A Novel Transient Wrinkle Detection Algorithm and Its Application for Expression Synthesis

Weicheng Xie, Linlin Shen, and Jianmin Jiang

Abstract—**Because facial wrinkle is a representative feature of facial expression, automatic wrinkle detection has been an important and challenging topic for expression simulation, recognition, and animation. Recently, most works about wrinkle detection have focused on permanent wrinkles (e.g., age wrinkles), which are usually linear shapes, whereas the detection of transient wrinkles (e.g., expression wrinkles) has not been sufficiently studied because of their shape diversity and complexity. In this work, a novel algorithm for automatic detection of transient wrinkles with linear, fixed, and chaotic shapes is proposed, which largely consists of edge pair matching, active-appearance-model-based wrinkle structure location, and support-vector-machine-based wrinkle classification. The proposed wrinkle detector is applied for expression synthesis and an improved Poisson wrinkle mapping approach is proposed. Experimental results illustrate the competitiveness of the proposed wrinkle detector in detecting different transient wrinkles. Compared with state-of-the-art algorithms, the proposed approach yields complete and accurate wrinkle centers. The expression synthesized by the improved wrinkle mapping is also much more realistic.**

Index Terms—**Edge detection, expression synthesis, transient wrinkle, wrinkle structure.**

I. Introduction

CURRENTLY, with the development of research on facial expression, the features of face expression have gained increasing attention for related applications. The wrinkle is the most prominent and representative feature; its research and applications have been topics of high interest.

Jain and Park [1] employed face marks including wrinkles for face identification, and more applications rely on age estimation, skin texture classification [2], expression recognition [3] and simulation [4], [5]. The location of wrinkle positions is critical for applications such as facial digital beauty [6], [7], in which the positions of wrinkle lines and regions must be detected before they are removed or trimmed. Related wrinkle detection can also be applied to retinal vessel detection [8]. For the application of transient wrinkle detection, it is productive to refer to 2D expression synthesis [9]–[11] and 3D expression simulation [12], [13]. However, wrinkle regions continue to be manually located in these works, which might largely restrict their applicability. In the application of facial expression recognition [14]–[17], because transient wrinkles are the representative and accompanying features of the expression, they are frequently employed as features for recognition. However, these location algorithms are limited to some simple wrinkles, and they are not sufficiently accurate or general. To improve the efficiency of these algorithms, wrinkle detection should be automatic and general.

For the analysis of the wrinkles, we roughly divide the wrinkles into two categories, permanent and transient wrinkles. Permanent wrinkles are usually located on the faces of the aged person, whereas transient wrinkles appear in a relatively wide region when the expression is generated, such as in the cheek region. Thus, the detection of transient wrinkles is more complicated and challenging.

Concerning the detection of permanent wrinkles with linear shape, Ng *et al.* [18], [19] applied a hybrid Hessian filter to locate the whole wrinkle line directly rather than the wrinkle edges or centerlines. Forehead wrinkles were detected by growing and stitching wrinkle centerline segments extracted from filtered images of the maximum Gabor response with different thresholds [20]. An active contour algorithm with snakelet was employed to detect thin age wrinkles [21]. However, randomness and sensitivity to the initial location of snakelets largely restricted this algorithm's efficiency. Wrinkles were detected by convolving the normalized image with finely tuned Gabor filters in the dominant directions [22]. This algorithm works for a distinctly thin wrinkle whose boundary edge can be properly detected. However, the approach is not efficient for blurry transient wrinkles, whose boundary edges might not be detected properly by the edge detectors. Batool and Chellappa [23] proposed to detect age wrinkles by marked point processes, and the reversible jump Markov chain Monte Carlo algorithm was used to locate the initial positions of wrinkles.

Manuscript received February 10, 2016; revised June 19, 2016 and August 31, 2016; accepted September 21, 2016. Date of publication September 28, 2016; date of current version January 17, 2017. The work was supported by the Natural Science Foundation of China under Grant 61272050, Grant 61672357, and Grant 61602315, by the Science Foundation of Guangdong Province under Grant 2014A030313556, by the China Postdoctoral Science Foundation under Grant 2015M572363, by the Science and Technology Innovation Commission of Shenzhen under Grant JCYJ20160422144110140, and by the Open Research Fund of China-U.K. Visual Information Processing Laboratory. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Cha Zhang. *(Corresponding author: Linlin Shen.)*

W. Xie is with the Computer Vision Institute, School of Computer Science and Software Engineering, and the Shenzhen Key Laboratory of Spatial Information Smart Sensing and Services, Shenzhen University, Shenzhen 518060, China, and also with the School of Computer Science, University of Nottingham, Nottingham NG8 1BB, U.K. (e-mail: wcxie@szu.edu.cn).

L. Shen is with the Computer Vision Institute, School of Computer Science and Software Engineering, and the Shenzhen Key Laboratory of Spatial Information Smart Sensing and Services, Shenzhen University, Shenzhen 518060, China (e-mail: llshen@szu.edu.cn).

J. Jiang is with the Research Institute for Future Media Computing, Shenzhen University, Shenzhen 518060, China (e-mail: jianmin.jiang@szu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2016.2614429

Because permanent wrinkles are often relatively simple with distinct shapes, detection algorithms aiming at this type of wrinkle might not be useful for transient wrinkles with variant and blurry shapes.

Considering transient wrinkle detection, the active shape model (ASM) was employed to train and locate 81 feature points including several points in the nasolabial region [16]. Furthermore, the wrinkles in some pre-extracted fixed regions were detected by geometry features such as the mean curvature change. Yin and Basu [24] selected several relatively fixed regions with low texture correlation to the corresponding region of the initial frame as the active wrinkle region for expression transition. Although these algorithms can provide a quantitative score indicating whether the candidate region is an active wrinkle using geometry deformation or texture correlation degree, they cannot accurately locate the positions of the transient wrinkles in these regions. Huang *et al.* [17] located feature points on relatively fixed positions such as the nasolabial field for active wavelet network (AWN) training; however, such a location is inapplicable to wrinkles near the corner of the mouth. Azazi *et al.* [14] used a genetic algorithm to detect transient wrinkle features in the expression-sensitive regions, in which the population-based evolutionary algorithm is not efficient. Tian *et al.* [25] used the ratio between the numbers of edge pixels in the wrinkle areas of the current frame and the first frame to judge whether there were wrinkles in the considered region. This algorithm can only locate wrinkles in pre-extracted regions by approximating to two lines. For wrinkle detection by edge detection and matching, Popa *et al.* [26] proposed an approach to edge pair construction for wrinkles on garments, in which the wrinkle lines were extracted by some trimming procedures; then, the pairs of wrinkles were determined. However, the wrinkles to be detected had to be regular, i.e., the line pairs with linear shape shared similar tangent directions, which made this method not applicable for irregular wrinkles. The wrinkle regions were located in [6] by texture orientation field, incorporating Markov random field modeling. An expectation-maximization algorithm was employed to classify skin versus skin wrinkles/imperfections, which only locates the range of the wrinkles; accurate positions and shapes are not provided. Chen *et al.* [27] obtained 3D expression wrinkles with patch-based regression of a high-fidelity expression feature database. Gabor wavelet [28] and maximum Gabor response [20] had also been used to detect transient wrinkle features and edge pixels. Although these algorithms are applicable to transient wrinkles with a distinct boundary, they do not work for wrinkles with a blurry border.

Current algorithms leave much room for further improvement and are summarized according to three characteristics. First, current wrinkle detectors rely on a distinct boundary texture of the wrinkle region. Second, the geometry deformation and texture information around the wrinkle lines are not fully utilized. Finally, current quantitative metrics are not sufficiently complete or effective to discriminate wrinkle from non-wrinkle regions.

To make wrinkle detection more effective and general, this work divides transient wrinkles into linear, fixed and chaotic shapes. Aiming at all types of wrinkles, a novel and general algorithm for automatic detection of transient wrinkles is proposed, which is useful not only for permanent wrinkles with linear shape but also for transient wrinkles with various shapes and randomly distributed positions. Furthermore, an improved wrinkle mapping form is proposed and applied to expression synthesis. A sketch map of this work is presented in Fig. 1.

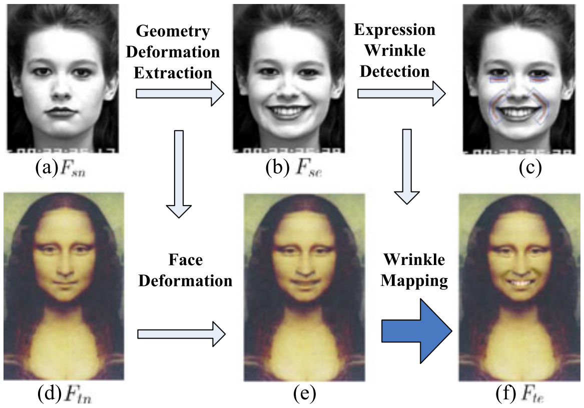

Fig. 1. Overview of the work. From (b) to (c), transient wrinkle regions are detected, which are mapped onto the deformed face (e) to synthesize the target expression face (f). From (a) to (b), geometry deformation is learned and applied onto the target neutral face (d) to obtain the deformed face (e).

The main contribution of this work is summarized as follows

1) An effective and general algorithm for transient wrinkle location is proposed;
2) Complete quantitative metrics of transient wrinkle structure are formulated for wrinkle discrimination;
3) An efficient form of wrinkle mapping is proposed for expression synthesis.

This paper is structured into the following sections. The proposed wrinkle detection algorithm and its application on expression synthesis are demonstrated in Section II. Then, the experimental results and the corresponding illustrations are demonstrated in Section III. Finally, the conclusion and discussions are presented in Section IV.

II. The Proposed Algorithm

A. *Framework of Algorithm*

As illustrated in Fig. 1, this paper is divided into two parts: the algorithm for transient wrinkle detection and its application in expression synthesis. The procedure for detecting transient wrinkles is demonstrated in Fig. 2.

The proposed detector largely consists of three steps. First, the Canny edge detector is applied to the input face, and pairs of continuous wrinkle edges are detected. Second, an active appearance model (AAM) is employed to find all of the candidate wrinkle lines, in which wrinkle structure around each wrinkle is constructed and adjusted. Finally, quantitative metrics are defined and used for SVM classification to discriminate wrinkle regions from non-wrinkle regions. These steps are summarized in Fig. 2 and will be introduced separately in the following three sections.

B. *Wrinkle Edge Detection and Pair Matching*

Edge detection and pair matching are the first step applied for wrinkles with linear and chaotic shapes. Forehead and chin re-

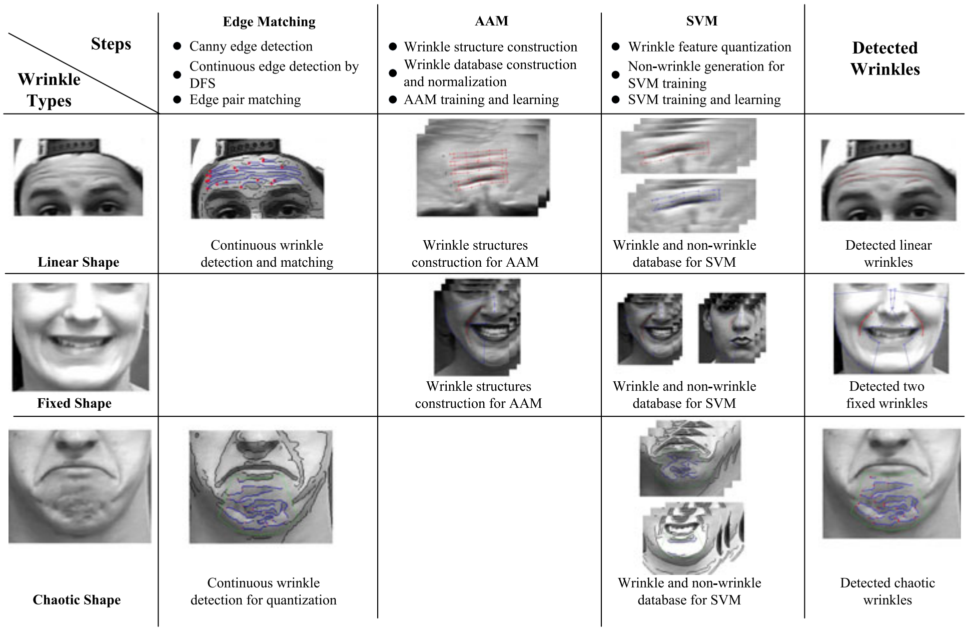
Steps
Wrinkle Types
Edge Matching
Canny edge detection
Continuous edge detection by DFS
Edge pair matching
AAM
Wrinkle structure construction
Wrinkle database construction and normalization
AAM training and learning
SVM
Wrinkle feature quantization
Non-wrinkle generation for SVM training
SVM training and learning
Detected Wrinkles
Linear Shape
Continuous wrinkle detection and matching
Wrinkle structures construction for AAM
Wrinkle and non-wrinkle database for SVM
Detected linear wrinkles
Fixed Shape
Wrinkle structures construction for AAM
Wrinkle and non-wrinkle database for SVM
Detected two fixed wrinkles
Chaotic Shape
Continuous wrinkle detection for quantization
Wrinkle and non-wrinkle database for SVM
Detected chaotic wrinkles

Fig. 2. Framework of transient wrinkle detection. Categories of linear, fixed, and chaotic wrinkles are demonstrated in the 1st column, the main steps of the algorithm are presented in the 2nd–4th columns, and the final detected wrinkles are shown in the last column.

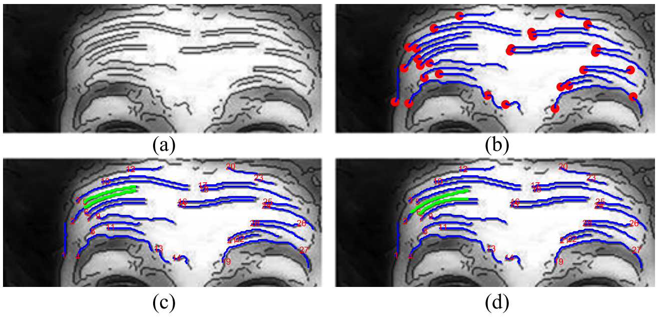

Fig. 3. Illustration of wrinkle edge pair matching. The edge pixels detected by the Canny detector are presented in (a). The continuous lines found after DFS and some trimming operators are demonstrated in (b), in which red circles record the starting pixels. The green lines in (c) and (d) denote two example edge pairs enclosing a wrinkle and non-wrinkle regions, respectively.

gions are initially extracted; then, the Canny edge detector with Gaussian filter [29] is employed to find the edge pixels in the considered region. Based on the detected edges, the procedure of continuous edge line detection and matching consists of the following steps:

1) Extract the pixels with only one adjacent edge pixel, and find the continuous wrinkle edges starting from the pixel by depth first search (DFS);
2) Abandon some short and distorted lines with length less than $CritLen = 10$ and cumulative rotation angle greater than $CritAng = \pi$; then, divide the line into several parts at the position where the rotation angle is greater than a critical level of $CritAng$;
3) Determine all of the candidate line pairs with small distance.

Fig. 3(a) and 3(b) present the wrinkle pixels detected by the Canny edge detector and continuous wrinkle edges searched by DFS on a forehead.

For both thick and thin wrinkles, the wrinkle centerline can lie in either the intersection region of a pair of edges or on a wrinkle edge. Thus, to decrease the possibility of missing detection, two types of regions are considered candidates for wrinkle detection:

1) The intersection region between the pair of detected wrinkle edges;
2) The region with single wrinkle edge as the center.

For the former case, the most challenging problem is to determine which two wrinkle edges are a pair. In this work, wrinkle lines with the preceding three least distances between each considered line are chosen for AAM wrinkle location and SVM discrimination, in which the distance between each pair of edges l_1, l_2 is defined as

$$\begin{cases} \mathcal{D}(l_1,l_2) = \frac{1}{2}(StDist + TrDist) \\ StDist = MIN(\mathcal{D}(l_1^{(1)}, l_2), \mathcal{D}(l_2^{(1)}, l_1)) \\ StP = argmin_{(i,j)}(\mathcal{D}(l_1^{(1)}, l_2^{(j)}), \mathcal{D}(l_2^{(1)}, l_1^{(i)})) \\ TrDist = MIN(\mathcal{D}(l_1^{(n_1)}, l_2), \mathcal{D}(l_2^{(n_2)}, l_1)) \\ TrP = argmin_{(i,j)}(\mathcal{D}(l_1^{(n_1)}, l_2^{(j)}), \mathcal{D}(l_2^{(n_2)}, l_1^{(i)})) \\ \mathcal{D}(P,l) = MIN_i(||P - l^{(i)}||_2) \end{cases} \tag{1}$$

where the wrinkle edges l_1, l_2 have been adjusted to be the same direction, $l^{(i)}$ is the ith integer point on edge l, and $l_1^{(n_1)}, l_2^{(n_2)}$ are the terminating points of wrinkle edges l_1, l_2, respectively. $StDist$ (or $TrDist$) is the minimum distance between the starting (or terminating) points and wrinkle edges, and $StP = (StP_1, StP_2), TrP = (TrP_1, TrP_2)$ record the $x - y$ positions on l_1, l_2 where the minimum distances $StDist, TrDist$ are achieved, respectively.

To remove the short line segments, the wrinkle pairs with intersection region length less than $CritLen = 10$ are abandoned.

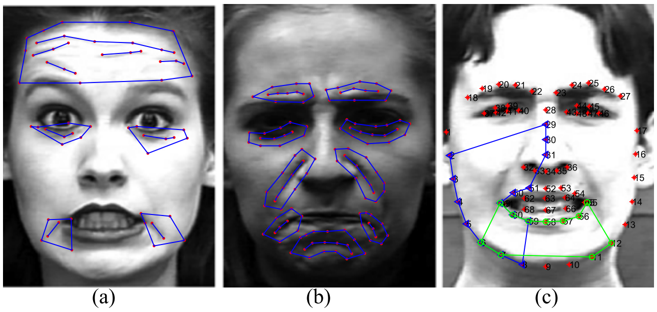

Fig. 4. (a) and (b) are two example faces from wrinkle database. The line segments are the centers of extracted wrinkle regions, the polygon regions record the range of these lines. (c) shows two fixed regions for AAM training and learning. Regions labeled by blue triangles and green circles are for locating left nasolabial and chin wrinkles, respectively.

For regions with minor wrinkles, the critical length $CritLen = 10$ should be decreased. The green lines in Fig. 3(c) are a pair of intersection wrinkle parts enclosed by the line segments $(l_1^{(StP_1)}, l_1^{(TrP_1)})$ and $(l_2^{(StP_2)}, l_2^{(TrP_2)})$, which are used for further wrinkle centerline location and discrimination. The green wrinkles in Fig. 3(d) are another candidate pair of wrinkle edges in which no wrinkle exists in the enclosed region.

For the region centered around a single line, two virtual wrinkle edges are generated in the normal direction of the considered line, whose closed regions are chosen for further testing. With this approach, not only the regular wrinkle pairs but also some irregular wrinkle regions are included; thus, the possibility of missing detection can be decreased.

The wrinkle edge detector provides the candidate region of wrinkle between each two edges, whereas the accurate wrinkle center between them must be determined for further wrinkle discrimination. Thus, AAM in the following section is introduced to locate the wrinkle centerline.

C. AAM-Based Wrinkle Structure Location

Once the candidate wrinkle regions are located by edge detection and matching, wrinkles with linear and fixed shapes are further refined using the wrinkle structure, which is motivated from the structure of facial feature points [30] and is defined as a set of feature points with fixed sequence extracted from the interior line and its exterior edge pair. Because the wrinkle structures of different persons might appear quite different, AAM is applied for such purpose. In this work, wrinkle structure-based AAM was trained using face images with different expressions. The wrinkles on the faces in the CK+ database [31] are manually located for AAM training. Fig. 4(a) and 4(b) show the wrinkles located for two example faces, in which the lines record the positions of the wrinkle center. The corresponding closed region denotes a region expanded from the wrinkle center in the tangent and normal directions with predetermined distance ratios.

1) Forehead Wrinkle Structure: For the wrinkles on the forehead region, each wrinkle structure contains 19 points with 14 points on the outer boundary edges (wrinkle edges) and 5 points on the wrinkle line, which are iso-parametrically extracted from the corresponding B-spline interpolation curves of the considered lines, such as the points labeled with red stars in Fig. 6. To apply the AAM algorithm, the training sets of feature points should share a similar shape under Procrustes transformation; thus, the structure of the wrinkle should satisfy the following conditions:

1) The direction of the wrinkle line should be the same as the first side of the wrinkle region;
2) The sequence of all of the feature points should be in a fixed order, i.e., clockwise or counter-clockwise;
3) The 1st and 7th feature points should lie on the convex side of the test shape.

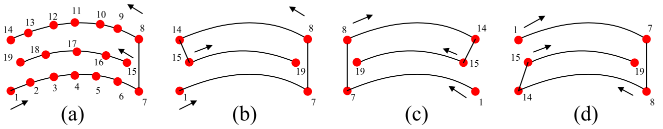

Fig. 5. Illustration of wrinkle structure adjustment. The points on the wrinkle line $\{15 : 19\}$ in (a) are reordered to make the entire shape be a wrinkle structure in (b). The point sequence is adjusted from counter-clockwise in (b) to clockwise in (c). These points are reordered to make the 1st and 7th points in (c) be on the convex side of wrinkle structure in (d).

To satisfy the first condition, we must only ensure that the dot product of the mean directions of the wrinkle line and the first side of the wrinkle structure (the preceding 7 points) is positive.

Fig. 5 presents an example of adjustment from a set of disordered points to the target sequence by reordering the point sequence to clockwise 5(c) and the convex shape on the side of the 1st and 7th points 5(d).

In Fig. 5, the wrinkle structure is set clockwise by uniforming the mean directions of points $\{P_1, \cdots, P_7\}$ and $\{P_8, \cdots, P_{14}\}$; i.e., we must only ensure that the rotation directions at three corner points P_7, P_8, P_{14} satisfy

$$|[P_i - P_{i-1}, 0; P_{i+1} - P_i, 0; BaseDir]| < 0, i = \{7, 8, 14\} \tag{2}$$

where $|\cdot|$ represents the determinant of a matrix, and $BaseDir = [0, 0, 1]$.

To make points P_1, P_7 lie on the convex side of the wrinkle structure, we must only check the condition

$$\mathcal{D}'(P_1, \overline{P_8 P_{14}}) + \mathcal{D}'(P_7, \overline{P_8 P_{14}}) < \mathcal{D}'(P_1^t, \overline{P_8 P_{14}}) + \mathcal{D}'(P_7^t, \overline{P_8 P_{14}}). \tag{3}$$

where $P_1^t = \frac{P_1 + P_2}{2}$, $P_7^t = \frac{P_6 + P_7}{2}$, $\mathcal{D}'(\cdot, \cdot)$ represents the distance from a point to a line, rather than the distance to the line segment defined in (1).

2) Fixed Shape Wrinkle Structure: Although the wrinkles in the forehead region have linear shape, the shapes of wrinkles in the nasolabial and chin regions are more variant, and the edges of these wrinkles are blurry with the adjacent regions. Thus, only finding the wrinkle edge pairs and locating the wrinkle lines no longer works. However, each category of wrinkles in these regions has a relatively fixed shape. For example, all of the wrinkles in the nasolabial regions, shown as blue triangles in Fig. 4(c), have a similar shape. The wrinkle is close to the nose and mouth tip and has a small distance to the muscular line

TABLE I

Point Sequences of Six Fixed Regions and the Corresponding Features for Extracting the Wrinkles From Wrinkle Database, in Which $A : B$ and $B : -1 : A$ Represent $A, A + 1, \cdots , B - 1, B$ and $B, B - 1, \cdots , A - 1, A$, Respectively

Region Label	Region Name	Point Sequence	Features for Extracting the Wrinkles
$R1$	Nasal Root	22, 23, 43, 31, 40	Between two eyes, with small distance to $\overline{P_{40}P_{43}}$.
$R2$	Left Pouch	2, 37, 42, 41, 40, 31	Below and near the left eye, with small distance to $\overline{P_{37}P_{40}}$.
$R3$	Right Pouch	16, 46, 47, 48, 43, 31	Below and near the right eye, with small distance to $\overline{P_{43}P_{46}}$.
$R4$	Left Cheek	2:8, 59, 60, 49, 50, 51, 31, 30, 29	Near the left mouth tip, with small distance to $\overline{P_{29}P_{6}}$.
$R5$	Right Cheek	16:-1:10, 57:-1:53, 31:-1:29	Near the right mouth tip, with small distance to $\overline{P_{29}P_{12}}$.
$R6$	Chin	6, 7, 11, 12, 55:60, 49	Below the mouth, with small distance to $\overline{P_{6}P_{12}}$ and relatively large length.

linking the 6th and 29th points. This type of wrinkle is defined as a fixed shape wrinkle.

Thus, for fixed shape wrinkles such as nasolabial wrinkles, the wrinkle structures must be constructed to utilize the local deformation for shape variance modeling. Table I lists the wrinkle structures defined for different regions, including the point sequences of outer boundary points and the features for extracting the training wrinkles in these considered regions. The numbers in Table I are the indices of the feature points labeled in the CK+ database [31]. Fig. 4(c) presents two example structures in the nasolabial and chin regions.

For a fixed wrinkle region, the initial wrinkle line approximating the center of the blurry wrinkle region must be located for the AAM learning, which is approximated with inverse Procrustes transformation of the mean positions of the training database. Procrustes transformation is used to eliminate the difference between shapes by affine transformation of scaling, translation and rotation, which is defined as follows:

$$TS = b \cdot S \cdot T + C \tag{4}$$

where b, T, C are the scaling, rotation matrix and translation vectors; S, TS are the initial and the transformed shapes, respectively.

For locating the initial wrinkle, the mean shape S_0 of all of the training wrinkles under Procrustes transformation is obtained; then, the estimated wrinkle positions are determined as follows:

$$\begin{cases} OS = Ob \cdot OS_0 \cdot OT + OC \\ IS = Ob \cdot IS_0 \cdot OT + W \cdot OC \\ W_{i,j} = \frac{1/d_{i,j}}{\sum_{k=1}^{n_O} 1/d_{ik}}, i = 1, \cdots , n_I, j = 1, \cdots , n_O \end{cases} \tag{5}$$

where Ob, OT, OC are the scaling, rotation matrix and translation vectors of the Procrustes transformation. OS_0, IS_0, n_O, n_I record the feature points and the corresponding numbers on the outer boundary and wrinkle line, respectively, OS is the input fixed shape points, IS is the estimated points of the wrinkle line, and $d_{i,j}$ is the distance from the ith interior point on the wrinkle line of S_0 to the jth exterior point of S_0. To obtain a better initial position, the initialized location IS in (5) is further locally adjusted such that the improved wrinkle line initialization TS_{imp} matches the darkest region in a local region around IS. The local adjustment is presented as follows:

$$TS_{imp} = \min_{is \in \Omega_{IS}} \sum_i l_{is_i} \tag{6}$$

where Ω_{IS} is a local testing region around shape IS constructed by translating IS in its normal and tangent directions, and l_{is_i} is the lighting intensity of the ith pixel is_i on trial line is.

3) AAM: After determining the candidate wrinkle regions, accurate positions of wrinkles in these regions are located by the AAM algorithm, which not only utilizes the deformation information of the shape but also tracks the wrinkle edge from the texture aspect.

AAM consists of shape and texture models [30], [32]. Before the shape modeling of a new wrinkle, the Procrustes transformation is applied to eliminate the shape differences of the training wrinkle structures under affine transformation. Then, principle component analysis (PCA) is employed on the shape variations to retain the principal information of identity, pose and expression. For a new wrinkle structure s with n feature points $s = (x_1, y_1 \cdots , x_n, y_n)^T$, the shape model is presented as

$$s = s_0 + Sp, p = S^T(s - s_0) \tag{7}$$

where S represents the eigenvectors of the shape database, and s_0 is the mean shape of S.

For an input image texture I of the wrinkle structure, wrap this texture to the mean shape s_0 with a piecewise warping model W to obtain a shape-free texture $\hat{I}$, which is further represented as the linear form of the principal texture variations

$$\hat{I} = A_0 + Ac, c = A^T(I - A_0) \tag{8}$$

where A_0, A represent the mean wrinkle texture and texture database, respectively.

Considering the shape of the input wrinkle structure, the feature points are obtained by minimizing the L_2 error between the input image and the modeled image using the relationship between the shape and texture model. The simultaneous optimization of model parameters p, c is demonstrated in [32] as follows:

$$\min_{p,c} ||I(W(x,p)) - A_0 - Ac||^2. \tag{9}$$

Tzimiropoulos and Pantic [30] proposed a fast form by optimizing p, c separately and iteratively to speed the optimization procedure. The proposed algorithm in this paper is based on the optimization form of AAM in [30].

To decrease the time complexity of the AAM training, a subdatabase of wrinkle shapes similar to the input face shape are initially extracted by a sparse representation, whose L_1 form is represented as follows:

$$\min_{w} \frac{1}{2}||s - Dw||_2^2 + \lambda ||w||_1 \tag{10}$$

where D is vectorization $(x_1, y_1, \cdots, x_n, y_n)$ of the wrinkle structures after Procrustes transformation, w is the sparse representation coefficient corresponding to wrinkle structure database D, s is the vectorization of the input shape, and λ is the regularization parameter fixed as $1e^{-3}$ in this work after several trials [33].

D. SVM-Based Wrinkle Classification

When different types of wrinkles are located by AAM or the edge matching described in the last section, they are input to SVM in this paper for further verification.

1) Quantitative Wrinkle Measurement: Different from texture or geometric features with 3D Gabor [34], wavelets [35] and curvature [36], the product of length and thickness was used [22] to assess intuitively wrinkling intensity. In this paper, we defined some novel quantitative metrics and used them as wrinkling features for SVM classification. Although the boundaries of wrinkle often show with relatively bright areas, the center of a wrinkle region is usually dark. We argue that the darker the wrinkle region is, the deeper the wrinkles are and the more significant deformation is implied [12].

The quantitative metrics defined in this paper are largely based on the Y component of the YUV face images and are listed as follows: the lighting standard variance $(StdLV)$ defined in (11), the mean lighting difference $LigDiff$ between the exterior and interior pixels of the wrinkle structure defined in (12), the absolute bias $LigBias$ between two sides of the considered wrinkle line in normal direction defined in (13), and the ratio $LigContinu$ of the lighting sum of central pixels on the wrinkle line divided by the lighting sum of all of the pixels defined in (14).

The metric $StdLV$ reflects the distribution range of the lighting intensities; the larger the value is, the larger is the possibility of a wrinkle.

$$StdLV = \sqrt{\frac{1}{N}\sum\nolimits_{i=1}^{N}(l_i - \mu)^2}, \mu = \frac{1}{N} l_i \tag{11}$$

where $\{l_i, i = 1, \cdots, N\}$ are the lighting intensities of pixels on the wrinkle line.

Wrinkle lines often show up with dark lighting; thus, the difference between the mean lighting values of pixels on the outer boundary and the wrinkle line can reflect the degree of wrinkling. From visual inspection, the larger the $LigDiff$ is, the deeper the wrinkle is.

$$LigDiff = \frac{1}{N_O}\sum_{i=1}^{N_O} l_i - \frac{1}{N_I}\sum_{i=1}^{N_I} l_i \tag{12}$$

where N_O, N_I are the numbers of pixels on the exterior ($\overline{P_1P_7}$ and $\overline{P_8P_{14}}$) and interior ($\overline{P_{15}P_{19}}$) lines of the wrinkle structure in Fig. 5(d).

To quantify the symmetry of both sides of the wrinkle line, the bias between the mean lighting values of pixels on both sides of the wrinkle line in the normal direction is employed. This metric is defined to abandon candidate wrinkle lines distributing on the face part boundary such as the lower eye boundary, in which the central line can be darker than the adjacent region, whereas the lighting intensities on the two sides of the boundary differ significantly.

$$LigBias = \left|\frac{1}{N_{UP}}\sum_{i=1}^{N_{UP}} l_i - \frac{1}{N_{DOWN}}\sum_{i=1}^{N_{DOWN}} l_i\right| \tag{13}$$

where N_{UP}, N_{DOWN} are the numbers of pixels on the two exterior boundaries of the wrinkle line in the normal direction.

To describe the continuity of the wrinkle line, the ratio in (14) is defined as the ratio between the lighting sums of pixels on the central line segment and the whole wrinkle line. This metric is used to remove the candidate wrinkle regions enclosing the end parts of two adjacent wrinkles, and the detected wrinkle line can show the feature that the end pixels on the wrinkle line are darker than are the central pixels.

$$LigContinu = \left|\frac{1}{N_{CI}}\sum_{i=1}^{N_{CI}} l_i - \frac{1}{N_I}\sum_{i=1}^{N_I} l_i\right| \tag{14}$$

where N_{CI} is the number of pixels on the central part of the wrinkle line.

Because the chaotic wrinkles are often relatively short, small and with random shapes, it makes little sense to locate the accurate positions of each wrinkle line. Thus, we only determine whether there are chaotic wrinkles in the considered region. For this type of wrinkle, we also define four metrics to discriminate wrinkle regions from non-wrinkle regions.

First, to reduce the effect of the boundary of the considered binary mask region, the erosion operator is performed to shrink the binary mask region into a smaller region in which the disk structure element is employed, and its size is set to be 5% of the minimum of the length and width of the chin region.

To quantitatively analyze the chaotic wrinkles, the following four metrics are defined and employed:

1) The number of wrinkle edges;
2) The information entropy of the lengths of the searched wrinkle edges;
3) The sum of the pixel lighting differences on both sides of the wrinkle edges;
4) The entropy of lighting differences with respect to (w.r.t.) the wrinkles;

The first metric can depict the chaotic degree of the considered wrinkle region. The more chaotic the region is, the more wrinkle fragments there are.

The second metric depicts the degree of inconformity of the searched edge lengths. If the wrinkle region is regular, only several wrinkles are long. However, for the chaotic region, the lengths of the detected wrinkle edges should be more uniform. Such a difference can be measured by information entropy depicting the dispersion degree of a set of numbers, which is formulated as follows:

$$DISP(p) = -\sum_{i=1}^{n} p_i log(p_i). \tag{15}$$

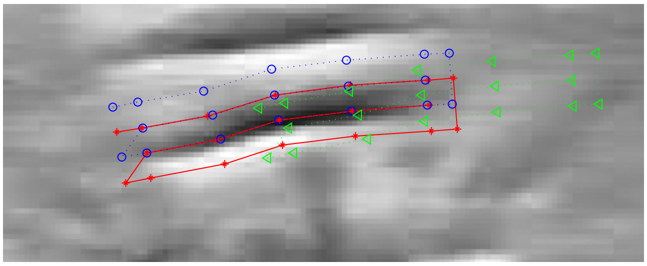

Fig. 6. Illustration of non-wrinkle region generation. Pixel sequences labeled with red stars, blue circles, and green triangles are the wrinkle structures of initial, upward, and rightward translations.

To quantify the second metric, p_i is set to be the ratio $LenR_i$ as follows:

$$LenR_i = \frac{LEN_i}{\sum_{j=1}^{n} LEN_j} \tag{16}$$

where LEN_i is the length of the ith searched wrinkle edge, n is the number of wrinkle edges.

The third term depicts the difference degree between the lighting intensities of pixels in the exterior and interior regions of wrinkles, which is defined as follows:

$$LigBiasS = \sum_{i=1}^{n} \sum_{j=1}^{LEN_i} (MAX(l_{i,j}^{EXT}, l_{i,j}^{INT}) - MIN(l_{i,j}^{EXT}, l_{i,j}^{INT})) \tag{17}$$

where $l_{i,j}^{EXT}, l_{i,j}^{INT}$ are the lighting intensities of pixels on both sides of the jth pixel in the normal direction of the ith wrinkle line.

The fourth metric depicts the degree of dispersion of the lighting difference in (17), which is defined by setting p_i in (15) to $LigDisp_i$ as follows:

$$LigDisp_i = \frac{\sum_{j=1}^{LEN_i} (MAX(l_{i,j}^{EXT}, l_{i,j}^{INT}) - MIN(l_{i,j}^{EXT}, l_{i,j}^{INT}))}{LigBiasS}. \tag{18}$$

The value $DISP(LigDisp)$ will be large when the lighting differences are uniformly distributed on all of the wrinkle edges.

2) SVM Classification: After the quantitative metrics of wrinkle regions are defined, SVM-based binary classification in [37] is employed to distinguish genuine wrinkles from false wrinkles. Although the quantitative metrics defined in (11)–(14) are used as the four-dimensional feature vector for wrinkles with linear and fixed shapes, the metrics defined in (15)–(18) are used as the features for wrinkles with chaotic shape. The default solver $C - SVC$ with RBF kernel ($C = 1$) is employed.

For SVM training, the non-wrinkle regions should be generated. Concerning linear and fixed shape wrinkles, each two wrinkles are a certain distance away from one another in the normal direction of the wrinkle line. We generate the non-wrinkle regions by moving the wrinkle regions in mean normal and horizontal directions. Thus, the generated region can be guaranteed to be non-wrinkle, and has the same shape and similar illumination with the wrinkle region. Fig. 6 presents two non-wrinkle regions in the normal and tangent directions corresponding to the linear shape wrinkle labeled with red stars. The number of non-wrinkle regions generated for SVM training is 60 for linear shape wrinkles and equal to the number of considered fixed shape regions for fixed shape wrinkles.

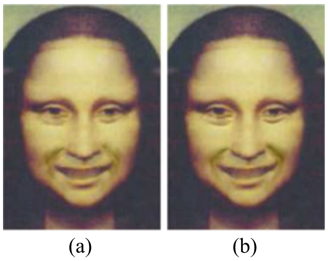

Fig. 7. Comparison of generated wrinkles with the initial (a) and improved (b) Poisson wrinkle mappings.

E. Application for Expression Synthesis

As shown in Fig. 1, the problem of expression synthesis is to generate target expression face F_{te} by applying the deformation and wrinkling texture, learned from the source neutral face F_{sn} and source expression face F_{se}, to the target neutral face F_{tn}.

The extraction of texture features such as the transient wrinkles is a preliminary step of this problem. However, current works [9], [11], [38] locate the transient wrinkle manually, which restricts their efficiency and applicability. Moreover, the algorithms of wrinkle mapping in these works are mostly related to the expression ratio image (ERI) between F_{sn} and F_{se} [9]. This mapping is sensitive to small pixel lighting intensities on F_{se}. In this work, we propose to transfer the lighting difference rather than the ratio for wrinkle mapping, which is similar to 3D displacement mapping [13]. The coefficient of the extent of lighting change is learned by linear regression.

The wrinkle mapping in this algorithm is based on region mapping, which is an improved version of seamless cloning [39] for general texture mapping

$$|N_p|T_p - \sum_{q \in N_p} T_q = \sum_{q \in N_p} (S_p - S_q) \tag{19}$$

where T_p is the lighting intensity of the pth pixel on the target face, T_q records the lighting of the adjacent pixels, S_p, S_q are the lighting intensities of corresponding pixels on the source face, $N_p = 4$ is the number of adjacent pixels.

The texture mapping in (19) only maps the average lighting difference around a considered pixel, whereas lighting intensities around a pixel can vary significantly in a local region. Thus, the mapped wrinkle might not be as distinct as the source features. As shown in Fig. 7, the generated wrinkles in Fig. 7(a) obtained by (19) are not as distinct as those in Fig. 19(h).

Aiming at the application of expression synthesis, we improve the mapping form in (19) in three aspects:

1) Change the number of adjacent pixels from four to eight;
2) Change the number of equations corresponding to each pixel from one to eight;

3) Learn the ratio coefficient of the lighting differences by linear regression.

The improved form of wrinkle mapping is presented as follows:

$$\begin{cases} T_p - T_q = \lambda_r \cdot (S_p - S_q), \text{ for } q \in N_p, \\ \lambda_r = \mu_r \cdot \frac{CE_{tn}}{CE_{sn}} \\ \text{where } \mu_r = argmin_\mu \sum_i (r_n^{(i)} \cdot \mu - r_e^{(i)})^2, \\ r_n^{(i)} = \frac{CE_{tn}^{(i)}}{CE_{sn}^{(i)}}, r_e^{(i)} = \frac{CE_{te}^{(i)}}{CE_{se}^{(i)}} \end{cases} \tag{20}$$

where $N_p = 8$, CE_{sn} and CE_{tn} are the changed extents of lighting intensities of the ith wrinkle region on F_{sn} and F_{tn}, respectively, which are approximated by the difference of the 95% largest and 5% largest lighting intensities of the wrinkle region. $CE_{sn}^{(i)}, CE_{se}^{(i)}, CE_{tn}^{(i)}, CE_{te}^{(i)}$ are the changed extents of pixel lighting intensities in boundary ring regions of the ith wrinkle mask on $F_{sn}, F_{se}, F_{tn}, F_{te}$, respectively. The employed variable λ_r is the lighting weight between source neutral and target neutral faces of all of the wrinkle regions, and variables $r_n^{(i)}$ and $r_e^{(i)}$ are the lighting ratios between the source and target faces with respect to neutral and expression face boundary regions, respectively. The lighting ratio μ_r of the entire wrinkle region is approximated with ratio $r_e^{(i)}/r_n^{(i)}$ on the wrinkle region boundary.

With this modification, the retained wrinkle features are more distinct than those retained by the mean form in equation (19). As illustrated in Fig. 7, the nasolabial wrinkles in Fig. 7(b) are more distinct than are those in Fig. 7(a).

To locate the facial shape and key feature points for correspondence matching, 77 feature points including the forehead points are located by ASM [40]; then, a triangulation mesh is generated based on these feature points. A state-of-the-art face deformation based on an elastic model [10] is adopted, which was reported to be able to produce a shape more similar to the target expression than does the face deformation algorithm [41]. To obtain the teeth texture with mouth open, the teeth region on the source expression face F_{se} is mapped to the corresponding region on the target expression face F_{te} with the proposed wrinkle mapping. The details of gray-scale processing of color faces, triangulation generation, and pixel correspondence are the same as in [9]–[11].

III. Experimental Results

We perform the experiments on a PC with a 3.2 GHZ core processor and 4 GB RAM. To evaluate the detection error, the metric in [19] is adopted with slight modification. The overall runtime (RT) is considered for the expression synthesis application. A hybrid of MATLAB and C++ is employed to make use of the advantage of MATLAB software at matrix computation and the efficiency of C++ at loop iteration, in which the mesh deformation and pixel-to-pixel mapping are implemented in C++.

The performance of wrinkle detections of linear, fixed and chaotic shapes are presented and compared with state-of-the-art methods. Then, wrinkle location and the improved wrinkle mapping are applied for expression synthesis.

A. Linear Wrinkle Detection

To test the performance of the proposed algorithm on linear shape wrinkles, a database of 37 forehead regions is extracted from the faces with different expressions in the CK+ database (D_{CK}) [31] for experiments.

To compare the effectiveness of the proposed algorithm with state-of-the-art algorithms, the hybrid Hessian filter (HHF)-based [19] and the Gabor feature ($GaborWrink$)-based [20] wrinkle detectors are chosen for comparison. For the algorithm HHF, parameters of the filter size for different scales of wrinkles and area threshold for segmenting connected wrinkle blocks are slightly adjusted for different types of wrinkles. Based on the results obtained by $GaborWrink$, the detected wrinkle line is segmented when the rotation angle is greater than π, and wrinkle lines with an overlapping ratio with the other wrinkle lines greater than 0.9 are abandoned.

The error metric, which is modified from the Jaccard Similarity Index (JSI) adopted in [19], i.e., the overlapping ratio between manual annotation and detected wrinkles, is employed in the evaluation. The index remains labeled with JSI and is defined as follows:

$$JSI = \frac{1}{n} \sum_{i=1}^{n} MIN \left(\frac{|\mathbb{M}_i \bigcap_\varepsilon \mathbb{D}_i|_M}{|\mathbb{M}_i|}, \frac{|\mathbb{M}_i \bigcap_\varepsilon \mathbb{D}_i|_D}{|\mathbb{D}_i|} \right) \tag{21}$$

where n is the number of wrinkles in the forehead region, two pixels with distance less than ε are deemed to be the same under operator $\bigcap_\varepsilon$, and $|\cdot|, |\cdot|_M$ and $|\cdot|_D$ represent the length of the considered line segment, the numbers of pixels on the manually labeled $\mathbb{M}$ and automatically detected $\mathbb{D}$ wrinkle lines, respectively. With the newly defined metric JSI, two wrinkle edges are considered matched if they are within ε pixels' bias. Thus, for algorithms such as HHF, in which the wrinkle centerlines are extracted by fitting the detected wrinkle blocks for evaluation, and a certain bias of the fitted curve away from the genuine wrinkle centerline is inevitable, the bias within ε pixels will not decrease the JSI value when evaluating with the defined metric. For the following numerical testing of forehead wrinkle detection, ε is set to six.

Table II shows the results of the proposed algorithm, together with those of HHF and GaborWrink, in which the average and standard deviation of JSI index values of the three algorithms are listed. Fig. 8 shows five example foreheads in database D_{CK} and the wrinkles detected by the three different algorithms. The JSI values of the competing algorithms for these five forehead regions are also listed in Table II.

Fig. 8 shows that $GaborWrink$ can locate a wrinkle edge that is several pixels' distance from the wrinkle centerline because the algorithm is based on the wrinkle growth along the Gabor features with maximum response. Thus, this algorithm works well for thin wrinkles such as age wrinkles, which is proven by the JSI value of $GaborWrink$ in the 4th region in Table II.

HHF locates whole wrinkle pixels directly by self-adaptive thresholding on the ridge-likeliness of the hybrid filtered image.

TABLE II
MEAN JSI VALUES AND RUNTIME OF DIFFERENT ALGORITHMS FOR WRINKLE DETECTION ON FIVE FOREHEAD REGIONS ($W_1 - W_5$) AND DIFFERENT FOREHEAD REGION DATABASES

	Mean JSI Values								
Algorithm	W_1	W_2	W_3	W_4	W_5	D_{CK}	D_{HP}	D_{LR}	$RT(s)$
HHF	0.65	0.51	**0.80**	0.38	**0.68**	**0.64** ± 0.11	**0.60**	**0.54**	**0.56**
$GaborWrink$	0.51	0.65	0.77	**0.50**	0.67	0.59 ± 0.09	0.53	0.47	3.2
The Proposed	**0.69**	**0.81**	0.68	0.41	0.63	0.61 ± 0.11	0.59	0.52	4.7

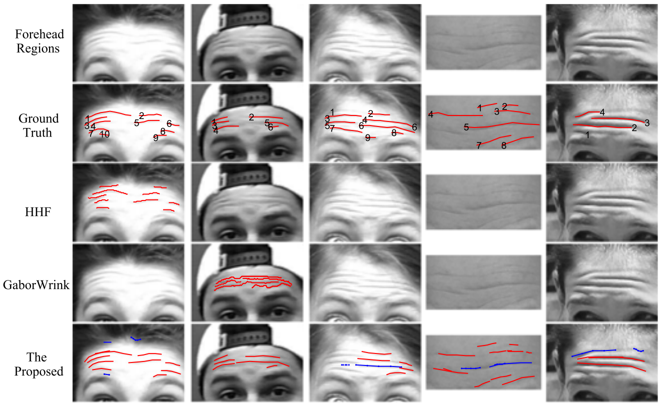

Fig. 8. Detected wrinkles on the forehead regions. The 1st and 2nd rows show the original forehead regions and the foreheads with wrinkles manually labeled. The wrinkles detected by HHF, GaborWrink, and the proposed algorithm are presented in the 3rd–5th rows, and labeled with red lines. The wrinkles detected by the proposed algorithm which are primarily different from the other two algorithms are labeled with blue stars.

As shown in Table II, HHF achieves the highest JSI values on two of the five forehead regions and the forehead wrinkle database D_{CK}. However, two parameters in HHF must be adjusted to apply at different scales of wrinkles and to segment the wrinkle blocks, although they can be made self-adaptive. With our testing, the parameter of the employed filter size must be adjusted according to different widths of forehead wrinkles, and appropriate parameter values of area threshold should be chosen to segment the wrinkle blocks when these blocks are densely distributed, as in the 4th wrinkle region.

For the first forehead region, the proposed algorithm detects some different wrinkles labeled with blue stars in Fig. 8, which are the 10th ground truth wrinkle and the wrinkle on the intersection region of forehead and hair. These different wrinkles decrease the JSI index value of the proposed algorithm. However, they imply that the proposed algorithm can detect hidden wrinkles from another aspect; hidden wrinkles labeled by blue stars are also detected in the 5th forehead region. For the 4th forehead region in Fig. 8, the proposed algorithm directly locates relatively accurate wrinkle centerlines compared with the other two algorithms, which proves the competitiveness of the proposed algorithm on locating thin forehead wrinkles. Because the edges detected by the Canny operator were used as the initial location for the following processing, there might be over-segmentation in the detected wrinkles due to broken edges caused by low image contrast.

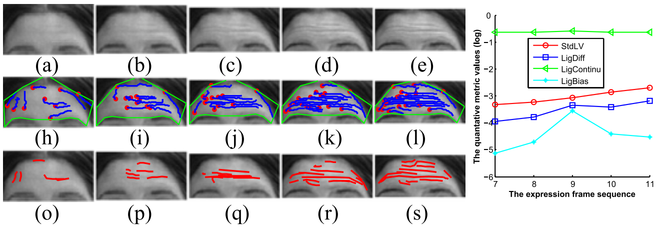

Fig. 9. Forehead wrinkle location for 7th–11th frames of temporal expression sequence and the corresponding quantitative metric values. The 1st–3rd rows show the original forehead regions, the detected wrinkle edges with DFS, and the finally detected wrinkle centerlines, respectively.

The average detection time of the three algorithms has also been recorded and included in Table II. The proposed detector requires approximately 4.7 seconds to complete the wrinkle detection process, in which the employed AAM is the main source of computational costs of our algorithm.

Fig. 9 presents the wrinkle detection results of the 7th–11th frames of the surprise expression sequence of the 87th woman [31] and the values of corresponding quantitative metrics (11)–(14). Although the wrinkles are not distinct, the proposed detector can properly locate them. Fig. 9 shows that the quantitative values are gradually increased with the wrinkling intensities, which illustrates the effectiveness of the defined metrics for forehead wrinkle discrimination.

To study the robustness of the proposed detector for wrinkles in complex conditions, ten older adults with different face colors, head poses and lighting conditions are tested. The results are presented in Fig. 10, in which faces (5) and (6) show with non-frontal head poses (denoted as D_{HP}), and faces (7) and (10) are of low resolution (denoted as D_{LR}). To assess the robustness quantitatively, the average JSI values for datasets D_{HP} and D_{LR} are listed in Table II, which shows that the proposed detector is more robust to head pose variations than to image resolution because the wrinkle edge detection largely relies on the quality of the provided face image. However, a JSI value of 0.52 is still achieved, which is competitive to the other two detectors. Moreover, Fig. 10 shows that almost all of the forehead wrinkles in the green polygon regions of the old faces are properly located, which illustrates that the proposed detector and corresponding parameter setting are robust to face colors, head poses and lighting conditions. Moreover, the proper location of the permanent wrinkles indicates that the proposed detector is applicable to not only newly appeared transient wrinkles but also changed permanent wrinkles.

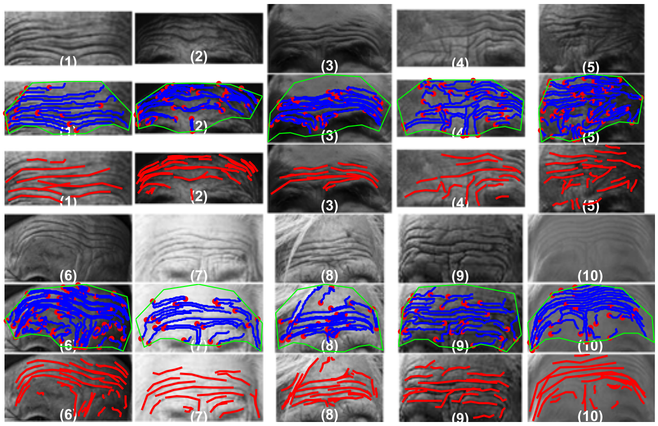

Fig. 10. Forehead wrinkle location on older adults with different conditions. The 1st and 4th rows list the original forehead regions, the 2nd and 5th rows show the detected wrinkle edges, and 3rd and 6th rows show the final detected wrinkle centerlines.

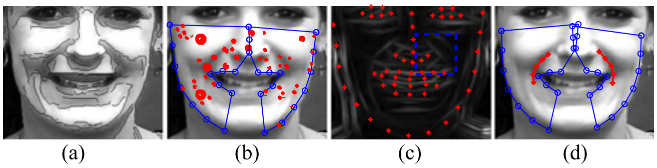

Fig. 11. Comparison of different operators for locating blurry wrinkle centerlines. (a) The Canny edge detector, (b) SIFT feature detector (the circle size denotes the feature distinctiveness), (c) the maximum response of Gabor filter, and (d) the proposed algorithm.

B. Fixed Wrinkle Detection

To illustrate that neither an edge-based nor a feature-point-based algorithm is applicable to some fixed shape wrinkles, the Canny edge detector, scale-invariant feature transform (SIFT [42], [43]) and the maximum Gabor response-based detector [20] are applied to regions with fixed shape and blurry wrinkles. The detected results are compared with those detected by the proposed algorithm in Fig. 11. Fig. 11(a) shows that the Canny detector does not detect the implied wrinkles close to the mouth tip. Although several SIFT feature points labeled with red circles are located on the blurry wrinkle regions in Fig. 11(b), a few detected feature points on non-wrinkle regions might lead to false detection. Fig. 11(c) shows that the Gabor-based detector does not detect the wrinkle edges in the region labeled by a blue dot rectangle. However, the proposed algorithm successfully locates reasonable wrinkles in the nasolabial region, which is blurry with adjacent regions because the database of nasolabial wrinkles is statistically learned.

To compare further the performance of the proposed algorithm with state-of-the-art algorithms, ASM-based [16] and AWN-based [17] transient wrinkle detectors are implemented for a comparison in which the same wrinkle initialization strategy in (6) and sub-database strategy in (10) as AAM are employed. To avoid abnormal shapes produced by ASM and AWN, when the texture of a detected wrinkle is not distinct or the detected wrinkle position and shape largely deviate from the initial wrinkle line, the initial shape obtained by the Procrustes transformation in equation (5) is chosen as the result. A database of 40 face images with nasolabial wrinkles from [31] was used for testing. Although the wrinkles of three example faces detected by different algorithms are shown in Fig. 12, the mean and standard deviation of JSI values for the entire database are shown in Table III. Considering the location accuracy of fixed shape wrinkles, each two detected wrinkle pixels are considered coincident when their distance apart is less than $\varepsilon = 6$ in (21). Fig. 12 shows that the proposed algorithm achieves more-similar wrinkles compared with those detected by the other two algorithms.

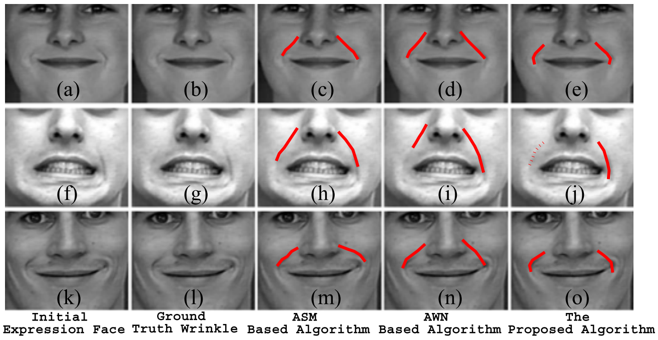

Fig. 12. Location results of three wrinkle detectors for the nasolabial wrinkles. The ground truth wrinkles and those detected by ASM, AWN, and the proposed algorithm are presented in the 2nd–5th columns, respectively. The abandoned wrinkle is denoted as a red dot line in (j).

TABLE III
JSI VALUES OF DIFFERENT ALGORITHMS FOR NASOLABIAL WRINKLE DETECTION

Algorithm	ASM	AWN	The Proposed
JSI	0.34 ± 0.08	0.39 ± 0.07	**0.47** ± 0.12

The ASM-based algorithm [16] estimates the positions of nasolabial wrinkles accompanying the locations of the other feature points on the face, primarily utilizing geometry deformation for wrinkle location. Thus, the algorithm is more prone to locate a wrinkle line with fixed shape rather than a wrinkle with maximum lighting variation. For example, a shallow wrinkle is detected in the right cheek region in Fig. 12(m), whereas the deeper wrinkle adjacent to the mouth tip cannot be located. Moreover, this algorithm is only suitable for nasolabial wrinkle detection.

The algorithm in [17] adopts AWN to replace the PCA-based texture model in ASM, predicting the positions of wrinkle feature points directly. The algorithm does not sufficiently make use of local deformation around the wrinkle line, and the initial location of the wrinkle is set to be around the edges with the first and second strongest strengths, which might not work when the wrinkle region is blurry with the adjacent regions such as in Fig. 11(a). Fig. 12(d) and 12(n) show that this algorithm locates the wrinkles with maximum lighting variation. However, the

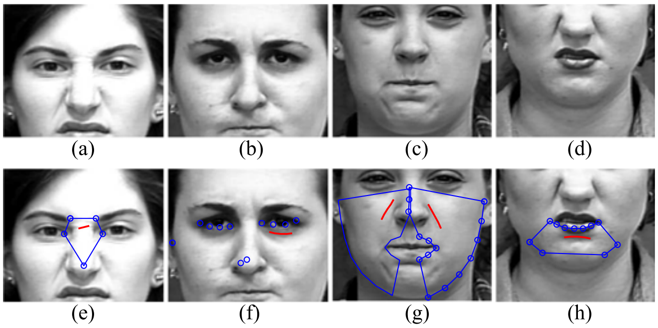

Fig. 13. Wrinkles verified by SVM. Blue polygon lines denote the outer boundaries of wrinkle structures. Red lines represent the wrinkle centerlines detected by AAM, while recognized as non-wrinkles by SVM.

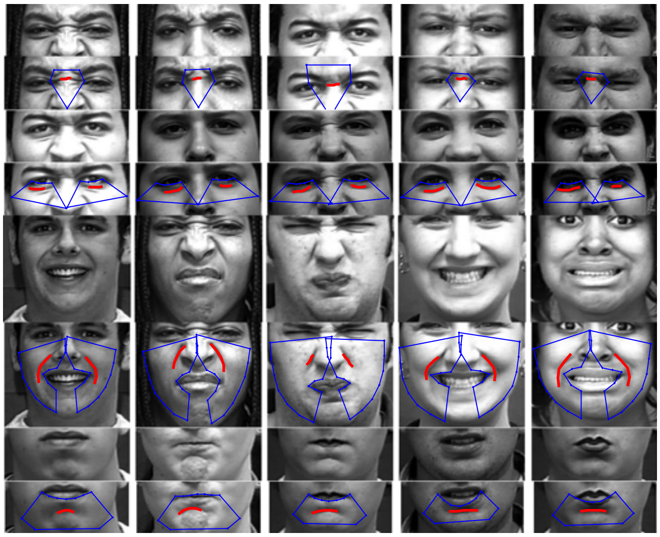

Fig. 14. Detection results of six categories of wrinkles with relatively fixed shape. Blue star lines are the outer boundaries of wrinkle structures, red lines are the detected wrinkle centerlines.

shape of the wrinkle line is not sufficiently similar to the ground truth shape in Fig. 12(b) and 12(l).

Unlike these two algorithms, the proposed algorithm not only learns from the part deformation by constructing a wrinkle structure enclosing the centerline of the wrinkle region but also utilizes the texture information learned by the intrinsic function of AAM. The competitiveness of the proposed algorithm can be proved by the observation that JSI index values of the proposed algorithm are greater than are those of the other two algorithms. Moreover, the proposed algorithm not only estimates the positions of wrinkles but also judges whether the candidate wrinkle is genuine in the generated wrinkle structure using SVM. For example, the detected wrinkle labeled with a red dot line in the left cheek of Fig. 12(j) is deduced to be non-wrinkle by SVM, which is not available in the other two algorithms. Thus, the proposed algorithm will be less prone to invalid transient wrinkles. More examples of wrongly detected wrinkles by AAM operators are presented in Fig. 13, in which SVM is further applied for verification. Fig. 14 shows the six categories of wrinkles detected by our algorithm in example fixed regions.

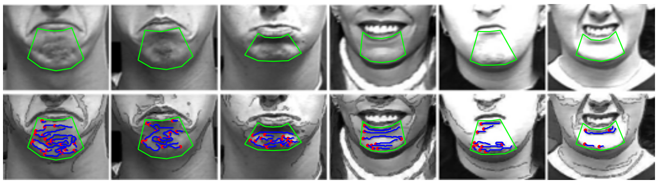

Fig. 15. First row demonstrates the chin regions labeled with green polygons. The preceding and subsequent three images of the 2nd row present the correctly detected wrinkle and non-wrinkle regions, respectively. Blue lines are continuous wrinkle edges and red dots denote the starting pixels of these edges.

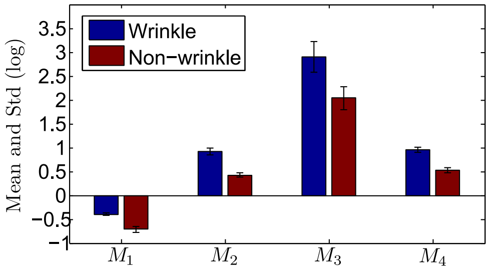

Fig. 16. Quantitative metric values for chin wrinkle and non-wrinkle regions. M_i represents the ith metric defined in Section II-D.1.

C. Chaotic Wrinkle Detection

For chaotic wrinkle detection, 49 pairs of chaotic and non-chaotic regions on the expression faces [31] are manually labeled and used for testing. This manual annotation is employed because the difference between chaotic wrinkle and non-wrinkle regions is not easily depicted sufficiently with a unique metric. Because relatively distinct boundary textures exist for chaotic wrinkles compared with fixed shape wrinkles, the Canny detector is initially employed to detect wrinkle edges for the manual annotation. Then, a committee of five volunteers decides whether the considered region contains chaotic wrinkles independently with a majority vote strategy, after learning the features of some certain chaotic wrinkle and non-wrinkle example regions. Thirty-seven pairs (74 regions) are used for the training, and the remaining 12 pairs are used for the testing. Fig. 15 shows three examples of wrinkle and non-wrinkle regions and the detection results. As shown in the figure, the chaotic wrinkle regions are filled with more wrinkles, and long wrinkles are distributed in the whole region.

We initially show the distribution of the four metrics defined from (15)–(18) for the training data in Fig. 16. The figure clearly shows that the values of the four metrics for the wrinkle regions are greater than are those of the non-wrinkle regions, which illustrates the validity of the defined metrics for discrimination.

Table IV lists the accuracies of SVM for both training and testing datasets. The training and testing data were randomly generated five times, and the average accuracies are reported. Another three features, i.e., the scale-invariant feature transform (SIFT [42], [43]), the local binary pattern (LBP [44]) and the histogram of oriented gradients (HOG [45]), are included for comparison. For extracting these features, the chin regions are resized to the same size 30 × 30.

TABLE IV
ACCURACY RATES (%) OF SVM DISCRIMINATION OF CHAOTIC WRINKLES W AND NON-WRINKLES NW WITH THE PROPOSED QUANTITATIVE METRICS AND THREE OTHER FEATURES

	Training		Testing	
Features	W	NW	W	NW
SIFT [42], [43]	71.7	85.0	46.7	51.7
LBP [44]	93.3	90.0	43.3	**95.0**
HOG [45]	91.7	86.7	81.7	70.0
The Proposed	88.3	85.0	**88.3**	86.7

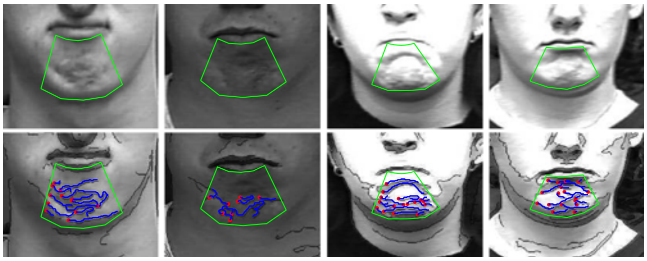

Fig. 17. Wrong classification of chaotic chin wrinkles. The left two columns represent the wrinkle regions wrongly classified as non-wrinkle, which are prelabeled with wrinkle regions for training. The right two columns represent non-wrinkle regions classified as chaotic wrinkles, which are prelabeled with non-wrinkle regions for training.

Compared with LBP and HOG, the quantitative metrics defined in this paper only represent the common characteristics of various wrinkles but neglect their specific characteristics. As a result, the proposed approach achieved lower accuracy than LBP and HOG for training samples in Table IV. From another perspective, these quantitative metrics intuitively represent the wrinkling degree of chin regions, which are less likely to be influenced by other factors such as face color and face id. Therefore, these metrics achieved relatively higher recognition rates for the testing samples. Examples of wrongly classified regions are presented in Fig. 17, in which the wrinkle features in the 2nd column are more similar to non-wrinkle regions, and the 3rd and 4th non-wrinkle features are similar to those of wrinkle features. Thus, most of these wrongly classified regions might lie in the intersection zone of discrimination space, which demonstrates another aspect of the effectiveness of the quantitative metrics.

One might observe from Fig. 14 that some minor wrinkles on the nose and near the eyes are not detected by the fixed shape wrinkle detector; a combination of forehead and chaotic wrinkle detectors has also been implemented to locate these wrinkles. In the first step, a chaotic wrinkle detector is employed to locate all of the wrinkle edge segments and determine whether there are chaotic wrinkles in the considered region. Then, a forehead wrinkle detector incorporating AAM and SVM is used to locate the accurate positions of the wrinkles. The detected wrinkle edges and centers of four example regions between two eyes or on the outer corners of eyes are presented in Fig. 18, which illustrates that the hybrid detector can compensate for the fixed wrinkle detector in regions with minor wrinkles.

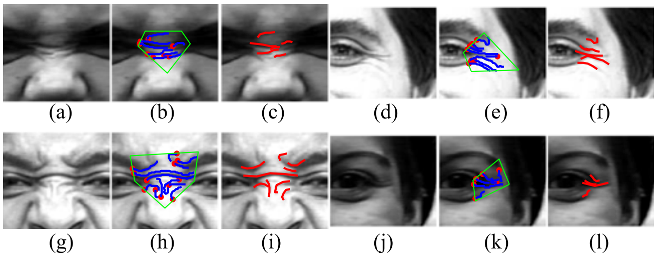

Fig. 18. Hybrid of chaotic and linear shape wrinkle detectors. The 1st and 4th columns are the original wrinkle images. The 2nd and 5th, and 3rd and 6th columns are the wrinkle edges, and wrinkle centerlines located by hybrid detector, respectively.

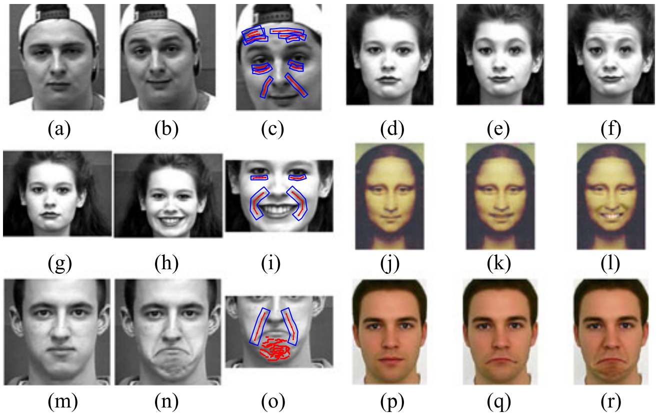

Fig. 19. Smile, laugh, and sad expressions synthesized by the proposed algorithm. From left to right, the column images list the source neutral faces F_{sn}, source expression faces F_{se}, the detected wrinkles on the source expression faces, the target neutral faces F_{tn}, the target deformed faces, and the final synthesized faces F_{te}.

D. Expression Synthesis

In expression synthesis, the wrinkles detected in the last section are mapped to the target neutral face based on lighting difference in this work. Fig. 19 shows the smile, laugh and sad expressions synthesized by our algorithm for the faces of Mona Lisa, an average man and a lady. As seen, the transient wrinkle detection algorithm locates the transient wrinkles properly on the three testing faces, and the proposed mapping form synthesizes genuine expressions.

To evaluate the synthesized results quantitatively, the similarity metric in (22) defines the correlation of the lighting differences between the source and synthesized faces and reflects the similarity of expressions F_{se} and F_{te}.

$$CorD = \frac{< \nabla L_{src}, \nabla L_{syn} >}{||\nabla L_{src}||_2 \cdot ||\nabla L_{syn}||_2} \tag{22}$$

where $< \cdot, \cdot >$ is the inner product of two vectors, ∇L_{src}, ∇L_{syn} record the lighting differences of corresponding pixels on (F_{se}, F_{sn}) and (F_{te}, F_{tn}), respectively.

We collected 10 neutral faces with corresponding anger and laugh expressions available in database [31] for testing. The 55th woman's neutral face and her anger and laugh expressions are used as the F_{sn} and F_{se}, and each of the 10 neutral faces is used as F_{tn} to synthesize the corresponding expressions F_{te}. The metric values of wrinkle mappings for ERI, IniPoisson and the

TABLE V
METRIC $CorD$ VALUES AND SUBJECTIVE SCORING OF THE EXPRESSIONS SYNTHESIZED BY DIFFERENT ALGORITHMS

	Objective ($CorD$)		Subjective	
Wrinkle Mapping	Anger	Laugh	Anger	Laugh
ERI	0.75 ± 0.10	0.74 ± 0.11	7.4 ± 0.9	7.2 ± 1.2
IniPoisson	0.78 ± 0.08	0.76 ± 0.07	8.2 ± 1.3	7.5 ± 1.0
ImpPoisson	**0.81** ± 0.09	**0.80** ± 0.10	**8.7** ± 1.2	**8.2** ± 0.9

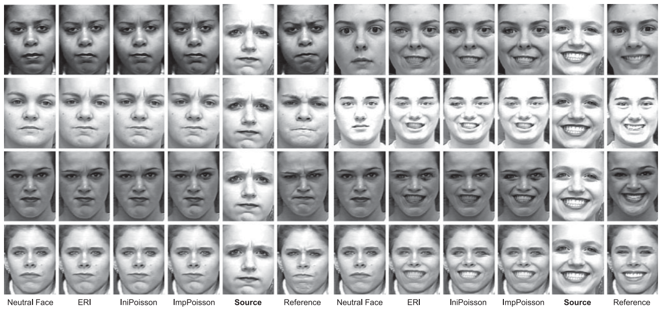

Fig. 20. Comparison of three wrinkle mapping algorithms for expression synthesis. The 5th and 11th columns are the source anger and laugh expressions F_{se}. The 1st and 7th columns are the target neutral faces F_{tn}. The other columns present the expressions synthesized with ERI, IniPoisson, the proposed ImpPoisson, source, and reference expressions.

proposed ImpPoisson are calculated according to equation (22), and their means and standard deviations are listed in Table V. One can observe from the table that our approach achieves the highest C or D; therefore, the expressions synthesized by our approach are the most similar to source expressions in terms of lighting difference. To evaluate the performance of the three wrinkle-mapping algorithms further, we asked five volunteers to assess the similarity of the synthesized expressions compared with that of the source expression. In their assessment, each of five volunteers independently scored each synthesized expression with scores from five to ten, where a higher score corresponds to a more realistic and similar expression. Averages and standard deviations of the scores for each expression are listed in Table V. Again, our approach achieves the highest scores. That is, the volunteers perceive that the expressions synthesized by our algorithm are the most similar to the true ones.

Fig. 20 shows example expressions synthesized by the three different algorithms. As seen in Fig. 20, the mapped wrinkles by ERI in the 2nd column are not as distinct as are those by ImpPoisson in the 4th column because the smoothing operator in ERI averages the lighting differences in the wrinkle region. Moreover, the mapping form based on lighting difference in the proposed algorithm handles the wrinkle boundary regions better than does the ratio-based mapping form in ERI.

IV. Discussion and Conclusion

In this work, a novel and general algorithm for transient (expression) wrinkle detection with linear, fixed and chaotic shapes is proposed. The proposed algorithm is further applied to automatic expression synthesis with an improved wrinkle mapping method. The novelties of the work are manifested on the following three aspects. First, a wrinkle database is constructed for transient wrinkle related analysis; second, a general algorithm based on wrinkle structure for detection is proposed; and third, an improved Poisson format for wrinkle mapping is proposed and applied for automatic expression synthesis.

Although competitive results are achieved by the proposed transient wrinkle detector and the improved wrinkle mapping method, there remains room for further improvement. For long wrinkles such as the wrinkles in the forehead region, the five points in the wrinkle structure can be replaced with more points for more accurate approximation of the wrinkle edge. For expression synthesis, Fig. 19 shows that the realism of the synthetic expressions depends upon the quality of eye and mouth synthesis, which will be compromised if the geometric shapes of these regions are significantly deformed or no reference texture information is available on the target neutral face. See Fig. 19(e) and 19(f) for examples. In the case of large deformation, more-refined deformation algorithms such as [13] can be considered for better preservation of the geometric features. When the reference texture information is missing after deformation, the texture information of the adjacent regions can be statistically learned with algorithms such as PCA and then further used to patch the target region to improve the realism of the synthesized expression. In addition to wrinkle mapping and geometric deformation, the effects of face alignment, variation of color and illumination shall be studied in the future. More applications of the proposed algorithm will be expanded in the future.

Acknowledgment

The authors thank the anonymous reviewers and their colleague B. George for their helpful comments and suggestions, and the codes and advice provided by C. C. Ng and Dr. N. Batool during our revision process.

References

[1] A. K. Jain and U. Park, “Facial marks: Soft biometric for face recognition,” in *Proc. IEEE Int. Conf. Image Process.*, Nov. 2009, pp. 37–40.

[2] O. G. Cula, K. J. Dana, F. P. Murphy, and B. K. Rao, “Skin texture modeling,” *Int. J. Comput. Vis.*, vol. 62, nos. 1/2, pp. 97–119, Apr. 2005.

[3] H. Ling, S. Soatto, N. Ramanathan, and D. W. Jacobs, “A study of face recognition as people age,” in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.

[4] L. Boissieux, G. Kiss, N. M. Thalmann, and P. Kalra, “Simulation of skin aging and wrinkles with cosmetics insight,” in *Computer Animation Simulation*. New York, NY, USA: Springer, pp. 15–27, 2000.

[5] Y. Wu, P. Kalra, L. Moccozet, and N. Magnenat-Thalmann, “Simulating wrinkles and skin aging,” *Vis. Comput.*, vol. 15, no. 4, pp. 183–198, 1999.

[6] N. Batool and R. Chellappa, “Detection and inpainting of facial wrinkles using texture orientation fields and Markov random field modeling,” *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3773–3788, Sep. 2014.

[7] K. Scherbaum, T. Ritschel, M. Hullin, T. Thormahlen, V. Blanz, and H. P. Seidel, “Computer-suggested facial makeup,” *Comput. Graph. Forum*, vol. 30, no. 2, pp. 485–492, 2011.

[8] M. Sofka and C. V. Stewart, “Retinal vessel centerline extraction using multiscale matched filters, confidence and edge measures,” *IEEE Trans. Med. Imag.*, vol. 25, no. 12, pp. 1531–1546, Dec. 2006.

[9] Z. Liu, Y. Shan, and Z. Zhang, “Expressive expression mapping with ratio images,” in *Proc. 28th Annu. Conf. Comput. Graph. Interactive Technol.*, 2001, pp. 271–276.

[10] Y. Zhang, W. Lin, B. Zhou, Z. Chen, and B. Sheng, , “Facial expression cloning with elastic and muscle models,” *J. Vis. Commun. Image Represent.*, vol. 25, no. 5, pp. 916–927, 2014.

[11] M. Song, Z. Dong, C. Theobalt, H. Wang, and Z. Liu, , “A generic framework for efficient 2-D and 3-D facial expression analogy,” *IEEE Trans. Multimedia*, vol. 9, no. 7, pp. 1384–1395, Nov. 2007.

[12] L. Li, F. Liu, C. Li, and G. Chen, “Realistic wrinkle generation for 3D face modeling based on automatically extracted curves and improved shape control functions,” *Comput. Graph.*, vol. 35, no. 1, pp. 175–184, 2011.

[13] Y. Huang *et al.*, “Reshaping 3D facial scans for facial appearance modeling and 3D facial expression analysis,” *Image Vis. Comput.*, vol. 30, no. 10, pp. 750–761, 2012.

[14] A. Azazi, S. L. Lutfi, I. Venkat, and F. Fernndez-Martnez, “Towards a robust affect recognition: Automatic facial expression recognition in 3D faces,” *Expert Syst. Appl.*, vol. 42, no. 6, pp. 3056–3066, 2015.

[15] Y. Zhang, and Q. Ji, “Active and dynamic information fusion for facial expression understanding from image sequences,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 699–714, May 2005.

[16] F. Tsalakanidou and S. Malassiotis, “Real-time 2D+ 3D facial action and expression recognition,” *Pattern Recog.*, vol. 43, no. 5, pp. 1763–1775, 2010.

[17] Y. Huang, Y. Li, and N. Fan, “Robust symbolic dual-view facial expression recognition with skin wrinkles: Local versus global approach,” *IEEE Trans. Multimedia*, vol. 12, no. 6, pp. 536–543, Oct. 2010.

[18] C. Ng, M. Yap, N. Costen, and B. Li, “Wrinkle detection using hessian line tracking,” *IEEE Access*, vol. 3, pp. 1079–1088, 2015.

[19] C. Ng, M. H. Yap, N. Costen, and B. Li, “Automatic wrinkle detection using hybrid Hessian filter,” in *Proc. 12th Asian Conf. Comput. Vis.*, 2015, pp. 609–622.

[20] N. Batool and R. Chellappa, “Fast detection of facial wrinkles based on Gabor features using image morphology and geometric constraints,” *Pattern Recog.*, vol. 48, no. 3, pp. 642–658, 2015.

[21] Y. H. Kwon, and N. D. V. Lobo, “Age classification from facial images,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recog.*, Jun. 1994, pp. 762–767.

[22] G. O. Cula, P. R. Bargo, A. Nkengne, and N. Kollias, “Assessing facial wrinkles: Automatic detection and quantification,” *Skin Res Technol.*, vol. 19, no. 1, pp. e243–e251, 2013.

[23] N. Batool and R. Chellappa, “Modeling and detection of wrinkles in aging human faces using marked point processes,” in *Proc. 12th Int. Conf. Comput. Vis.*, 2012, pp. 178–188.

[24] L. Yin and A. Basu, “Generating realistic facial expressions with wrinkles for model-based coding,” *Comput. Vis. Image Understand.*, vol. 84, no. 2, pp. 201–240, 2001.

[25] Y. Tian, T. Kanade, and J. F. Cohn, “Recognizing action units for facial expression analysis,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 97–115, Feb. 2001.

[26] T. Popa *et al.*, “Wrinkling captured garments using space-time data-driven deformation,” *Comput. Graph. Forum*, vol. 28, no. 2, pp. 427–435, 2009.

[27] C. Cao, D. Bradley, K. Zhou, and T. Beeler, “Real-time high-fidelity facial performance capture,” *ACM Trans. Graph.*, vol. 34, no. 4, 2015, Art. no. 46.

[28] R. Samad and H. Sawada, “Edge-based facial feature extraction using Gabor wavelet and convolution filters,” in *Proc. IAPR Conf. Mach. Vis. Appl.*, 2011, pp. 430–433.

[29] J. Canny, “A computational approach to edge detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, Nov. 1986.

[30] G. Tzimiropoulos and M. Pantic, “Optimization problems for fast AAM fitting in-the-wild,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 593–600.

[31] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, and Z. Ambadar, “The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog. Workshops*, Jun. 2010, pp. 94–101.

[32] I. Matthews and S. Baker, “Active appearance models revisited,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 135–164, 2004.

[33] H. Lee, A. Battle, R. Raina, and A. Y. Ng, “Efficient sparse coding algorithms,” in *Proc. Adv. Neural Inf. Process. Syst. Conf.*, 2006, pp. 801–808.

[34] L. Shen and L. Bai, “3D Gabor wavelets for evaluating SPM normalization algorithm,” *Med. Image Anal.*, vol. 12, no. 3, pp. 375–383, 2008.

[35] M. Jian, J. Dong, and L. Shen, “Visual-patch-attention-aware saliency detection,” *IEEE Trans. Cybern.*, vol. 45, no. 8, pp. 1575–1586, Aug. 2015.

[36] F. Liu, D. Zhang, and L. Shen, “Study on novel curvature features for 3D fingerprint recognition,” *Neurocomputing*, vol. 168, pp. 599–608, 2015.

[37] C. Chang and C. Lin, “LIBSVM: A library for support vector machines,” *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27, 2011.

[38] K. Li, Q. Dai, R. Wang, Y. Liu, and F. Xu, , “A data-driven approach for facial expression retargeting in video,” *IEEE Trans. Multimedia*, vol. 16, no. 2, pp. 299–310, Feb. 2014.

[39] P. Perez, M. Gangnet, and A. Blake, “Poisson image editing,” *ACM Trans. Graph.*, vol. 22, no. 3, pp. 313–318, 2003.

[40] S. Milborrow and F. Nicolls, “Active shape models with SIFT descriptors and MARS,” *VISAPP*, vol. 2, pp. 380–387, 2014.

[41] T. Beier and S. Neely, “Feature-based image metamorphosis,” in *Proc.19th Annu. Conf. SIGGRAPH Comput. Graph. Interactive Techn.*, 1992, pp. 35–42.

[42] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[43] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, “Visual categorization with bags of keypoints,” in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 1–22.

[44] T. Ojala, M. Pietikainen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[45] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proc. Comput. Vis. Pattern Recog.*, 2005, pp. 886–893.

Weicheng Xie received the B.S. degree in statistics from Central China Normal University, Wuhan, China, in 2008, and the M.S. degree in probability and mathematical statistics and Ph.D. degree in computational mathematics from Wuhan University, Wuhan, China, in 2010 and 2013, respectively.

He was previously a Visiting Research Fellow with the School of Computer Science, University of Nottingham, Nottingham, U.K. He is currently a Postdoctoral Researcher with the Shenzhen Key Laboratory of Spatial Smart Sensing and Services, School of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His current research interests include 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, Shenzhen, China. His current research interests include Gabor wavelets, face/palmprint recognition, medical image processing, and image classification.

Jianmin Jiang received the Ph.D. degree from the 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. He has authored or coauthored more than 400 refereed research papers. 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 artificial intelligence applications in digital media processing, retrieval, and analysis.

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.