number of considered parts on the face.

In the first term of the optimization, the matrix  $M$  describes the weight of each interior point and the corresponding adjacent points. This term is used to restrict the bias between a point and the corresponding mean coordinate, which tries to make the interior point be close to the weighted center of the adjacent points and formulated as below:

$$v_i - \sum_{(i,j) \in E} w_{i,j} \cdot v_j = 0, \text{ for } v_i \in V_g. \quad (4)$$

where  $v_i$  is the  $i$ -th point of the vector representation of mesh points  $V$ ,  $E$  denotes the set of all edges,  $V_g$  records all the interior points.

In the second term, the matrix  $H$  records the constraints of the lengths of the edges, i.e. the lengths of all the edges keep constant before and after deformation, which is formulated as below:

$$\sum_{(i,j) \in E_g} \left\| (v_i - v_j) - e(v_i^{(k)}, v_j^{(k)}) \right\|^2, \text{ with } e(v_i^{(k)}, v_j^{(k)}) = \frac{l_{i,j}^{(0)}}{l_{i,j}^{(k)}} (v_i^{(k)} - v_j^{(k)}). \quad (5)$$

where  $E_g$  records all the interior edges,  $l_{i,j}^{(k)}$ ,  $v_i^{(k)}$  denote edge length and point of current iteration.

The preceding two terms are introduced to regulate the deformation of points and shape in the work [30], while the last two terms are newly proposed for the specific expression synthesis model.

The third term is newly introduced, where the matrix  $D$  corresponds to the coefficient matrix of  $\{V_{i,j}^{(k+1)}\}$  in the (6), and  $\delta(V)$  corresponds to the right side vector of this term.

$$\begin{aligned} & \left\| V_{i,j+1}^{(k+1)} - V_{i,j}^{(k+1)} - \delta(V_{i,j}^{(se)}, V_{i,j}^{(k)}) \right\|^2 \\ \text{with } & \delta(V_{i,j}^{(se)}, V_{i,j}^{(k)}) = \left\| V_{i,j+1}^{(k)} - V_{i,j}^{(k)} \right\| \cdot r_{i,j}^{se} \cdot \frac{V_{i,j+1}^{(se)} - V_{i,j}^{(se)}}{\left\| V_{i,j+1}^{(se)} - V_{i,j}^{(se)} \right\|} \\ \text{for } & \{V_{i,j}, V_{i,j+1}\} \subset E_p. \end{aligned} \quad (6)$$

where  $V_{i,j}^{(k)}$  is the  $j$ -th point on the  $i$ -th part of the face  $F_{te}$  at the  $k$ -th iteration,  $V_{i,j+1}^{(se)} - V_{i,j}^{(se)}$  is the directional vector on  $i$ -th part of the face  $F_{se}$ ,  $r_i^{se}$  is the ratio between the width and height of the  $i$ -th part on  $F_{se}$ ,  $E_p$  records all the boundary points. This term is proposed to preserve the proportional direction vectors (field) on the part boundary of  $F_{se}$  and the corresponding length proportions, i.e., the deformation algorithm not only learns from the displacements of the feature points, but also from the direction field of geometry shape  $F_{se}$ . To prevent the deformed points being far from the initial feature points, the fourth term is introduced.

In the fourth term, matrixes  $C, U$  are set for the position constraints. In this work, this term reflects the geometry changes of the feature point positions. We try to restrict the feature points within a close distance from the obtained initial positions by (7):

$$V_{i,j} = \tilde{V}_{i,j}, \text{ for } 1 \leq i \leq N, 1 \leq j \leq i_n. \quad (7)$$

where  $\tilde{V}_{i,j}$  denotes the initial position of the feature point  $V_{i,j}$ .

The constraints in the optimization problem reflect the restrictions of the areas before and after deformation. In (8), operator  $\Lambda(V_i)$  denotes the area of the region enclosed by the  $i$ -th part, as formulated below:

$$\begin{cases} \Lambda(V_i) = \frac{1}{2} \sum_{j=1}^{i_n} (x_{i,j} y_{i,j_n} - x_{i,j_n} y_{i,j}) \\ j_n = j, 1 \leq j \leq i_n; j_n = 1, j = i_n. \end{cases} \quad (8)$$

The parameters  $\{\lambda_i, 1 \leq i \leq 3\}$  control the importance of each term to the global optimization (3). In this work, we set the values as  $\lambda_1 = \lambda_3 = 1, \lambda_2 = 2$  for all the test problems after a few trials. The coefficient  $\lambda_2$  can be set to be larger when a closer matching with the learned geometry shape is needed. This least square optimization is solved using iterative Gauss-Newton method [30].

After obtaining the feature points on the face  $F_{te}$ , these points in the unified framework (i.e. relative coordinate) are transformed back into the coordinate system of  $F_{tn}$  (i.e. absolute coordinate). With an inverse mapping, the pixel colors in the deformed triangle on  $F_{te}$  are approximated with the corresponding pixel colors in the initial triangle on  $F_{tn}$ . The color at each point is computed by bilinear interpolation using the colors of the neighboring pixels.

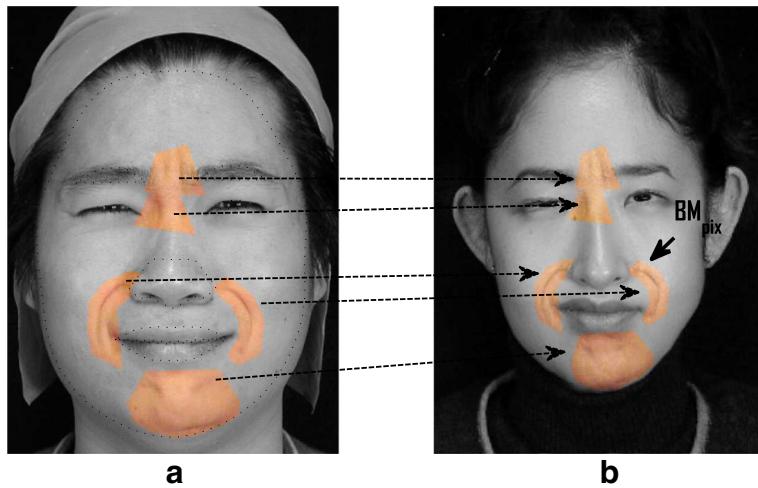
Compared to the optimization proposed in [30], our optimization has three new ideas. Firstly, not only the geometry shape of the exterior face profile but also those of the interior parts are considered in the objective function. Secondly, the curve Laplacian term in the original optimization introduced to preserve the degree of curve blending during deformation is removed, while the directional vectors along the edges of parts are introduced to preserve the detailed geometry features of the parts. Lastly, multiple boundary edges rather than one edge of the considered object are considered for the overall deformation.

## 2.4 Wrinkle mapping based on lighting difference fitting

After the part deformation, the texture details (such as the wrinkle) are mapped onto the target face subsequently to transfer the fine expression features. In the work [15], the wrinkle was reflected as the lighting variants at the pixel (the Y component of the YUV colors). They further proposed to simulate the lighting intensity of the pixel on the face  $F_{te}$  based on the ratio of the lighting intensities, and learn the lighting change from the corresponding pixel on  $F_{se}$ .

Different from lighting ratio-based mapping in [15], a difference based lighting mapping is introduced by solving a linear equation. In the proposed algorithm, each pixel learns from the lighting differences of the adjacent eight pixels, which makes the algorithm robust for abnormal lighting ratios when the pixel lighting of  $F_{sn}$  is relatively small.

First, we denote the regions on the face  $F_{te}$  where the wrinkles need to be manually extracted and mapped as  $BM_{pix}$ . Figure 7 demonstrates the extraction and mapping of wrinkle regions. The corresponding lighting intensities of faces  $F_{sn}, F_{se}, F_{tn}$  in the  $3 \times 3$  adjacent region around the position  $(p, q)$  are denoted as  $\{l_{p,q}^{(sn)}, l_{p,q}^{(se)}, l_{p,q}^{(tn)}; i-1 \leq p \leq i+1, j-1 \leq q \leq j+1\}$ . Then the corresponding lighting intensities on the face  $F_{te}$  are assumed to satisfy the constraints in (9)–(10).



**Fig. 7** The extraction and correspondence of wrinkle regions. **a:** The extraction of wrinkle regions on  $F_{se}$  with colorized label. **b:** The corresponding wrinkle regions  $BM_{pix}$  on  $F_{te}$

For  $(i, j) \in BM_{pix}$

$$\begin{cases} \left( l_{p,q}^{(te)} - l_{i,j}^{(te)} \right) = \left( l_{p,q}^{(tn)} - l_{i,j}^{(tn)} \right) + \lambda_r \cdot \left[ \left( l_{p,q}^{(se)} - l_{i,j}^{(se)} \right) - \left( l_{p,q}^{(sn)} - l_{i,j}^{(sn)} \right) \right], \\ i-1 \leq p \leq i+1, j-1 \leq q \leq j+1. \end{cases} \quad (9)$$

where  $\lambda_r = \frac{CE_{t,r}}{CE_{s,r}}$ ,  $CE_{s,r}$ ,  $CE_{t,r}$  are the changed extents of lighting intensities of the  $r$ -th wrinkle region on  $F_{sn}$ ,  $F_{tn}$ , respectively, which are approximated by the difference of the 95% largest and 5% largest lighting intensities of each region  $BM_{pix}$  to alleviate the influence of abnormal lighting intensities.

For  $(i, j) \notin BM_{pix}$

$$l_{i,j}^{(te)} = l_{i,j}^{(tn)}. \quad (10)$$

The (9) incorporating with (10) are then converted into one whole system of linear equations:

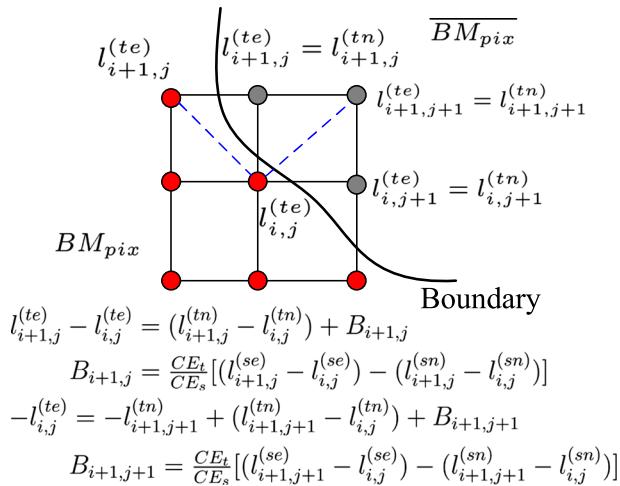
$$\begin{cases} A \cdot L|_{BM_{pix}} = B \\ L|_{\overline{BM_{pix}}} = L_0 \end{cases} \quad (11)$$

where  $L = \left\{ l_{i,j}^{(te)}, (i, j) \in BM_{pix} \right\}$ ,  $A$ ,  $B$  correspond to the coefficient matrixes constructed by (9),  $\overline{BM_{pix}} = \Omega - BM_{pix}$  and  $\Omega$  is the entire facial region,  $L_0$  records the corresponding lighting intensities on the face  $F_{tn}$ . The least squares of this system  $(A^T \cdot A) \cdot L = A^T \cdot B$  is then solved by the sparse LU decomposition. The construction of the equations in the linear system around a boundary pixel is illustrated in Fig. 8.

Equation (11) is not well formulated when the lighting differences between  $F_{se}$  and  $F_{sn}$  along edges of wrinkle regions are significant, which makes the mapped wrinkles look not genuine. Thus, attenuation weight coefficient  $\lambda_r(d)$  is introduced to substitute  $\lambda_r$  in (9)

$$\lambda_r(d) = \begin{cases} \lambda_r \cdot \sin \left( \frac{\pi}{2} \cdot \frac{d}{d_0} \right), & d \leq d_0 \\ \lambda_r, & d > d_0 \end{cases} \quad (12)$$

where  $d$  is the distance between the considered pixel in the wrinkle region and the nearest edge,  $d_0 = 5$  is a critical distance,  $\lambda_r$  is the  $r$ -th lighting ratio which is defined in (9).



**Fig. 8** The linear equations for an example boundary pixel  $(i, j)$

After the details accompanied with the expression are mapped onto the target face, the teeth texture needs to be padded for the opened mouth. However, the teeth region may be invisible, which varies from person to person and from expression to expression. [5, 33] generated new transformed teeth texture with several fixed teeth templates for different kinds of open mouth regions. [13] defined a similarity metric of optimal flow field to match the most similar expression including the teeth region for the expression retargeting. In this work, the teeth in the mouth region of  $F_{se}$  are directly extracted and embedded into the corresponding region on  $F_{te}$ . The procedures of teeth extracting, embedding and blending are described in the following:

- Convert the RGB colors of  $F_{se}$  and  $F_{te}$  to YUV representation;
- Extract the teeth region from  $F_{se}$  by shrinking the outer mouth boundary and scale its size to corresponding size of the face  $F_{te}$  by nearest interpolation;
- Perform color blending around the boundary of the lip region by linear interpolation as (13);
- Convert the blended YUV colors of  $F_{te}$  back to those in the format of RGB and output the image.

$$BlenY_{i,j} = \frac{k}{Len+1} InitY_{i,j-1} + \frac{Len+1-k}{Len+1} InitY_{i,j+1}, \quad (13)$$

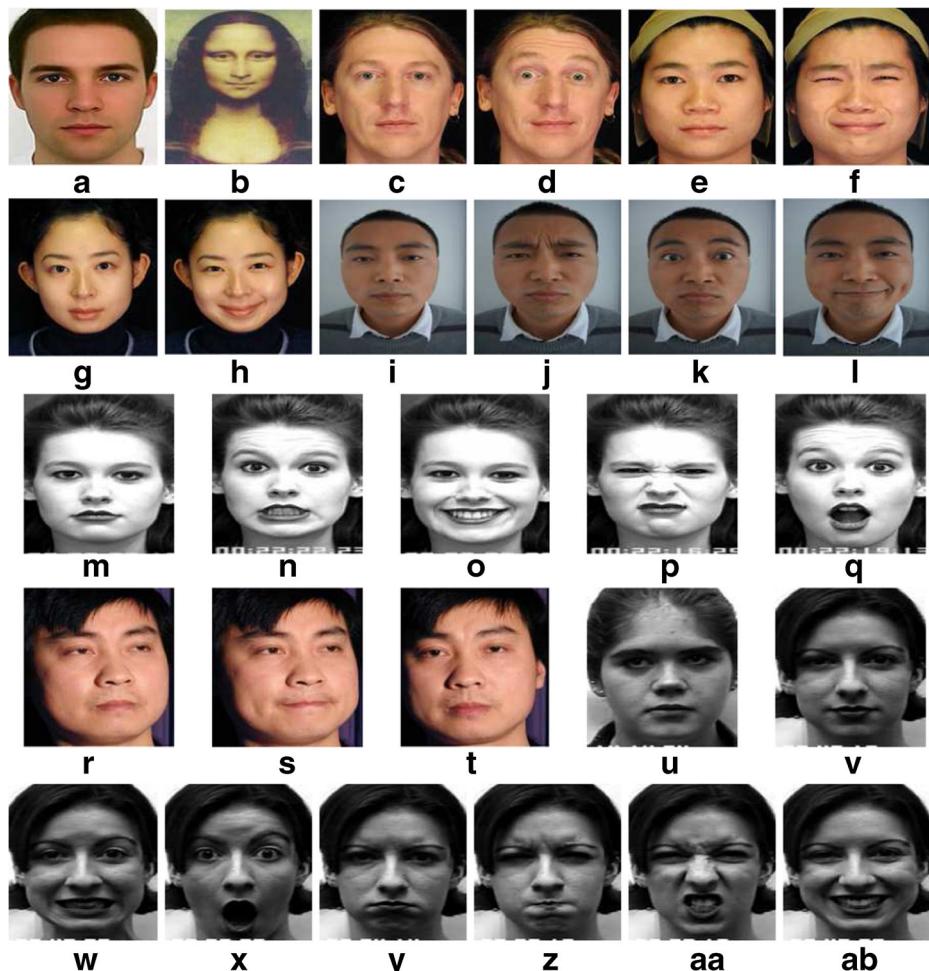
for  $(i, j) \in \delta(B_{lip}, 2), 1 \leq k \leq Len$ .

where  $InitY$ ,  $BlenY$  are the initial and blended Y components, respectively,  $Len$  is the length for blending,  $\delta(B_{lip}, 2)$  is the adjacent two rings of pixels around the lip boundary.

The proposed wrinkle mapping algorithm is significantly different from the traditional lighting ratio-based mapping. Firstly, the proposed algorithm is based on the lighting difference, which will not introduce abnormal lighting ratios when the pixel lighting intensity on  $F_{sn}$  is relatively small. Secondly, the proposed mapping uses the information of multiple adjacent pixels around each pixel to construct the lighting fitting system, which is less sensitive to abnormal pixels compared with the pixel-to-pixel mapping algorithm of ERI.

### 3 Experimental results

We perform the experiments on a PC with a 3.2 GHZ core processor and 4 GB RAM. We first test the performance of our mesh deformation in Section 3.1 and wrinkle mapping algorithms in Section 3.2, then study the expression synthesis on a face database and compare the results with state of the art algorithms in Section 3.3. The employed images are shown in Fig. 9, where (a) shows an average man face downloaded from web, (b) shows Mona Lisa's neutral face from paper [15], (c)–(h), (r)–(t) are four neutral face images together with the corresponding frown (raising-eyebrow), sad, smile and chuckle expression images used in paper [15] for testing. The expressions (i)–(l) are photographed by iPhone 5, and the third and fifth rows of expressions and (u),(v) are faces of the 74-th,11-th,10-th persons in the database [17].



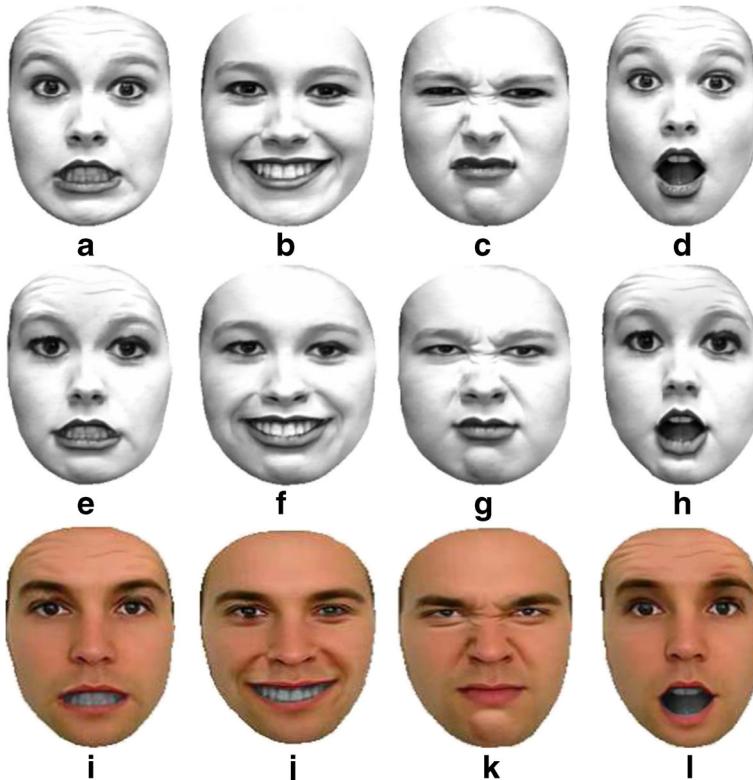
**Fig. 9** The expression set used in the experiments

### 3.1 Mesh deformation

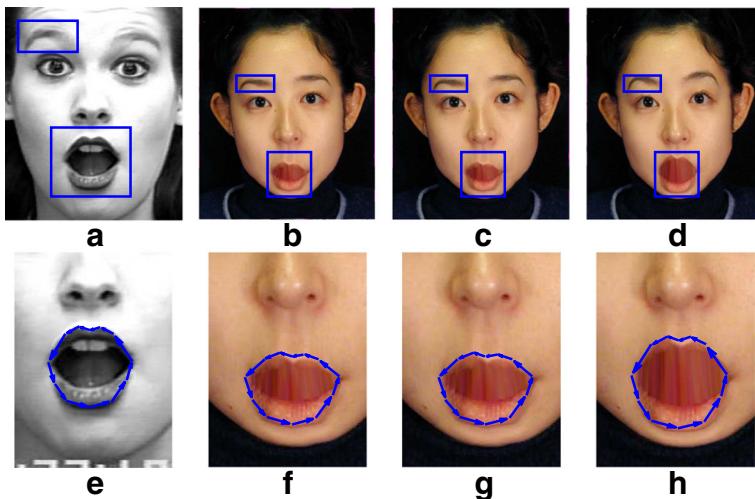
In this section, we first investigate the performance of deformation algorithm by mapping the expressions of a woman to herself, and then compare the proposed deformation with a baseline wrapping algorithm both visually and quantitatively. We use Fig. 9m as  $F_{sn}$  and  $F_{tn}$ , Fig. 9n–q as  $F_{se}$  to synthesize the fear, happy, and surprise expressions of the woman. Figure 10 shows the synthesized fear, happy, disgust and surprise expressions ((e)–(l)) of the women and the average man, together with those of ground truths ((a)–(d)).

To quantitatively evaluate the performance of the geometry preservation of the mesh deformation, similarity of the boundary directions and the runtime ( $RT$ ) of deformation are considered in this paper. When  $RT$  is the overall runtime for obtaining the colors of all the pixels on  $F_{te}$ , the mean error ( $ME$ ) of the directional vectors is defined as below:

$$ME = \frac{1}{N} \sum_{i=1}^N \frac{\left\| DV_i^{(te)} - \frac{\text{LenPart}_i^{(te)}}{\text{LenPart}_i^{(se)}} r_i^{(se), (tn)} \cdot DV_i^{(se)} \right\|_2}{\| DV_i^{(se)} \|_2} \quad (14)$$



**Fig. 10** The expression synthesis of a woman and the average man by learning the expressions of the woman. The first row are the expression faces for learning, the second and third rows are the synthesized expressions



**Fig. 11** The geometry deformations of the little girl’s face with different deformation algorithms. The source target face and mouth (**a**,**e**), the face and mouth wrapped with *IniWrap* (**b**,**f**), *SimpWrap* (**c**,**g**) and the proposed DaWF (**d**,**h**)

where  $N$  is the number of the parts,  $\text{LenPart}_i^{(te)}$  is the length of the  $i$ -th part,  $DV^{(te)}$  records all the directional vectors of the  $i$ -th part,  $r_i^{(se),(tn)}$  records the ratio of the width and height of the  $i$ -th part on  $F_{sn}$  and  $F_{tn}$ .

The wrapping algorithm in [3] obtains the position of each pixel by incorporating the directional field of several key parts on  $F_{te}$ , which is time-consuming when all the directional vectors are utilized to obtain the position of a single pixel. Thus, we adopt two modified editions of the algorithm. For *IniWrap*, each pixel is influenced by all the edges of adjacent triangles around the triangle the pixel lies in. For *SimpWrap*, each pixel is influenced only by the edges of the triangle the pixel lies in.

The comparison of the deformation algorithms *IniWrap*, *SimpWrap* and DaWF using the little girl’s face is demonstrated in Fig. 11, when Fig. 9m and g are used as  $F_{sn}$  and  $F_{tn}$ , respectively. Figure 11b, c, d show the synthesized surprise expressions using Fig. 11q as  $F_{se}$  for *IniWrap*, *SimpWrap* and the proposed DaWF, respectively. In particular, deformations of the mouth region are shown in Fig. 11f–h.

From Fig. 11b–d, it can be seen that the proposed DaWF produces more similar geometry features than *IniWrap* and *SimpWrap*. For regions labeled with blue rectangles, the deformation above the left eye by the proposed DaWF is more smooth and generates more similar blending than *IniWrap* and *SimpWrap*.

While the wrapping algorithm proposed in [3] mainly learns the information from the movements of the feature points, the proposed algorithm learns from the shape of the target part by preserving the directional field of the part edge. It is more obvious in Table 1 that the proposed DaWF achieves the minimum matching errors regarding the directional field for learning. Moreover, because a sequence of algebraic computations are needed for each pixel on the target face, *SimpWrap* is more time-consuming than DaWF. The efficiency of the proposed algorithm is verified in Table 1.

**Table 1** The comparison of two deformation algorithms in terms of geometry preservation error and runtime (s)

Algorithm	Fear (ME, RT)	Happy (ME, RT)	Disgust (ME, RT)	Surprise (ME, RT)
<i>IniWrap</i>	(0.289,4.1)	(0.347,3.7)	(0.363,3.0)	(0.304,3.2)
<i>SimpWrap</i>	(0.289,2.8)	(0.347,2.5)	(0.363,2.2)	(0.304,2.3)
DaWF	(0.106, 2.2)	(0.113, 1.8)	(0.131, 1.6)	(0.094, 1.7)

The size of  $F_{tn}$  is  $490 \times 640$

### 3.2 Wrinkle mapping

To view the performance of the wrinkle mapping, expression synthesis from gray to gray, gray to color, and color to color faces is firstly studied. Then the robustness of the proposed algorithm against large geometric deformation and lighting variation is tested. Finally, the comparison between the proposed algorithm and several wrinkle mappings available in literature are conducted.

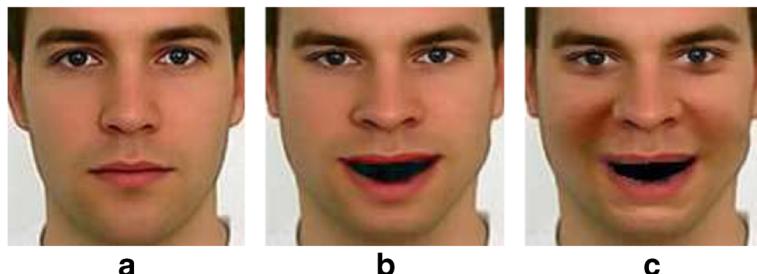
Using Fig. 9v, w and u as  $F_{sn}$ ,  $F_{se}$  and  $F_{tn}$ , respectively, the synthesized expressions from gray face to gray face with increasing intensities of geometry deformation and wrinkles are demonstrated in Fig. 12. To study the synthesis performance from gray face onto color face, the synthesized happy expression of the average man's face after geometry deformation and wrinkle mapping is presented in Fig. 13, where Fig. 9v, ab and a are used as  $F_{sn}$ ,  $F_{se}$  and  $F_{tn}$ , respectively.

To test the performance from color face onto color face, Fig. 14 presents the results of disgust, surprise and smile expressions mapped onto a little girl, a middle-aged man and a boy, when Fig. 9e, c, g are used as  $F_{sn}$  and Fig. 9f, d, h are used as  $F_{se}$ . It can be seen from Fig. 14 that not only complicated geometry shape but also obvious or hidden wrinkles on the three expressions are synthesized properly.

To test performance of the algorithm to synthesize expressions with large deformation, Fig. 15 presents the fearing, laughing (i.e. happy), disgusting and surprising Mona Lisa when Fig. 9m is used as  $F_{sn}$  and Fig. 9n–q are used as  $F_{se}$ , respectively. For these



**Fig. 12** The gradual changes of the fear expression in Fig. 9w. From (a–e), the intensities of geometry deformation and wrinkle are gradually increased with the size of opened mouth



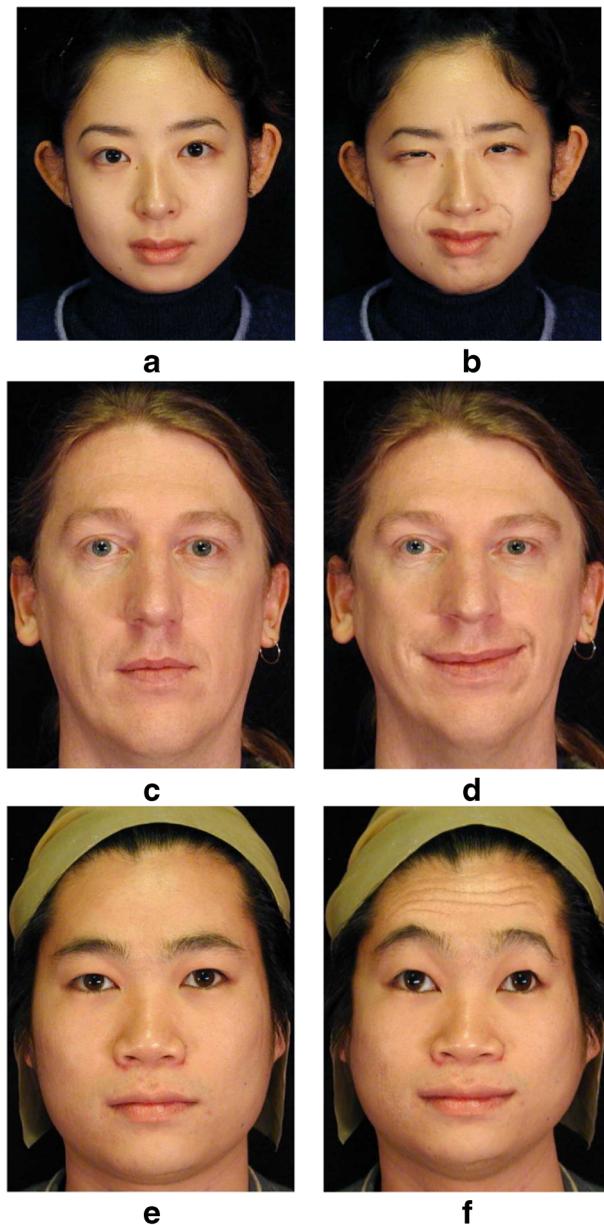
**Fig. 13** The expression synthesis of the average man's face (**a**) after geometry deformation (**b**) and wrinkle mapping (**c**)

expressions, the mouth region, eyes and brows are shown with large geometry deformations, while the proposed algorithm can effectively preserve these geometry and the corresponding texture details.

To test robustness of the proposed algorithm against large lighting variation, we use Fig. 9i as  $F_{sn}$  and Fig. 9j–l as  $F_{se}$  to synthesize the expressions of disgust, surprise and smile for the girl shown in Fig. 16. The lighting histograms of the two neutral faces shown in Fig. 16a, e demonstrate that there are great lighting differences between the two faces. However, the proposed DaWF achieves visually good wrinkle mapping performance in synthesizing the three expressions.

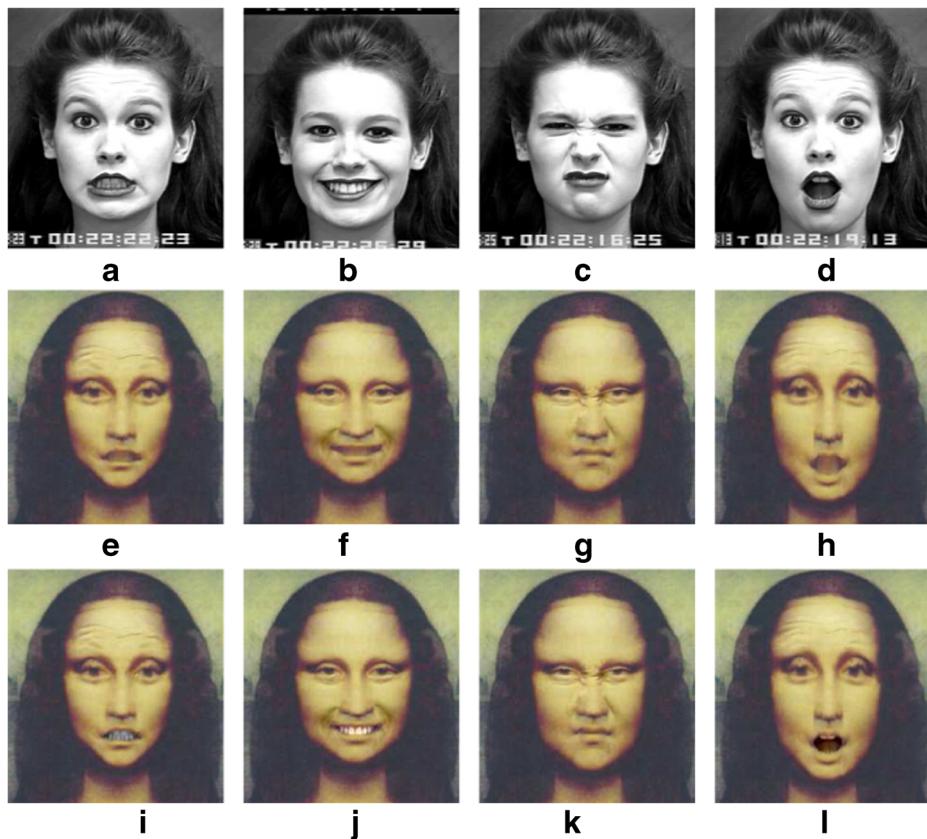
As a lighting difference based approach, we also compared our wrinkle mapping with that of three related algorithms, i.e. expression ratio image (ERI) [15], vertex tent coordinate (VTC) [25] and musical model (MD) [34]. Considering the similarity of the synthesized lighting differences  $\nabla L_{syn}$  with the ground truth differences  $\nabla L_{src}$  (i.e. the lighting difference of the corresponding pixels on source expression and source neutral faces) in Fig. 17a, Fig. 17 presents the comparative results of our algorithm with several related algorithms on sad expression synthesis using Fig. 9e, f, g as  $F_{sn}$ ,  $F_{se}$  and  $F_{tn}$ , respectively. In this test, the mesh deformation of the proposed DaWF was used for all competing approaches. The average similarity values corresponding to the algorithms ERI, VTC, MD and DaWF in Fig. 17b–e on two cheek regions are 0.59, 0.78, 0.81 and 0.89, respectively, where the lighting differences with absolute values larger than one-tenth of the ground truth are considered. Meanwhile, it is demonstrated in Fig. 17a–e that the proposed DaWF preserves more complete wrinkles than the other algorithms on the cheek region. To demonstrate the detailed information of these algorithms on lighting preservation, Fig. 17f–j present lighting differences between the source and synthesized expressions on the region between two eyes, two cheeks and the lower jaw by different algorithms. While the lighting differences between corresponding pixels on  $F_{sn}$  and  $F_{se}$  are demonstrated in Fig. 17f, the lighting differences between  $F_{tn}$  and  $F_{te}$  by the algorithms ERI, VTC, MD and the proposed DaWF are presented in Fig. 17g–j, respectively. One can observe from Fig. 17g–j that the wrinkle difference obtained by the proposed DaWF is more similar to that of Fig. 17f than the other algorithms.

In the paper [15], Fig. 9r, s, t were used to test the algorithm performance on synthesizing frontal expression based on non-frontal wrinkle features, where ERI format was reported to lose efficiency. In this work, we use Fig. 9r, s, t as  $F_{sn}$ ,  $F_{se}$  and  $F_{tn}$  for the testing. It can be seen from Fig. 18a that the dimples in the both face cheeks are not obvious, expression



**Fig. 14** Three neutral and synthesized expressions. **a, b:** The neutral and synthesized sad expressions of a little girl using Fig. 9e and f as  $F_{sn}$  and  $F_{se}$ . **c, d:** The neutral and synthesized smile expressions of a middle-aged man using Fig. 9g and h as  $F_{sn}$  and  $F_{se}$ . **e, f:** The neutral and synthesized frown expressions of a boy using Fig. 9c and d as  $F_{sn}$  and  $F_{se}$

synthesized in Fig. 18c by ERI does not obtain wrinkles as distinct as those on the face shown in Fig. 18a. However, the wrinkles on the left cheek synthesized by the proposed algorithm shown in Fig. 18d are much more similar to those on the source expression, which

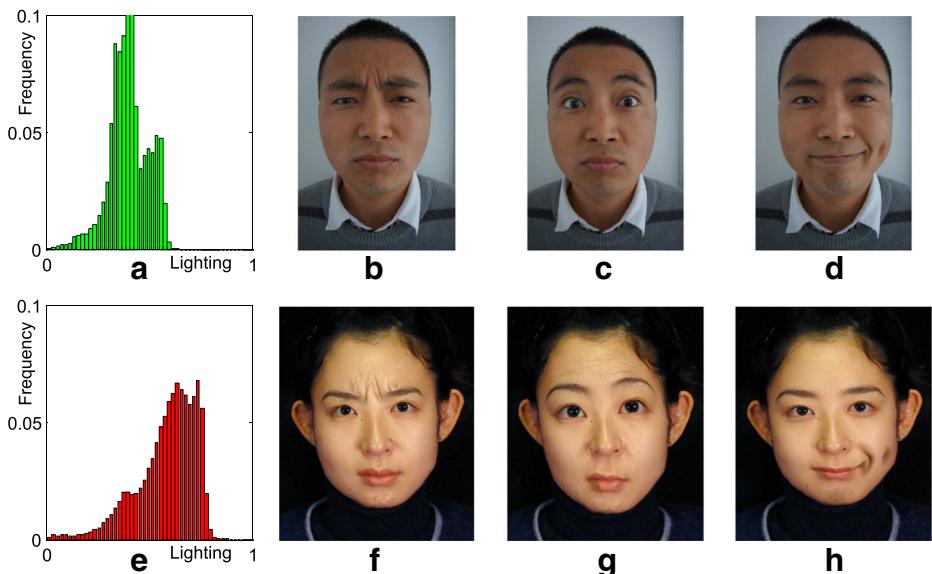


**Fig. 15** Synthesis of expressions with large deformation. The 1st-3rd rows demonstrate the expressions for learning, the expressions synthesized with geometry deformation and wrinkle mapping, and the embedded teeth regions, respectively

further illustrates that the proposed algorithm outperforms ERI on preserving non-obvious features presented in the non-frontal face.

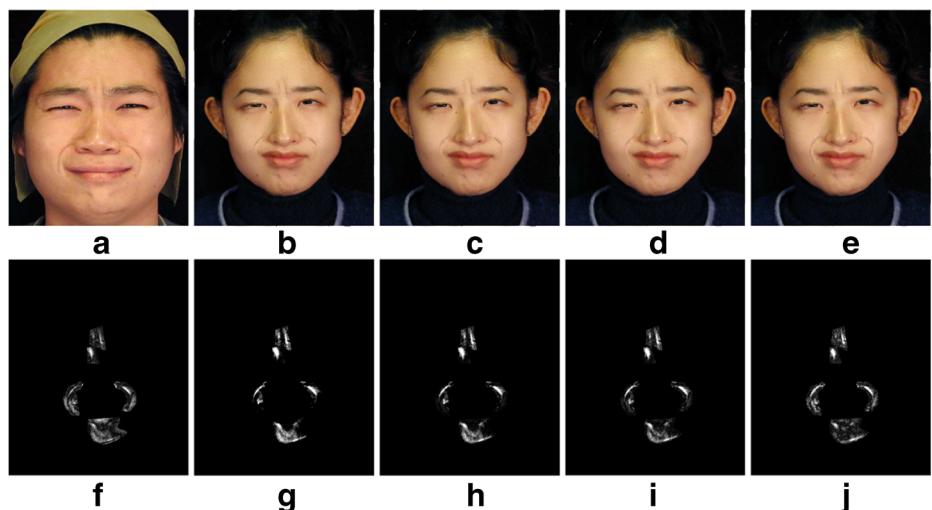
### 3.3 Expression synthesis comparison

To fully compare the performance of different expression synthesis algorithms, we first list the operations included in each algorithm, and then compare the results of these algorithms on an expression database with objective evaluation in Section 3.3.1 and subjective evaluation in Section 3.3.2. Table 2 lists the preprocessing steps and parameters of the algorithms ERI, VTC, MD and DaWF for the comparison. The ERI algorithm requires several preprocessing steps and manual determination of the correspondences of feature points. The correspondences of feature points in VTC also need manual adjustment. For MD algorithm, it introduces several parameters and a database of expressions is needed for weight decision. Compared with these algorithms, the proposed DaWF needs less manual operations and only the coefficients of weights need to be determined. The overall runtime cost of all stages for the proposed algorithm is presented in Table 3, where runtime of correspondence

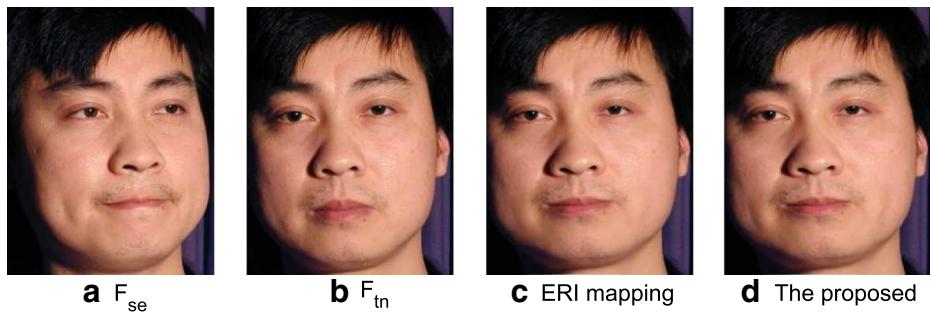


**Fig. 16** The synthesis of disgust, surprise and smile expressions with large lighting differences. Images (a), (e) are the lighting histograms of  $F_{sn}$  and  $F_{tn}$ , respectively. b–d demonstrate the expressions of disgust, surprise and smile for learning, f–h show the synthesized expressions of a little girl

matching only includes the processing time on  $F_{se}$ . It can be seen from the table that each expression can be synthesized within one minute, which illustrates the applicability of the proposed algorithm in the real application.



**Fig. 17** The comparison of four algorithms on wrinkle mapping. a is the source sad expression, b–e are sad expressions synthesized by the algorithm ERI [15], VTC [25], MD [34] and the proposed DaWF, respectively. f reflects the lighting difference between  $F_{sn}$  and  $F_{se}$  on four key wrinkle regions, g–j demonstrate the lighting differences between  $F_{tn}$  and  $F_{te}$  by the algorithms ERI, VTC, MD, DaWF, respectively



**Fig. 18** The comparison of ERI and the proposed mapping for non-frontal expression synthesis

### 3.3.1 Objective evaluation of the synthesis results

To compare the performance quantitatively, two metrics are defined to compute the robustness and similarity of the synthesis results. One metric formulated in (15) is defined as the minimum sum of the largest  $N = 1000$  biases of the lighting differences between two faces, which reflects the robustness of an algorithm against noise. The other metric formulated in (16) is defined as the correlation of the lighting differences between the source and synthesized faces, which reflects the similarity of  $F_{se}$  and  $F_{te}$ .

$$BiasD = \min_a \sum_{i=1}^N \left| \frac{\nabla L_{src} - a \cdot \nabla L_{syn}}{\nabla L_{src}} \right|^{(i)}. \quad (15)$$

where  $\nabla L_{src}$  is the lighting difference between the corresponding pixels on  $F_{sn}$  and  $F_{se}$ ,  $\nabla L_{syn}$  is the lighting difference between  $F_{tn}$  and  $F_{te}$ ,  $(\cdot)^{(i)}$  denotes the  $i$ -th largest number of the considered vector.

$$CorD = \frac{\langle \nabla L_{src}, \nabla L_{syn} \rangle}{\|\nabla L_{src}\|_2 \cdot \|\nabla L_{syn}\|_2}. \quad (16)$$

where  $\langle \cdot, \cdot \rangle$  is the inner product of two vectors.

The runtime ( $RT$ ) of expression synthesis is also considered in the comparison, where the time cost on extracting the wrinkle regions is not included, as all the considered algorithms need to manually determine the locations of wrinkle regions. Only the runtime of deformation and wrinkle mapping stage listed in Table 3 is considered. For the programming implementation, hybrid of MATLAB and C++ is employed to make use of the advantage of MATLAB software at the matrix computation and the efficiency of C++ at loop iteration.

**Table 2** Comparison of operations and parameters among different algorithms

	Algorithm	Preprocessing Needed	Lighting Mapping	Additional Parameters
ERI [15]	Smoothness filtering	ERI	$c$	
VTC [25]	Correspondence modification	Covariance of ratios	—	
MD [34]	A database of expressions	Shift ERI	$h(u_0, v_0), \lambda_k, w_{db}, r(u_0, v_0)$	
DaWF	—	Lighting difference	$\lambda_1-\lambda_3$	

**Table 3** Overall runtime (seconds) of the proposed algorithm to process the faces shown in Fig. 16

Expression (Size $F_{tn}$ )	Feature Point Location and Correspondence Matching	Wrinkle Extracting	Deformation and Wrinkle Mapping	Overall Runtime
Disgust ( $746 \times 621$ )	10.4	10.8	8.1	29.3
Surprise ( $746 \times 621$ )	11.2	11.2	7.4	29.8
Smile ( $746 \times 621$ )	10.9	8.6	7.6	27.1

The smoothing operator and lighting mapping in ERI, the lighting mapping with shift ERI in MD algorithm and the procedures of constructing coefficient matrices in VTC and DaWF are coded in C++ to accelerate the implementation.

A face database consists of neutral faces of 20 subjects with lighting and pose variations is employed for testing. Each neutral face was used as  $F_{tn}$  to synthesize the six expressions shown in the last row of Fig. 9. The  $BiasD$ ,  $CorD$  and  $RT$  of the four competing algorithms are then recorded and listed in Table 4. Figure 19 shows the six expressions synthesized by different algorithms for an example subject.

Considering the runtime of expression synthesis in Table 4, ERI and MD algorithms require the least computational time as these algorithms map the lighting ratios pixel by pixel onto the target face directly. While the runtime of ERI mainly concentrates on the smoothness operation and lighting mapping, that of MD mainly concentrates on the weight and lighting intensity computation. However, ERI algorithm requires manual location of feature point correspondence and MD needs a procedure of variable determination, which increase their time costs. For the VTC algorithm, normal direction of the mesh, an additional point in the normal direction and matrix inverse calculation for constructing the equation system make the algorithm more time consuming than ERI and MD, when the wrinkle regions are large. The proposed DaWF also needs to construct an equation system, which makes it more time consuming than ERI and MD. However, the runtime of VTC and DaWF relies on not only the size of face  $F_{tn}$  but also the sizes of wrinkle regions. When the size of wrinkle region is small, these fitting-based algorithms outperform ERI based algorithms because the overall smooth operation is not needed.

Regarding to the metric of maximum bias, the proposed DaWF almost achieves the best performance among the four algorithms. As the ERI algorithm in [15] learns from the pixel-by-pixel ratio of the lighting intensities on  $F_{sn}$  and  $F_{se}$ , it is easy to introduce lighting noise even after a smoothness filtering. Meanwhile, the ERI formula will introduce some

**Table 4** Objective and subjective evaluations of the six expressions synthesized by different algorithms

Algo.	Objective			Subjective					
	$RT(s)$	$BiasD(\%)$	$CorD(\%)$	<i>Fear</i>	<i>Surprise</i>	<i>Sadness</i>	<i>Angry</i>	<i>Disgust</i>	<i>Happy</i>
ERI	5.1	$8.4 \pm 0.7$	$72 \pm 6$	$7.5 \pm 1.1$	$6.6 \pm 1.0$	$7.2^\dagger \pm 0.7$	$6.8^\dagger \pm 0.7$	$6.6^\dagger \pm 1.2$	$7.0 \pm 1.2$
VTC	8.3	$4.5 \pm 0.8$	$78 \pm 8$	$7.5 \pm 0.8$	$7.3 \pm 0.9$	$7.1^\dagger \pm 0.7$	$6.9^\dagger \pm 0.5$	$6.5^\dagger \pm 1.0$	$6.9^\dagger \pm 0.8$
MD	5.6	$4.9 \pm 0.7$	$77 \pm 5$	$7.3^\dagger \pm 1.4$	$7.4 \pm 1.0$	$7.3 \pm 0.9$	$7.0^\dagger \pm 0.6$	$6.8^\dagger \pm 1.0$	$7.3 \pm 0.8$
DaWF	7.1	$41 \pm 0.8$	$81 \pm 8$	$7.7 \pm 1.1$	$6.9 \pm 0.9$	$7.7 \pm 0.8$	$8.2 \pm 0.6$	$7.7 \pm 0.9$	$7.6 \pm 1.0$

$\dagger$  indicates that the proposed DaWF achieves significantly better performance than the considered algorithm with Wilcoxon rank sum test at 5% significance level

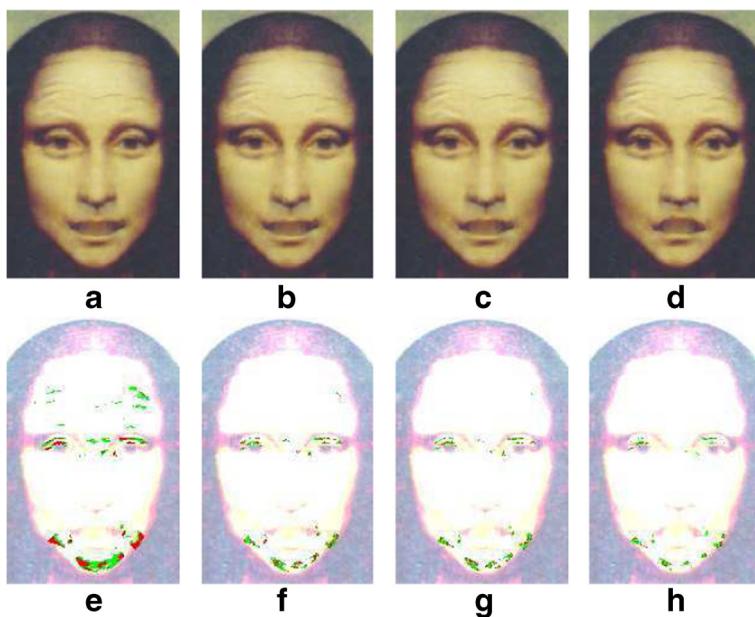


**Fig. 19** The comparison of four algorithms on six expressions

abnormal target lighting intensities when the ratio of the lighting intensities of  $F_{sn}$  and  $F_{se}$  is large, e.g. the lighting  $l_{i,j}^{sn}$  is a small number compared with  $l_{i,j}^{se}$ . Adopting a modified form of ERI operator (shift ERI), the muscle model in [34] has similar problem although Gaussian damping operator is adopted. The VTC algorithm in [25] adopts a fitting of the lighting intensities and tries to preserve the lighting covariance before and after expression synthesis. However, this algorithm is still based on the ratio of the lighting intensity, it can not solve large lighting variation when lighting  $l_{i,j}^{sn}$  is a small value compared with  $l_{i,j}^{se}$ . Different from these algorithms, the proposed DaWF adopts a fitting of the lighting difference for the wrinkle mapping. Thus, the noise is restrained and abnormally large lighting intensities on  $F_{te}$  are not introduced. It can be seen from  $BiasD$  values in Table 4 that VTC and DaWF achieve relatively better performance than ERI and MD.

Figure 20 gives an example of the fear expression synthesized by different algorithms for Mona Lisa's face, where the red pixels record the high biases with  $BiasD$  larger than 0.05, and the green pixels record the moderate biases with  $BiasD$  between 0.02 and 0.05. It can be seen that the mouth shape of the synthesized frown expression by DaWF is more similar to the source fear expression in Fig. 9n than those by other algorithms. Meanwhile, the size of regions labeled with colors in Fig. 20h is smaller than those of regions in Fig. 20e–g. Thus, biases of lighting differences between the synthesized and source faces are smaller than those obtained by other algorithms, which further proves the advantages of the proposed DaWF over the competing algorithms.

Considering the metric of similarity, the difference form of the lighting adopted in the proposed DaWF achieves comparable results with the other algorithms. Wrinkle mappings in ERI and MD are based on the ratio of lighting intensities, which may introduce abnormal lighting intensities into the target face when the corresponding lighting intensities on the



**Fig. 20** The fear expressions synthesized by ERI, VTC, MD and the proposed DaWF are presented in (a–d). The corresponding values of  $BiasD$  are demonstrated in e–h: high (red), moderate (green) and low (non-labeled)

pixels of  $F_{sn}$  and  $F_{se}$  differ much. Although trying to preserve the covariances of the lighting intensities, the VTC algorithm adopts a similar ratio form of lighting which may result in similar abnormal results. While the proposed DaWF attempts to preserve the difference of lighting intensities insensitive to abnormal lighting ratios, it is more robust and can map subtle lighting differences when the lighting intensity  $I_{i,j}^{sn}$  on  $F_{sn}$  is small. One can observe from  $Cor D$  values in Table 4 that DaWF achieves relatively better performance than the other algorithms.

### 3.3.2 Subjective evaluation of the synthesis results

We further evaluate the performance of the these algorithms subjectively with the help of ten volunteers. In the testing, each volunteer independently scores each synthesized expression with scores from five to ten, where higher score corresponds to more genuine and similar expression to the source expression. The ten scores of each synthesized expression are recorded to derive the average score and the standard deviation, Wilcoxon rank sum test method [7] (nonparametric method) with significance level 5% is employed to study the performance difference between the competing algorithms and the proposed algorithm in Table 4. It can be seen from Table 4 that the proposed DaWF achieves the highest scores on most of the six expressions and significantly better performance is achieved on some testing expressions.

More expressions synthesized by the proposed algorithm are presented in Figs. 21 and 22. It can be seen from the two figures that the proposed algorithm achieves visually genuine expressions on most of the faces despite of different lighting conditions and head poses.



**Fig. 21** Expression synthesis with face database. The first column demonstrates source neutral face  $F_{sn}$  and six source expression faces  $F_{se}$ . The first row demonstrates target neutral faces  $F_{tn}$ , the others are the synthesized faces  $F_{te}$



**Fig. 22** Expression synthesis with face database. The first column demonstrates source neutral face  $F_{sn}$  and six source expression faces  $F_{se}$ . The first row demonstrates target neutral faces  $F_{tn}$ , the others are the synthesized faces  $F_{te}$

## 4 Discussion and conclusion

In this work, an algorithm of expression synthesis (DaWF) with novel strategies of correspondence matching, mesh deformation and wrinkle mapping is proposed. The proposed DaWF has several advantages. Firstly, the algorithm is semi-automatic in obtaining the correspondences between sets of feature points. Secondly, statistical learning of the expressions w.r.t. the considered person is not needed. Thirdly, not only the displacements of the feature points but also the geometry features of the expression are learned in the face deformation. Lastly, the fitting form of the lighting differences makes the generated wrinkles more robust and complete.

Experimental results on various expressions and comparison with state-of-the-art algorithms illustrate the effectiveness of the proposed deformation algorithm on geometry shape preservation and the robustness of the wrinkle mapping on detail texture preservation. Although visually genuine expressions are synthesized by the proposed algorithm, there is still much margin for further improvements. Firstly, the algorithm of view morphing in [24] or 3D reconstruction algorithm in [5] can be integrated to synthesize expressions for largely rotated face in 3D space. Secondly, the runtime of lighting fitting in the procedure of wrinkle mapping needs to be improved by incorporating the GPU parallel computation. Thirdly, wrinkle detection algorithm should be designed and incorporated into the proposed algorithm to automatically locate the wrinkle regions. Since the embedded teeth region may significantly influence the genuineness of the synthesized expression, it is important to select the most similar teeth region. Based on the assumption that the teeth texture is most related with the surrounding mouth region, sparse dictionary learning could be used to match teeth regions in terms of face color, lighting condition and mouth shape. Further work will concentrate on the applications of the proposed DaWF such as tinting, which needs a good correspondence between each pixel between the source and target faces. Other applications are expression recognition and expression invariant face recognition.

**Acknowledgments** The authors thank the anonymous reviewers for their helpful comments and suggestions. The work was supported by Natural Science Foundation of China under grant no. 61672357, 61602315, 61402289 and 61272050, China Postdoctoral Science Foundation under grant no. 2015M572363, the Science Foundation of Guangdong Province under grant no. 2014A030313556 and 2014A030313558.

## References

1. Al-Osaimi F, Bennamoun M, Mian A (2009) An expression deformation approach to non-rigid 3d face recognition. *Int J Comput Vis* 81(3):302–316
2. Bando Y, Kuratake T, Nishita T (2002) A simple method for modeling wrinkles on human skin. In: the 10th Pacific Conference on Computer Graphics and Applications, IEEE Computer Society, p 166
3. Beier T, Neely S (1992) Feature-based image metamorphosis. In: ACM SIGGRAPH Computer Graphics, ACM, pp 35–42
4. Beymer D, Shashua A, Poggio T (1993) Example based image analysis and synthesis. PhD thesis, Artificial Intelligence Laboratory, Massachusetts Institute of Technology
5. Blanz V, Bass C, Poggio T, Vetter T (2003) Reanimating faces in images and video. *Comput Graph Forum* 22(3):641–650
6. Debevec P (1998) Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography. In: SIGGRAPH, ACM, pp 189–198

7. Gibbons J, Chakraborti S (2003) Nonparametric Statistical Inference. Marcel Dekker
8. Huang D, De la Torre F (2010) Bilinear kernel reduced rank regression for facial expression synthesis. In: the 11th European Conference on Computer Vision. Springer-Verlag, Verlag, pp 364–377
9. Kazemi V, Josephine S (2014) One millisecond face alignment with an ensemble of regression trees. In: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 1867–1874
10. Kotsia I, Pitas I (2007) Facial expression recognition in image sequences using geometric deformation features and support vector machines. *IEEE Trans Image Process* 16(1):172–187
11. Larboulette C, Cani M (2004) Real-time dynamic wrinkles. In: the Computer Graphics International, IEEE Computer Society, pp 522–525
12. Lee Y, Terzopoulos D, Waters K (1995) Realistic modeling for facial animation. In: the 22nd Annual Conference on Computer Graphics and Interactive Techniques, ACM, pp 55–62
13. Li K, Dai Q, Wang R, Liu Y, Xu F, Wang J (2014) A data-driven approach for facial expression retargeting in video. *IEEE Trans Multimedia* 16(2):299–310
14. Liao J, Lima R, Nehab D, Hoppe H, Sander P, Yu J (2014) Automating image morphing using structural similarity on a halfway domain. *ACM Trans Graph* 33(5):1–12
15. Liu Z, Shan Y, Zhang Z (2001) Expressive expression mapping with ratio images. In: the 28th Annual Conference on Computer Graphics and Interactive Techniques, ACM, pp 271–276
16. Lu X, Jain A (2008) Deformation modeling for robust 3d face matching. *IEEE Trans Pattern Anal Mach Intell* 30(8):1346–1356
17. Lucey P, Cohn J, Kanade T, Saragih J, Ambadar Z, Matthews I (2010) The extended cohn-kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPRW, IEEE, pp 94–101
18. Matthews I, Xiao J, Baker S (2007) 2d vs. 3d deformable face models: representational power, construction, and real-time fitting. *Int J Comput Vis* 75(1):93–113
19. Pan G, Han S, Wu Z, Zhang Y (2010) Removal of 3d facial expressions: a learning-based approach. In: Computer Vision and Pattern Recognition, CVPR, IEEE, pp 2614–2621
20. Qian K, Wang B, Chen H (2014) Automatic flexible face replacement with no auxiliary data. *Comput & Graph* 45(0):64–74
21. Raouzaïou A, Tsapatsoulis N, Karpouzis K, Kollias S (2002) Parameterized facial expression synthesis based on mpeg-4. *EURASIP J Appl Signal Processing* 2002(1):1021–1038
22. Ren S, Cao X, Wei Y, Sun J (2014) Face alignment at 3000 fps via regressing local binary features. In: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 1685–1692
23. Sanchez A, Ruiz J, Moreno A, Montemayor A, Hernandez J, Pantrigo J (2011) Differential optical flow applied to automatic facial expression recognition. *Neurocomputing* 74(8):1272–1282
24. Seitz S, Dyer C (1996) View morphing. In: the 23rd Annual Conference on Computer Graphics and Interactive Techniques, ACM, pp 21–30
25. Song M, Dong Z, Theobalt C, Wang H, Liu Z, Seidel H (2007) A generic framework for efficient 2-d and 3-d facial expression analogy. *IEEE Trans Multimedia* 9(7):1384–1395
26. Sumner R, Popovic J (2004) Deformation transfer for triangle meshes. *ACM Trans Graph* 23(3):399–405
27. Tzimiropoulos G, Pantic M (2013) Optimization problems for fast aam fitting in-the-wild. In: IEEE International Conference on Computer Vision, ICCV, IEEE, pp 593–600
28. Vlasic D, Brand M, Pfister H, Popovic J (2005) Face transfer with multilinear models. *ACM Trans Graph* 24(3):426–433
29. Weise T, Bouaziz S, Li H, Pauly M (2011) Realtime performance-based facial animation. *ACM Trans Graph* 30(4):76–79
30. Weng Y, Xu W, Wu Y, Zhou K, Guo B (2006) 2d shape deformation using nonlinear least squares optimization. *Visual Comput* 22(9–11):653–660
31. Xie W, Shen L, Yang M, Hou Q (2015) Lighting difference based wrinkle mapping for expression synthesis. In: 2015 8th International Congress on Image and Signal Processing, CISIP, IEEE, pp 636–641
32. Yu H, Garrod O, Jack R, Schyns P (2015) A framework for automatic and perceptually valid facial expression generation. *Multimed Tools Appl* 74(21):9427–9447
33. Zhang Q, Liu Z, Guo B, Terzopoulos D, Shum H (2006) Geometry-driven photorealistic facial expression synthesis. *IEEE Trans Vis Comput Graph* 12(1):48–60
34. Zhang Y, Lin W, Zhou B, Chen Z, Sheng B, Wu J (2014) Facial expression cloning with elastic and muscle models. *J Vis Commun Image Represent* 25(5):916–927



**Weicheng Xie** received the BS degree in statistics from Central China Normal University in 2008, the MS degree in probability and mathematical statistics and PhD degree in computational mathematics from Wuhan University, China in 2010 and 2013. He has been a visiting research fellow with School of Computer Science, University of Nottingham, UK. He is now a postdoctoral researcher at Shenzhen Key Laboratory of Spatial Smart Sensing and Services, School of Computer Science and Software Engineering, Shenzhen University. His current researches focus on image processing and facial expression analysis.



**Linlin Shen** received the B.Sc. degree from Shanghai Jiaotong University, Shanghai, China, and the Ph.D. degree from University of Nottingham, Nottingham, U.K., in 2005. He was a Research Fellow with the Medical School, University of Nottingham, researching brain image processing of magnetic resonance imaging. He is currently a Professor and the Director of the Computer Vision Institute, College of Computer Science and Software Engineering, Shenzhen University, China. His current research interests include Gabor wavelets, face/palmprint recognition, medical image processing, and image classification.



**Meng Yang** is currently an associate professor at School of Computer Science & Software Engineering, Shenzhen University, Shenzhen, China. He received his Ph.D degree from The Hong Kong Polytechnic University in 2012. Before joining Shenzhen University, he has been working as Postdoctoral fellow in the Computer Vision Lab of ETH Zurich. His research interest includes sparse coding, dictionary learning, object recognition and machine learning. He has published 8 CVPR/ICCV/ECCV papers and several IJCV, IEEE TNNLS and TIP journal papers. Now his Google citation is over 1700.



**Jianmin Jiang** received the Ph.D. degree from University of Nottingham, Nottingham, U.K., in 1994. He joined Loughborough University, Loughborough, U.K, as a Lecturer in computer science. From 1997 to 2001, he was a Full Professor of Computing with the University of Glamorgan, Wales, U.K. In 2002, he joined the University of Bradford, Bradford, U.K, as a Chair Professor of Digital Media, and Director of Digital Media and Systems Research Institute. In 2014, he moved to Shenzhen University, Shenzhen, China, to carry on holding the same professorship. He is also an Adjunct Professor with the University of Surrey, Guildford, U.K. His current research interests include image/video processing in compressed domain, computerized video content understanding, stereo image coding, medical imaging, computer graphics, machine learning, and AI applications in digital media processing, retrieval, and analysis. He has published over 400 refereed research papers. Prof. Jiang is a Chartered Engineer, a member of EPSRC College, and EU FP-6/7 evaluation expert. In 2010, he was elected as a scholar of One-Thousand-Talent-Scheme funded by the Chinese Government.