Video Face Editing Using Temporal-Spatial-Smooth Warping

XIAOYAN LI, TONGLIANG LIU, JIANKANG DENG, and DACHENG TAO,
University of Technology, Sydney

Editing faces in videos is a popular yet challenging task in computer vision and graphics that encompasses various applications, including facial attractiveness enhancement, makeup transfer, face replacement, and expression manipulation. Directly applying the existing warping methods to video face editing has the major problem of temporal incoherence in the synthesized videos, which cannot be addressed by simply employing face tracking techniques or manual interventions, as it is difficult to eliminate the subtly temporal incoherence of the facial feature point localizations in a video sequence. In this article, we propose a temporal-spatial-smooth warping (TSSW) method to achieve a high temporal coherence for video face editing. TSSW is based on two observations: (1) the control lattices are critical for generating warping surfaces and achieving the temporal coherence between consecutive video frames, and (2) the temporal coherence and spatial smoothness of the control lattices can be simultaneously and effectively preserved. Based upon these observations, we impose the temporal coherence constraint on the control lattices on two consecutive frames, as well as the spatial smoothness constraint on the control lattice on the current frame. TSSW calculates the control lattice (in either the horizontal or vertical direction) by updating the control lattice (in the corresponding direction) on its preceding frame, i.e., minimizing a novel energy function that unifies a data-driven term, a smoothness term, and feature point constraints. The contributions of this article are twofold: (1) we develop TSSW, which is robust to the subtly temporal incoherence of the facial feature point localizations and is effective to preserve the temporal coherence and spatial smoothness of the control lattices for editing faces in videos, and (2) we present a new unified video face editing framework that is capable for improving the performances of facial attractiveness enhancement, makeup transfer, face replacement, and expression manipulation.

Categories and Subject Descriptors: I.2.10 [**Artifical Intelligence**]: Vision and Scene Understanding—*Video analysis*; I.4.9 [**Image Processing and Computer Vision**]: Applications

General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Video face editing, warping, spatial smoothness, temporal coherence

ACM Reference Format:
Xiaoyan Li, Tongliang Liu, Jiankang Deng, and Dacheng Tao. 2016. Video face editing using temporal-spatial-smooth warping. ACM Trans. Intell. Syst. Technol. 7, 3, Article 32 (February 2016), 28 pages.
DOI: http://dx.doi.org/10.1145/2819000

1. INTRODUCTION

The dramatic growth in the availability of online videos has resulted in a great demand for editing the faces that appear in videos. Recently, many data modeling technologies have been successfully proposed for image-based face representation. Subspace

This work is supported in part by the Australian Research Council Projects under grants FT-130101457 and LP-140100569.
Authors' address: X. Li, T. Liu, J. Deng, and D. Tao, Centre for Quantum Computation & Intelligent Systems and the Faculty of Engineering & Information Technology, University of Technology, Sydney, Ultimo, NSW 2007, Australia; emails: {xiaoyanli.swjtu, tliang.liu, jiankangdeng}@gmail.com; dacheng.tao@uts.edu.au.

DOI: http://dx.doi.org/10.1145/2819000

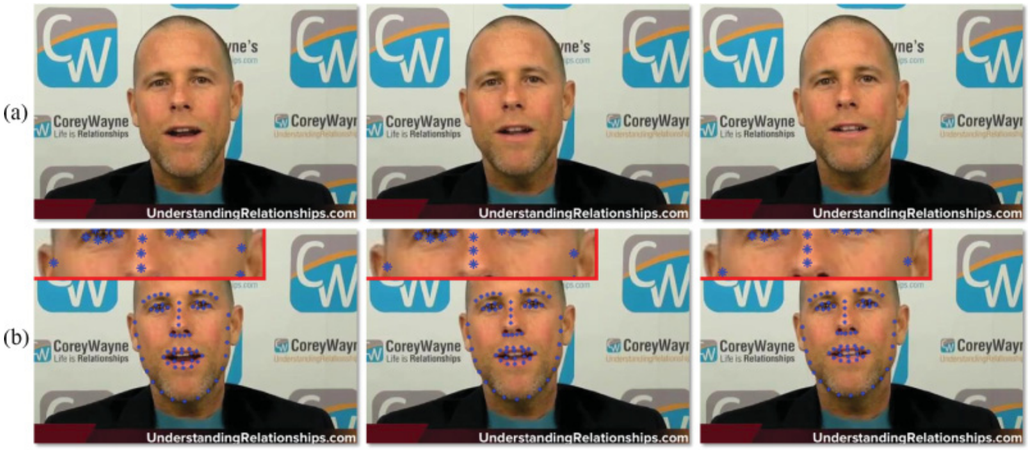


Fig. 1. Temporal incoherence of the facial feature point localizations. (a) Three consecutive frames of an input video. (b) The 66 facial feature point localizations are obtained by using Xiong and de la Torre [2013], which have subtly temporal incoherence of the facial outline points on the three consecutive frames.

learning [Xu et al. 2015], principal component [Tao et al. 2015b], reweighting [Liu and Tao 2015], and optimization [Han et al. 2015] have produced promising results in computer vision and graphics. In practice, the four most required video face editing applications are (1) enhancing facial "attractiveness," (2) transferring makeup from one face to another face, (3) replacing the face in the target video with a source face, and (4) manipulating facial expressions. However, the existing face editing methods (e.g., Leyvand et al. [2008], Tong et al. [2007], Scherbaum et al. [2011], Kemelmacher-Shlizerman et al. [2010], and Yang et al. [2011]) share the major problem of temporal incoherence when applied to videos.

The warping generation is important to ensure the temporal coherence in the synthesized video for video editing. However, it is difficult to achieve temporal coherence by directly employing the existing image- or video-based warping approaches (e.g., Lee et al. [1996], Zhang et al. [2015], and Lin et al. [2013]) to video face editing tasks. There always exist subtly temporal incoherence of facial feature point localizations even after employing the existing facial point tracking techniques (e.g., Ong and Bowden [2011], Korshunov and Ooi [2011], and Xiong and de la Torre [2013]) or manual interventions for refining the motions of the facial feature points. Figure 1 shows subtly temporal incoherence of the facial outline points on three consecutive frames. When the temporal incoherence of the facial feature point localizations exists, the most commonly used multilevel B-spline approximation (MBA) method [Lee et al. 1997] with a coarse-to-fine hierarchy of the control lattices at different levels (illustrated in Figure 2(a)), cannot achieve temporally coherent results for video face editing tasks. Specifically, if the localizations of the same feature point on two consecutive frames are different, (1) the 16 control points[1] for different localizations will be different, as shown in Figure 2(b) and (c); (2) the distances between the facial feature point localizations and the positions of the upper-left control points, e.g., d_1 and d_2 in Figure 2(b) and (c), will be different; and (3) the displacements between the source and target localizations of the facial feature points, e.g., d_3 and d_4 in Figure 2(d), will be different. These differences will cause the temporal incoherence of the control lattices on two consecutive frames, which limits

[1] Given the localization of a feature point, the 16 neighbor control points in a 4 × 4 square grid are extracted by starting from its upper-left control points.

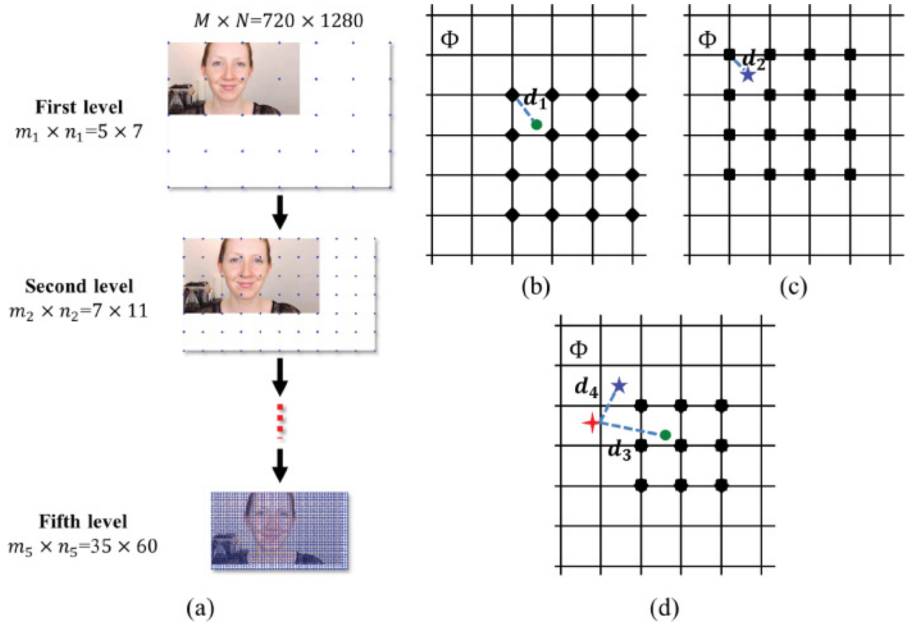


Fig. 2. Control lattices and positional relationship between feature and control points. (a) A coarse-to-fine hierarchy of the control lattices at different levels. Given an image of size $M \times N$, the size of the h-level control lattice will be $m_h \times n_h = \lceil (M \times N)/(\min\{M, N\}/2^h) \rceil + 3$. (b) Let Φ be the control lattice. A feature point on the previous frame is marked by bullet symbol. According to the localization of the feature point, the 16 control points are extracted and marked by diamond symbol. The distance between the feature point and the upper-left control point, denoted by d_1. (c) The feature point on the current frame is marked by star symbol and then used to select the neighbor control points (square symbol). d_2 represents the distance between the feature point and its upper-left control point. (d) After editing, if the target feature points on the two consecutive frames are the same (marked by cross symbol), the displacements between the source and target feature points are denoted as d_3 and d_4. There are nine overlapped control points.

the applicability of MBA to videos. Therefore, the focus of this article is to design an effective approach to improve the performance of temporal coherence in the synthesized videos for video face editing tasks when subtly temporal incoherence of facial feature point localizations exists.

This article presents the temporal-spatial-smooth warping (TSSW) method for video face editing based on the following observations: (1) the control lattices are critical for generating warping surfaces and achieving the temporal coherence between consecutive frames, and (2) the temporal coherence and spatial smoothness need to be simultaneously and effectively preserved. Therefore, we introduce spatial and temporal smoothness constraints on the control lattices rather than directly on the warping surfaces. TSSW obtains the control lattice (in either the horizontal or vertical direction) on the current frame by updating the control lattice (in the corresponding direction) on its preceding frame (using MBA to estimate the control lattices in the horizontal and vertical directions on the first frame), i.e., optimizing a new energy function that contains three terms: a data-driven term, a smoothness term, and feature point constraints. The data-driven term measures the differences of the control lattices on two consecutive frames, which is designed to preserve the temporal coherence of the control lattices. The smoothness term is defined by the summations of the gradients of the control lattice in the horizontal and vertical directions, which is designed to guarantee the spatial smoothness of the control lattice on the current frame. The feature point constraints enhance the one-to-one mapping from the source feature points to the target feature points. The control lattices on the video frames (except on the first frame) are obtained by minimizing the proposed energy function to achieve highly temporally coherent results in a range of video face editing tasks. Experimental results on four video face editing applications (i.e., facial attractiveness enhancement, makeup transfer, face replacement, and expression manipulation) demonstrate that the proposed approach for video face editing achieve a higher temporal coherence and spatial smoothness than the existing warping methods. More results are available online on the project Web page.[2]

In summary, this article makes two major contributions:

(1) We develop TSSW, which is robust to the subtly temporal incoherence of facial feature point localizations and is effective to preserve the temporal coherence and spatial smoothness of the control lattices for editing faces in videos.
(2) We present a new unified video face editing framework that is capable for improving the performances of facial attractiveness enhancement, makeup transfer, face replacement, and expression manipulations.

The remainder of this article is organized as follows. Section 2 reviews the related work. In Section 3, we introduce the three main steps of video face editing and compare the differences in the integrated framework and the existing editing works. Section 4 presents the proposed TSSW approach. Section 5 shows the experimental results of the four face editing tasks by using different algorithms. Section 6 discusses the conclusions, limitations, and future work.

2. RELATED WORK

The proposed TSSW method can be applied to and improve the performance of temporal coherence for various video face editing tasks, which is related to the existing works in the following fields:

—*Image-based warping* is used to retain 2D geometric transformations between feature point pairs.
  (i) Traditional image-based warping methods include the radial basis function (RBF)-based method [Arad et al. 1994] and the thin plate spline (TPS)-based method [Donato and Belongie 2002]. Multilevel free-form deformation (MFFD) [Lee et al. 1996] and its improved version by applying a coarse-to-fine hierarchy of B-spline refinement to the control lattices [Lee et al. 1997] were also presented for image metamorphosis.
  (ii) Moving least squares (MLS) [Schaefer et al. 2006] was designed to make the image deformation as rigid as possible. Moving regularized least squares (MRLS) [Ma et al. 2013] was proposed to solve the problem of nonrigid image deformation.

—*Image-based face editing* has gained extensive attention in recent years.
  (i) Leyvand et al. [2008] presented a method to enhance the facial attractiveness by exploring a set of training faces. A framework proposed by Guo and Sim [2009] was designed to mock physical makeup by creating the makeup upon a face through a template image.
  (ii) A system [Bitouk et al. 2008] was introduced to swap faces by finding candidates similar to the appearance and pose of the input face from a dataset of faces. Yang et al. [2011] replaced two different facial expressions of the same person by using the optical flow technique [DeCarlo and Metaxas 2000].

—*Shape registration* aims to construct the optimal transformation for shapes.

[2] https://sites.google.com/site/tsswmethod/.

   (i) An efficient registration method based on a cubic deformation model [Taron et al. 2009] was designed to recover the one-to-one correspondence between source and target shapes. A dual decomposition approach [Torresani et al. 2013] was proposed to establish the correspondences between sparse image feature points.
   (ii) Huang et al. [2006] and Munim et al. [2013] estimated the global and local transformation parameters for shape registration by using an implicit distance function and an energy optimization.

—*3D-based face editing approaches* try to edit faces by creating 3D facial models from a set of 2D images.
   (i) A 3D-aware appearance optimization technique is applied to the tasks of face morphing [Yang et al. 2012b] and face component transfer [Yang et al. 2011]. The 3D morphable face models are used to enhance the symmetry and proportion of face geometry [Liao et al. 2012] and suggest the best-fit makeup for an input human face [Scherbaum et al. 2011].
   (ii) Dale et al. [2011] and Yang et al. [2012a] extended Vlasic's 3D tensor approach [Vlasic et al. 2005] to edit the facial components of one or two identities in videos. A FaceWarehouse database [Chen et al. 2014] consisting of a set of 3D facial expressions was constructed for face editing applications.

—*Video-based warping* is often applied to video retargeting and video stabilization.
   (i) Wang et al. [2010] and Lin et al. [2013] exploited several spatial deformation (e.g., nonuniform global mesh warping) and temporal coherence constraints to preserve visually salient content (e.g., foreground objects) for video retargeting.
   (ii) Video stabilization techniques have been developed to smooth shaky camera motions, such as the structure from motion (SFM) model [Liu et al. 2009] and the spatial-temporal optimization method [Wang et al. 2013].

To sum up, the image-based warping and image-based face editing methods are mostly single-frame based and do not consider the challenging problem of the temporal coherence in video editing. The shape registration approaches strongly focus on the feature point constraints between the source and target points, which are very sensitive to the temporal incoherence of the feature point localizations. The existing 3D-based face editing approaches usually require intensive manual interventions for adjusting the facial feature point localizations, which is time consuming for various videos. The existing video-based warping methods are mostly performed for video retargeting and video stabilization, to achieve temporal coherence by smoothing the motions of objects or a shaky camera. However, these methods are not suitable to video face editing because of the high salient property of human faces. The proposed TSSW method achieves temporally coherent results by effectively preserving the temporal coherence and spatial smoothness of the control lattices across the video frames, without the costly 3D information or manual interventions.

3. FACE EDITING IN VIDEO

Figure 3 shows a video face editing framework that integrates the four video face editing tasks by exploiting the proposed TSSW method. The framework mainly contains three main steps: facial feature point localizations, facial component editing, and warping generation. First, the facial feature points on each frame of an input video are detected and tracked by employing one of the face tracking techniques (e.g., Xiong and de la Torre [2013], Tao et al. [2015a], Liu et al. [2015], and Shi et al. [2015]). Second, the localizations of the facial feature points on each video frame are edited and modified by using one of the desired video face editing tasks, e.g., facial attractiveness enhancement. Third, according to the source and target localizations of the facial feature points and the control lattice (in either the horizontal or vertical direction) on the previous

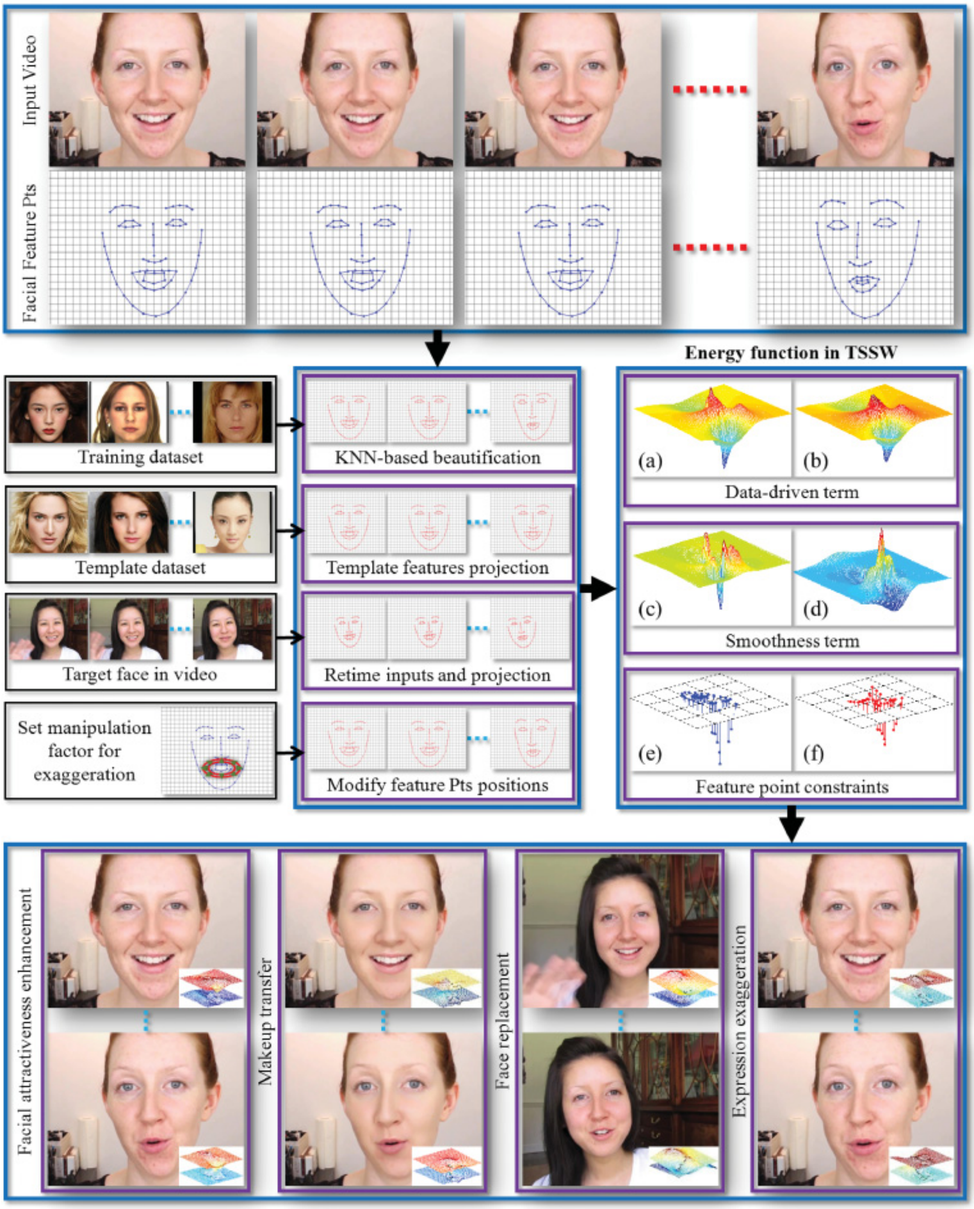


Fig. 3. A video face editing framework by exploiting the proposed TSSW method. The "Energy function in TSSW" part visually illustrates an example of the warping surfaces obtained by Equation (11). The warping surfaces are used to compute the data-driven term (Equation (8)), the smoothness term (Equation (9)), and the feature point constraints (Equation (10)). (a) and (b) are the warping surfaces (in the horizontal direction) on the current frame and its preceding frame, i.e., $f(\Psi_t^1)$ and $f(\Psi_{t-1}^1)$, respectively. (c) and (d) are the gradients of the warping surface in the horizontal and vertical directions, i.e., $f(\partial\Psi_t^1/\partial x)$ and $f(\partial\Psi_t^1/\partial y)$, respectively. (e) and (f) are the horizontal and vertical warping surface values on the target point localizations, i.e., $\{f_k(\Psi_t^1) \mid (p_k^1, p_k^2) \in P_t\}$ and $\{f_k(\Psi_t^2) \mid (p_k^1, p_k^2) \in P_t\}$, respectively. The two warping surfaces, i.e., $f(\Psi_t^1)$ and $f(\Psi_t^2)$ are used to generate the synthesized frames (shown in the two bottom rows).

frame, the control lattice (in the corresponding direction) on the current frame is obtained by minimizing the proposed energy function that unifies a data-driven term, a smoothness term, and feature point constraints.

3.1. Facial Feature Point Localizations

Video face editing is usually based on facial feature point localizations, which has achieved remarkable improvements on standard benchmarks [Çeliktutan et al. 2013]. Recently, the regression-based model [Dollar et al. 2010] has attracted intensive attention, which learns a regression function to directly map the facial appearance features of a face image to the facial shape.

There are two popular ways to learn such a regression function, one of which is based on the deep network. Sun et al. [2013] introduced the deep convolutional neural network (DCNN) in the regression framework to locate five fiducially facial landmarks. Zhou et al. [2013] proposed a four-level convolutional network cascade, in which each level is trained to locally refine the outputs of the previous network levels. Zhang et al. [2014] proposed coarse-to-fine autoencoder networks (CFANs) for real-time face alignment. The cascade regression model is another popular approach, which depends on the shape-indexed features and the stacked regressors. The explicit shape regression (ESR) [Cao et al. 2012] combined a two-level boosted regression, the shape-indexed features and a correlation-based feature selection. The supervised descent method (SDM) [Xiong and de la Torre 2013] exploited the cascade regression model on the scale-invariant feature transform (SIFT) features.

Traditional face alignment methods usually assume that a reliable initialization is obtained from the face detector. However, this assumption is too strict to be held to many real applications, which requires an automatic face alignment. Challenging factors for facial feature point localizations such as pose, illumination, expression, and occlusion also degrade the performance of the face detector [Mathias et al. 2014]. Recent face detection algorithms are usually measured by the 50% overlap criterion [Jain and Learned-Miller 2010], which may generate drift face detections and are not sufficiently accurate for the initialization of facial feature point localizations. Despite the importance of the initialization in real applications, few works are proposed to handle the initialization problem.

To improve the robustness of the initialization of face detector, we integrate the stacked autocoders (SAs) [Vincent et al. 2010] and SDM. Since the shape variance in the first-level regression is very large and the capacity of SAs is larger than that of least squares, we replace least squares by the SAs to improve the shape convergence. Given the relatively accurate shape obtained by SAs, we exploit SDM to refine the alignment results. Figure 4 shows the flow chart of the improved SDM (ISDM) method and compares the initial shapes obtained by ISDM and SDM. ISDM is more accurate and robust than SDM regarding the exaggerate shape variations.

3.2. Facial Component Editing

In this article, we demonstrate four scenarios: facial attractiveness enhancement, makeup transfer, face replacement, and expression manipulation. Since the integrated framework of the four video face editing tasks is based on the facial feature point localizations, the goal of editing facial components is therefore to obtain the target localizations of the facial feature points.

Figure 5 shows a brief description of facial component editing in the four video editing tasks. The source and target localizations of the facial feature points $\{Q, P\}$ on each frame are obtained by one of the desired facial component editing tasks (e.g., Leyvand et al. [2008], Guo and Sim [2009], Dale et al. [2011], and Yang et al. [2012a]). The

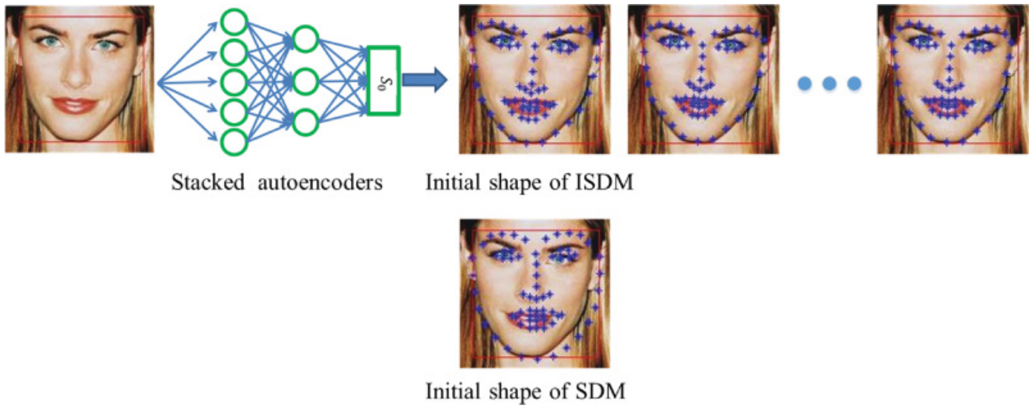


Fig. 4. The flow chart of improved SDM (ISDM) and the initial shapes obtained by ISDM and SDM.

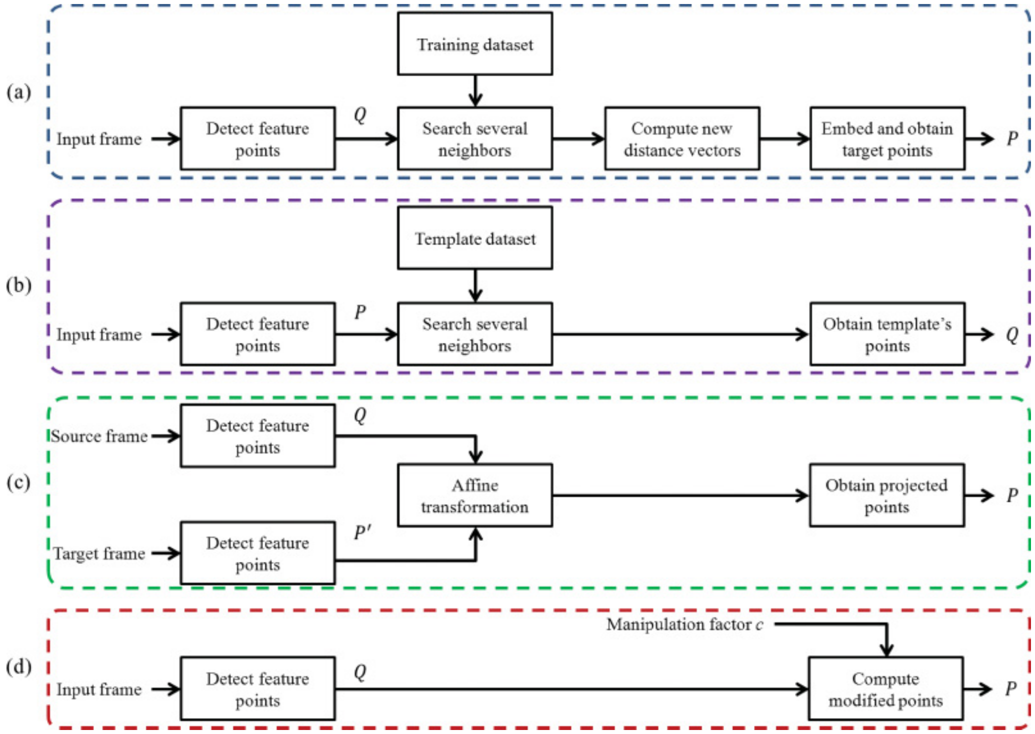


Fig. 5. Facial component editing in the four video face editing tasks. The feature point pair $\{Q, P\}$ of each frame is obtained by one of the facial component editing schemes. (a) Facial attractiveness enhancement. (b) Makeup transfer. Note that the feature points on the select template face are considered as the original feature points. (c) Face replacement. After affine transformation, the feature points on the source frame are projected on the target frame to obtain the target positions. (d) Expression manipulation. The target point positions are computed by the manipulation factor with the source points.

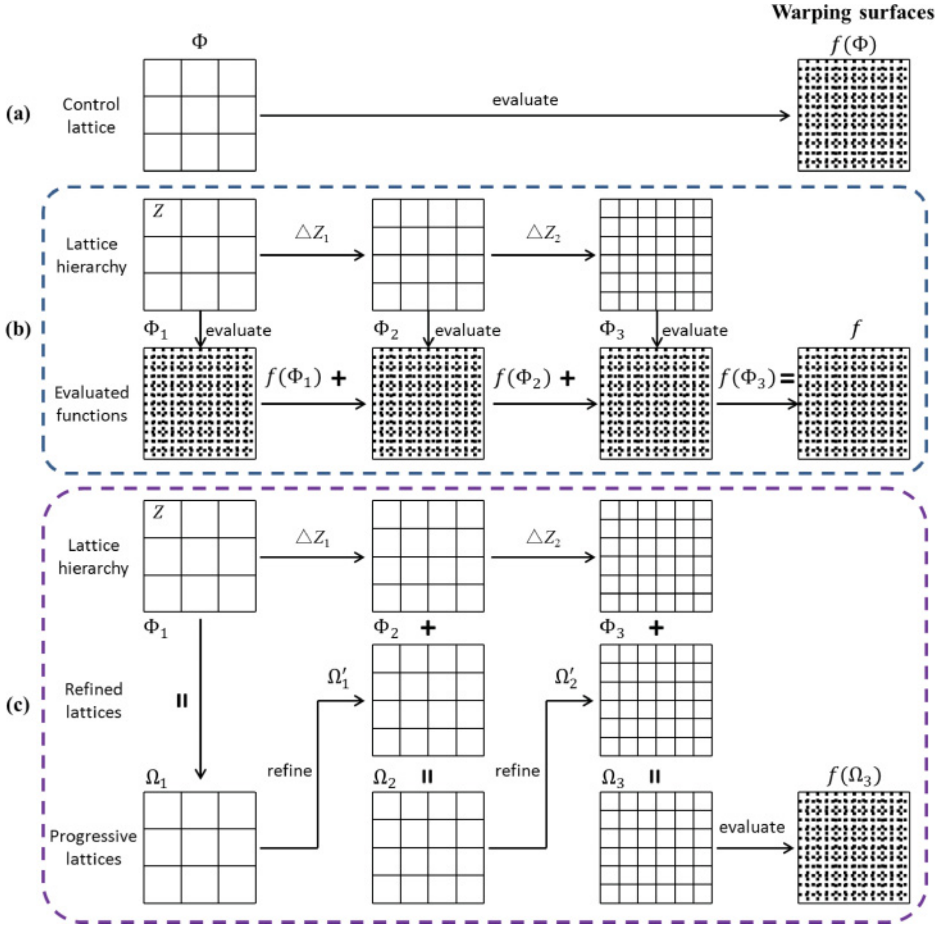


Fig. 6. Warping surfaces obtained by different algorithms. (a) Single-level B-spline approximation. (b) MFFD algorithm. For each level, the evaluated function is computed by the corresponding control lattice. The warping surface is the sum of these evaluated functions. Z represents the displacement from the original feature set Q to the target feature set P, i.e., $Z = Q - P$. $\triangle Z_1$ and $\triangle Z_2$ are the residual displacements at the first- and second-level frames. (c) MBA algorithm with a hierarchical refinement. The warping surface is evaluated by the final control lattice.

technical descriptions of the four video editing tasks are described on the project Web page [Li et al. 2014].

3.3. Warping Generation

Figure 6 shows the warping surfaces obtained by three methods, i.e., single-level B-spline approximation, MFFD, and MBA. The single-level B-spline approximation method has a low estimation accuracy. MFFD computes the evaluated function by using the control lattice at each level. However, it is very time consuming for the computation of the warping surface values on all image points. MBA is a more effective method to generate the warping surfaces by a hierarchical refinement than MFFD. However, it is very difficult to extend MBA for video face editing tasks because temporal incoherence of the facial feature point localizations seriously affects its performance, as discussed in Section 1.

Table I. Comparisons of the Proposed Approach and the Existing Works

Task		Facial Feature Point Localizations		Facial Component Editing	Warping Generation
		Number of Points	Automatic or Not		
1	Leyvand et al. [2008]	84	no	Distance-based similarity	MFFD
	Ours	66	yes	Kernel-based similarity	TSSW
2	Guo and Sim [2009]	83	no	Layer decomposition	TPS
	Ours	66	yes	Layer decomposition	TSSW
3	Dale et al. [2011]	24	no	3D-based replacement	Mesh warping
	Ours	66	yes	2D-based replacement	TSSW
4	Yang et al. [2012a]	66	yes	3D-based manipulation	MLS
	Ours	66	yes	2D-based manipulation	TSSW

1, facial attractiveness enhancement; 2, makeup transfer; 3, face replacement; 4, expression manipulation.

TSSW significantly improves MBA by exploiting the temporal information of the control lattices between two consecutive frames, as well as the spatial smoothness within the current frame, which is effectively applicable to various video face editing tasks. To obtain the control lattices in the horizontal and vertical directions, we apply MBA to the first frame, as summarized in Algorithm 1. For each of the subsequent frames, the control lattice in the horizontal or vertical direction can be obtained by solving the energy function minimization problem, as described in Section 4.2. Once obtained the control lattice, the corresponding warping surface is computed by Equation (11). The horizontal and vertical warping surfaces are then used to generate the warped frame by Equation (18).

3.4. Comparisons

The existing related face editing works are (1) Leyvand et al. [2008] for facial attractiveness enhancement, (2) Guo and Sim [2009] for makeup transfer, (3) Dale et al. [2011] for face replacement, and (4) Yang et al. [2012a] for expression manipulation. Table I shows the comparisons of the integrated framework and the existing works with respect to the steps of facial feature point localizations, facial component editing, and warping generation.

From Table I, we can see that the warping generation is the major difference between the proposed approach and existing works. Due to the lack of the consideration of temporal coherence in the generation of warping surfaces (or even the control lattice generation in MBA), the existing warping methods cannot produce natural-looking results when applied to videos, especially for video face editing tasks. The proposed TSSW method simultaneously and effectively preserves the temporal coherence and spatial smoothness of the control lattices on the video frames and improves the performances of the four video face editing tasks.

4. TEMPORAL-SPATIAL-SMOOTH WARPING

Among the three main steps in the video face editing framework, the generation of warping surfaces is critical to achieve temporal coherence in the synthesized videos. The proposed TSSW method significantly advances MBA, where the warping surface (in either the horizontal or vertical direction) is evaluated by the control lattice (in the corresponding direction) at the finest level and the B-spline basis functions. Due to the clarity of the B-spline basis functions [Anjyo et al. 2014], the warping generation is limited to the control lattice generation, i.e., Equation (11). The goal of this work is to calculate the control lattice (in either the horizontal or vertical direction) on the current frame by updating the control lattice (in the corresponding direction) on its preceding frame

Table II. Important Notations Used in Section 4

Notation	Description	Notation	Description
$M \times N$	Size of a video frame	Ω_h	Progressive lattice at h-th level
$m_h \times n_h$	Size of control lattice at h-th level	Ω'_h	Refined lattice at h-th level
s_h	Scaling factor for h-th level	θ_{ij}	Control point value in Ω'_h
H	Number of levels in MBA	l	Index for direction
K	Number of facial feature points	I	$m_H n_H \times m_H n_H$ identity matrix
$\mathbb{T}$	Number of video frames	W_k	Weight matrix for k-th image point
$Q = \{(q_k^1, q_k^2)\}_{k=1}^K$	Source facial feature point localizations	L	Laplacian matrix
$P = \{(p_k^1, p_k^2)\}_{k=1}^K$	Target facial feature point localizations	X_{in}^t	The t-th input frame
d_k	Displacement for k-th feature point	X_{warp}^t	The t-th warped frame
Φ_h^l	Control lattice at h-th level in MBA	Ψ_t^l	Control lattice on t-th frame in TSSW
ϕ_{ij}	Control point value in Φ_h^l	$f(\Psi_t^l)$	Warping surface
$\{B_i(\cdot)\}_{i=0}^3$	B-spline basis functions	α, β	Regularization parameters

for preserving the temporal coherence of the control lattice on two consecutive frames. For convenience, Table II lists the important notations of TSSW used in this section.

Let $\mathbb{T}$ be the number of frames in an input video. Applying the proposed ISDM method, the facial feature points on an input frame are obtained and denoted as $Q = \{(q_k^1, q_k^2)\}_{k=1}^K$. After applying one of the four facial component editing engines (described in Section 3.2), the target localizations of the facial feature points are obtained and denoted as $P = \{(p_k^1, p_k^2)\}_{k=1}^K$. For the t-th frame ($1 \leq t \leq \mathbb{T}$), the source and target localizations of the K facial feature points are unified as $\{Q_t, P_t\}$.

4.1. MBA for Control Lattice Initialization

The proposed TSSW method needs the control lattices in the horizontal and vertical directions on the first frame. To achieve it, we directly apply the MBA method [Lee et al. 1997] to obtain the two control lattices, as summarized in Algorithm 1.

4.1.1. B-Spline Basis Functions. Recently, a scattered data interpolation technique based on the B-spline basis functions was applied in computer graphics and vision [Anjyo et al. 2014]. The B-spline basis functions are defined as follows [Lee et al. 1997]:

$$B_i(u) = a_i [u^3\ u^2\ u^1\ u^0]^T, \quad i = 0, 1, 2, 3, \tag{1}$$

where $0 \leq u < 1$. $\{a_i\}_{i=0}^3$ are the basis vectors, i.e., $a_0 = [-1\ 3\ -3\ 1]/6$, $a_1 = [3\ -6\ 0\ 4]/6$, $a_2 = [-3\ 3\ 3\ 1]/6$, and $a_3 = [1\ 0\ 0\ 0]/6$. The symbol T represents the transpose operation. The B-spline basis functions are used to weigh the contribution of the 16 neighbor control points. As illustrated in Figure 2(b) and (c), the distances between the facial feature point localizations and the positions of the upper-left control points, e.g., d_1 and d_2, are used to compute the values of the B-spline basis functions by using Equation (1).

4.1.2. Feature Point on Control Lattice. Given an image of size $M \times N$, the control lattice at the h-level of a coarse-to-fine hierarchy (illustrated in Figure 2(a)) is of size $m_h \times n_h$. The target localizations of the facial feature points are embedded in the h-level control lattice and scaled by $p_{h,k}^l = s_h \cdot (p_k^l - 1) + 1$, where $s_h = \min\{m_h - 3, n_h - 3\}/\min\{M, N\}$. The control point value ϕ_{ij} of the h-level control lattice Φ_h^l is computed by Lee et al. [1997]:

```
ALGORITHM 1: MBA for Control Lattice Initialization
Input: Facial feature point pair $\{Q_1, P_1\}$ and the total number of levels $H$.
Output: Estimated control lattices $\Psi_1^1$ and $\Phi_1^2$.
Estimation:
Compute the first-level control lattice $\Phi_1^1$ and $\Phi_1^2$ using Equations (2) and (5);
for $l = 1, 2$ do
    Set the progressive lattice $\Omega_1 = \Phi_1^l$;
    for $h = 2, 3, \ldots, H$ do
        Compute the control lattice $\Phi_h^l$ using Equations (2) and (5);
        Compute the refined lattice $\Omega'_{h-1}$ by the progressive lattice $\Omega_{h-1}$ using Equation (6);
        Update the progressive lattice by $\Omega_h = \Phi_h^l + \Omega'_{h-1}$;
    end
    Obtain the control lattice $\Psi_1^l = \Omega_H$;
end
```

$$\phi_{ij} = \frac{\hat{\omega}_{ij} d_k}{\sum_{s=0}^{3} \sum_{r=0}^{3} \hat{\omega}_{sr}^2}, \tag{2}$$

where $\hat{\omega}_{ij} = B_i(u_k)B_j(v_k)$, $i_k = \lfloor p_{h,k}^1 \rfloor$, $j_k = \lfloor p_{h,k}^2 \rfloor$, $u_k = p_{h,k}^1 - i_k$, and $v_k = p_{h,k}^2 - j_k$. $\lfloor \cdot \rfloor$ is the round down operation. To simplify, $\phi_{ij} = \Phi_h^l(i + i_k, j + j_k)$. d_k represents the k-th displacement, i.e.,

$$d_k = \begin{cases} q_k^l - p_k^l, & if\ \ h = 1, \\ q_k^l - p_k^l - f_k(\Phi_{h-1}^l), & if\ \ 2 \le h \le H, \end{cases} \tag{3}$$

in which

$$f_k(\Phi_{h-1}^l) = \sum_{i=0}^{3} \sum_{j=0}^{3} \hat{\omega}_{ij} \Phi_{h-1}^l(i + i_k, j + j_k), \tag{4}$$

where Φ_{h-1}^l is the $(h-1)$-th level control lattice. In this article, $H = 6$.

The control points for multiple feature points are often overlapped, as shown in Figure 2(d), which will affect the final control point values. To address it, the control point value is updated by

$$\hat{\phi}_{ij} = \frac{\sum_{s=0}^{3} \sum_{r=0}^{3} \hat{\omega}_{sr}^2 \phi_{sr}}{\sum_{s=0}^{3} \sum_{r=0}^{3} \hat{\omega}_{sr}^2}. \tag{5}$$

4.1.3. B-Spline Refinement. The B-spline refinement (shown in Figure 6(c)) is applied to reduce the computation of the warping surface at each level. The control point spacing of Ω'_h is as half large as that of Ω_h [Lee et al. 1997]. For this restriction, the B-spline curves [Lyche and Morken 1986] is applied to the refinement. Let ϕ_{ij} and θ_{ij} be the control points in Ω_h and Ω'_h, respectively. The control point value in Ω'_h is obtained by

$$\theta_{2i+s,2j+r}^l = \Theta\left(A_{sr} \odot U\right), \quad s, r = 0, 1, \tag{6}$$

where $A_{00} = [1\ \ 6\ \ 1;\ \ 6\ \ 36\ \ 6;\ \ 1\ \ 6\ \ 1]/64$, $A_{01} = [0\ \ 1\ \ 1;\ \ 0\ \ 6\ \ 6;\ \ 0\ \ 1\ \ 1]/16$, $A_{10} = [0\ \ 0\ \ 0;\ \ 1\ \ 6\ \ 1;\ \ 1\ \ 6\ \ 1]/16$, $A_{11} = [0\ \ 0\ \ 0;\ \ 0\ \ 1\ \ 1;\ \ 0\ \ 1\ \ 1]/4$. $U = [\phi_{i-1,j-1}\ \phi_{i-1,j}\ \phi_{i-1,j+1};\ \phi_{i,j-1}\ \phi_{i,j}\ \phi_{i,j+1};\ \phi_{i+1,j-1}\ \phi_{i+1,j}\ \phi_{i+1,j+1}]$. The symbols $\odot$ and $\Theta(\cdot)$ represent the dot product operation and the sum of matrix elements, respectively.

4.2. Energy Function

The control lattice Ψ_1^1 in the horizontal direction (or Ψ_1^2 in the vertical direction) on the first frame is initialized by the MBA method, as shown in Algorithm 1. The control lattice Ψ_t^l (in either the horizontal or vertical direction) on each of the subsequent frames ($t > 1$) is obtained by updating the control lattice (in the corresponding direction) on its preceding frame (i.e., Ψ_{t-1}^l), i.e., minimizing the proposed energy function (Equation (7)).

To highly preserve the temporal coherence and spatial smoothness of the control lattices, the goal of TSSW is to solve a problem of an energy function minimization. To calculate the control lattice Ψ_t^l, a new energy function is formulated by a weighted sum of a data-driven term, a smoothness term, and feature point constraints:

$$E(\Psi_t^l) = E_d(\Psi_t^l) + \alpha E_s(\Psi_t^l) + \beta E_f(\Psi_t^l), \quad t > 1, \tag{7}$$

where $\alpha > 0$ and $\beta > 0$ are two regularization parameters that balance the tradeoff among the data-driven term $E_d(\Psi_t^l)$, the smoothness term $E_s(\Psi_t^l)$, and the feature point constraints $E_f(\Psi_t^l)$.

4.2.1. Data-Driven Term. The data-driven term is designed to penalize the difference between the control lattice on the current frame Ψ_t^l and the control lattice on its preceding frame Ψ_{t-1}^l by using the sum-of-squared-differences (SSD) criterion, which is defined as

$$E_d(\Psi_t^l) = \sum_{i=1}^{m_H} \sum_{j=1}^{n_H} \left(\Psi_t^l(i,j) - \Psi_{t-1}^l(i,j)\right)^2. \tag{8}$$

Due to the salient property of the human faces, the data-driven term can effectively measure the control lattices on two consecutive frames on the face region. This term also guarantees that the differences of the two control lattices are as small as possible to achieve a high temporal coherence of the control lattices.

4.2.2. Smoothness Term. In mathematical analysis, spatial smoothness has to do with how many partial derivatives of a function exist and are continuous in the spatial domain. A smoothness term imposed by B-spline functions can guarantee the first-order partial derivative continuity at the control points and the second-order partial derivative continuity everywhere else [Huang et al. 2006]. The smoothness term is designed to preserve the spatial smoothness of the control lattice (in either the horizontal or vertical direction) on the current frame using its gradients. An efficient smoothness term for Ψ_t^l is defined as

$$E_s(\Psi_t^l) = \sum_{i=1}^{m_H} \sum_{j=1}^{n_H} \left(\nabla_x^2(i,j) + \nabla_y^2(i,j)\right), \tag{9}$$

where $\nabla_x = \partial\Psi_t^l/\partial x$ and $\nabla_y = \partial\Psi_t^l/\partial y$ represent the gradients of the control lattice Ψ_t^l in the horizontal and vertical directions. A small value of the smoothness term means that the control lattice is spatially smooth in the image domain; otherwise, the synthesized results obtained by the control lattice will be neither natural nor plausible.

4.2.3. Feature Point Constraints. Only the data-driven and smoothness terms in the energy function are suboptimal, which leads to a large estimation error. To address it, we impose the constraints on the facial feature points, which are modeled as an SSD measure between the warping surface values on the target localizations of the facial feature points (i.e., P_t) and the displacements $Z_t = \{(z_{t,k}^1, z_{t,k}^2)\}_{k=1}^K = Q_t - P_t$. The feature point constraints are formulated into

$$E_f(\Psi_t^l) = \sum_{(x_k,y_k)\in P_t} \left(f_k(\Psi_t^l) - z_{t,k}^l\right)^2, \tag{10}$$

where $z_{t,k}^1$ and $z_{t,k}^2$ represent the displacements in the horizontal and vertical directions, respectively. $f_k(\Psi_t^l)$ is the warping surface value on the k-th image point (x_k, y_k), which is computed by using the matrix notation:

$$f_k(\Psi_t^l) = {W_k}^T \Psi_t^l, \tag{11}$$

where W_k is the weight matrix and defined as

$$W_k(i + i_k, j + j_k) = \begin{cases} B_i(u_k)B_j(v_k), & \mathit{if}\ \ 0 \le i, j \le 3, \\ 0, & \mathit{otherwise}. \end{cases} \tag{12}$$

Since TSSW is still based on the facial feature point localizations, the feature point constraints are significant for improving the estimation accuracy of the control lattices. Hence, the regularization parameter β is set to a larger value than α in Equation (7).

4.3. Optimization

The energy function (Equation (7)) that combines Equations (8) through (10) is a quadratic form with respect to the control lattice Ψ_t^l, which can attain the optimal minimum when Ψ_t^l satisfies the following Euler-Lagrange equation:

$$\frac{\partial E}{\partial \Psi_t^l} - \frac{\partial}{\partial x}\frac{\partial E}{\partial \nabla_x} - \frac{\partial}{\partial y}\frac{\partial E}{\partial \nabla_y} = 0, \tag{13}$$

where

$$\frac{\partial E}{\partial \Psi_t^l} = 2\left(\Psi_t^l - \Psi_{t-1}^l\right) + 2\beta \sum_{k=1}^{K} W_k \left(W_k^T \Psi_t^l - z_{t,k}^l\right), \tag{14}$$

$$\frac{\partial}{\partial x}\frac{\partial E}{\partial \nabla_x} = 2\alpha \frac{\partial^2 \Psi_t^l}{\partial x^2}, \qquad \frac{\partial}{\partial y}\frac{\partial E}{\partial \nabla_y} = 2\alpha \frac{\partial^2 \Psi_t^l}{\partial y^2}. \tag{15}$$

Combining the preceding equations, Equation (13) is rewritten as

$$\left(\Psi_t^l - \Psi_{t-1}^l\right) + \beta \sum_{k=1}^{K} \left(C_k \Psi_t^l - W_k z_{t,k}^l\right) - \alpha L \Psi_t^l = 0, \tag{16}$$

where $C_k = W_k W_k^T$ and $L = D_x^T D_x + D_y^T D_y$ is the homogeneous Laplacian matrix. D_x and D_y represent the forward difference operators. C_k $(1 \le k \le K)$ and L are both symmetric. After reorganizing and simplifying, the resultant linear system for Ψ_t^l is formulated as

$$\left(I + \beta \sum_{k=1}^{K} C_k - \alpha L\right) \Psi_t^l = \left(\Psi_{t-1}^l + \beta \sum_{k=1}^{K} W_k z_{t,k}^l\right), \tag{17}$$

where I is an identity matrix of size $m_H n_H \times m_H n_H$. The control lattices in the horizontal and vertical directions (i.e., $\{\Psi_t^1, \Psi_t^2\}$ on the t-th frame) are obtained by solving Equation (17), which is optimized by using the conjugate gradient technique in a sequence of iterations. In the iterative process, the control lattice is gradually updated with highly temporally coherent information. According to the control lattice Ψ_t^l, the corresponding warping surface in the image domain $\{(x_k, y_k) \mid 1 \le x_k \le M, 1 \le y_k \le N\}$ is computed by using Equation (11). Then, the warped frame X_{warp}^t regarding the input

ALGORITHM 2: Temporal-Spatial-Smooth Warping (TSSW) Method

```
Input: The original video frames $\{X^t_{in}\}^{\mathbb{T}}_{t=1}$ and the source and target localizations of the facial
        feature points $\{(Q_t, P_t)\}^{\mathbb{T}}_{t=1}$, and two regularization parameters $\alpha$ and $\beta$.
Output: The warped video frames $\{X^t_{warp}\}^{\mathbb{T}}_{t=1}$.
Initialization:
Compute the initial control lattices $\{\Psi^l_1\}_{l=1,2}$ by Algorithm 1;
Calculate the warping surfaces on $\{(x_k, y_k) \mid 1 \leq x_k \leq M, 1 \leq y_k \leq N\}$ by using Equation (11);
Obtain the first warped frame $X^1_{warp}$ by using Equation (18);
Compute the Laplacian matrix $L$ of size $m_H n_H \times m_H n_H$;
Estimation:
for $t = 2, 3, \ldots, \mathbb{T}$ do
    Calculate $\sum_{k=1}^{K} W_k W_k^T$ regarding $P_t$ by using Equations (12) and (1);
    for $l = 1, 2$ do
        Calculate $\sum_{k=1}^{K} W_k z^l_{t,k}$ by the geometric displacements derived from $(Q_t, P_t)$;
        repeat
            Calculate Equation (17) by using the conjugate gradient technique;
        until reach the tolerance or the maximum number of iterations;
        Obtain the control lattice $\Psi^l_t$ on the current frame;
    end
    Calculate the two warping surfaces $f(\Psi^l_t)$ and $f(\Psi^2_t)$ in the $M \times N$ domain by using
    Equation (11);
    Obtain the warped frame $X^t_{warp}$ by using Equation (18);
    $t = t + 1$.
end
```

frame X^t_{in} is obtained by using the bicubic interpolation method:

$$X^t_{warp} = Interp\big(X^t_{in},\ \ f(\Psi^1_t),\ \ f(\Psi^2_t)\big), \tag{18}$$

where $Interp(\cdot)$ represents the bicubic interpolation operation.

The process of the proposed TSSW method is summarized in Algorithm 2, which leads to higher temporally coherence and spatial smoothness, and therefore more natural-looking results.

5. APPLICATIONS AND RESULTS

We implement and test the proposed approaches on an Intel Core 2 Duo 3GHz CPU and 4GB memory in the Matlab environment. First, we conduct a reconstruction performance experiment to show the advantages of the proposed method, especially in preserving temporal coherence. Section 5.1 illustrates that TSSW has better effectiveness than the MBA algorithm. Then, we conduct a number of validations for the four applications: facial attractiveness enhancement, makeup transfer, face replacement, and expression manipulation. Table III shows the information of videos from the YouTube Web site[3] and used in our work, as well as the corresponding timing statistics of our methods. For all experiments, the two regularization parameters in Equation (17) are empirically set to $\alpha = 0.8$ and $\beta = 1$. Using the conjugate gradient technique, the optimization step is iterated about 30 times. The overall results are available online.[4]

[3] https://www.youtube.com/.

[4] http://youtu.be/LQCLeQcBS74.

Table III. Video Information and Runtime (Seconds) Obtained by the Proposed Method

(a) Attractiveness and Makeup

Name		Resolution	NoF	TPF
Attractiveness	Video 1	480 × 856	301	6.269
	Video 2	360 × 640	3,000	4.412
	Video 3	720 × 1280	144	6.097
Makeup	Video 4	720 × 1280	115	19.211
	Video 5	720 × 1280	250	22.934
	Video 6	720 × 1280	1,625	15.755

(b) Replacement and Manipulation

Name		Resolution	NoF	TPF
Replacement	Video 7	704 × 1243	699	23.063
	Video 8	720 × 1280	88	17.193
	Video 9	688 × 1280	86	57.632
Manipulation	Video 10	720 × 1280	2,000	7.017
	Video 11	720 × 1280	150	6.001
	Video 12	720 × 1280	699	6.882

NoF, total number of video frames; TPF, average runtime per frame.
Note: The timing of feature detection and tracking is not included in this table. Regarding replacement, this table only shows the information of the target video.

5.1. Reconstruction Performance

According to Franke [1982], the accuracy for reproducing a known surface is an effective performance measure. The test function [Franke 1982] is used to generate the corresponding surface values on the sampled data points and then serves as a baseline for evaluating the performance of reconstruction accuracy. Five test functions that are often used in Franke [1982], Nielson [1993], and Lee et al. [1997] are defined as

$$\begin{aligned}
g_1(x, y) &= 0.75\exp\big(-(9x_s-2)^2/4-(9y_s-2)^2/4\big)\\
&\quad+0.75\exp\big(-(9x_s+1)^2/49-(9y_s+1)/10\big)\\
&\quad+0.5\exp\big(-(9x_s-7)^2/4-(9y_s-3)^2/4\big)-0.2\exp\big(-(9x_s-4)^2-(9y_s-7)^2\big),\\
g_2(x, y) &= \big(\tanh(9-9x_s-9y_s)+1\big)/9,\\
g_3(x, y) &= \big(1.25+\cos(5.4y_s)\big)/\big(6+6(3x_s-1)^2\big),\\
g_4(x, y) &= \exp\big(-20.25(x_s-0.5)^2-20.25(y_s-0.5)^2\big)/3,\\
g_5(x, y) &= \sqrt{64/81-(x_s-0.5)^2-(y_s-0.5)^2}-0.5,
\end{aligned}$$

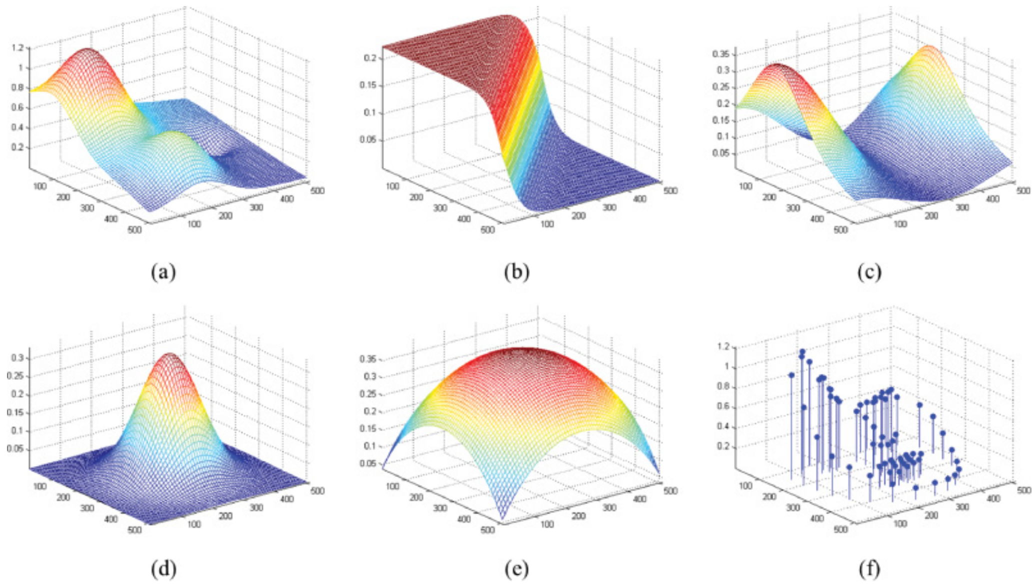
where $x_s = (x-1)/(M-1)$ and $y_s = (y-1)/(N-1)$ for $x = 1, 2, \ldots, M$ and $y = 1, 2, \ldots, N$. In this experiment, we chose $M = N = 512$.

The surfaces of the five test functions are shown in Figure 7(a) through (e). The approximate ranges of these functions are g_1:[0.0, 1.22]; g_2:[0.0, 0.22]; g_3:[0.0, 0.38]; g_4:[0.0, 0.33]; g_5:[0.0, 0.39]. These functions are spatially smooth in the image domain, which is in accordance with the real-world sampling. Hence, in this article, the five test functions are used as benchmarks to measure and evaluate the performance of the reconstruction accuracy.

The difference between an approximation function $f(x, y)$ and $g(x, y)$ is measured by computing the root mean square (RMS) error (Equation (19)):

$$\mathrm{RMS} = \sqrt{\frac{\sum_{x=1}^{M}\sum_{y=1}^{N}(f(x, y)-g(x, y))^2}{MN}}. \tag{19}$$

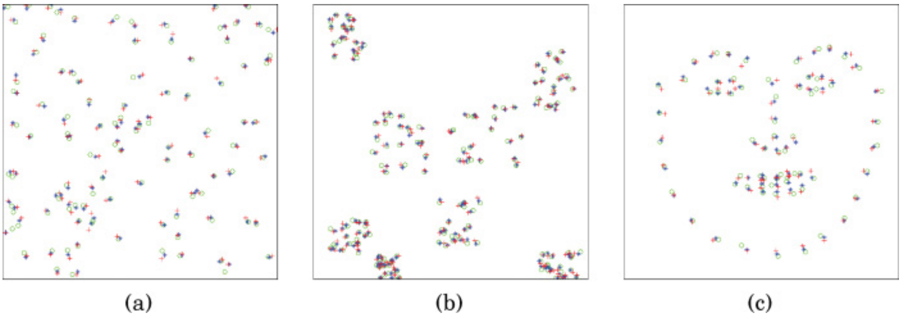
Figure 8 shows three types of sampled data points placed in the unit image domain. R100 represents 100 points randomly sampled and placed in the image domain. For C160, we first divided the image domain into eight regions with different sizes and randomly located 20 point positions in each region. C160 is a dataset of densely sampled areas with large gaps between cluster centers. F66 consists of 66 data points sampled from the active appearance model (AAM) [Li et al. 2013].



(a) (b) (c)

(d) (e) (f)

Fig. 7. Test functions used for evaluating reconstruction accuracy. (a) g_1. (b) g_2. (c) g_3. (d) g_4. (e) g_5. (f) The surface values of g_1 on the data points $\mathcal{P}_1$ from F66.



(a) (b) (c)

Fig. 8. Sampling positions for test functions, where "o," $\mathcal{P}_0$; "$*$," $\mathcal{P}_1$; and "+," $\mathcal{P}_2$. (a) R100. (b) C160. (c) F66.

For each type of sampled data points, we first construct the data points in the image domain using the Matlab function "randi." This dataset is denoted as $\mathcal{P}_1$. Then, we use the Matlab function "randn" multiplied by a factor s ($s = 3 \sim 5$) to generate two sets of displacements added to $\mathcal{P}_1$ for obtaining the two datasets $\mathcal{P}_0$ and $\mathcal{P}_2$. As illustrated in Algorithm 1 and Equation (3), when applied to warping, we need to know the target point position and the corresponding displacement, defined as $\{(x_k, y_k, d_k) \mid k = 1, \ldots, K\}$, where $d_k = g(\lfloor x_k \rfloor, \lfloor y_k \rfloor)$.

To better illustrate the difference between TSSW and MBA, we have separately compared the performance of reconstruction accuracy and temporal coherence, which are determined by computing the average RMS (ARMS) error (Equation (20)) and average sum-of-squared-differences (ASSD) (Equation (21)), respectively. The smaller the error value, the better performance will be:

$$\text{ARMS} = \frac{\sum_{t=1}^{\mathbb{T}} \text{RMS}_t}{\mathbb{T}}, \tag{20}$$

Table IV. Comparisons of TSSW and MBA on Five Test Functions and Three Types of Data Points

Functions	MBA			TSSW		
	R100	C160	F66	R100	C160	F66
g_1	0.1748	0.1685	0.1680	0.1756	0.1678	0.1689
	0.4332	0.3120	0.7820	0.1939	0.1035	0.1238
g_2	0.0333	0.1629	0.1063	0.0334	0.1625	0.1052
	0.1306	0.1167	0.1827	0.0657	0.0470	0.0369
g_3	0.3349	0.2597	0.2782	0.3332	0.2610	0.2788
	0.1604	0.1298	0.1743	0.1154	0.0685	0.0797
g_4	0.0113	0.0929	0.1105	0.0120	0.0921	0.1091
	0.1018	0.0772	0.2050	0.0775	0.0712	0.1188
g_5	0.0502	0.0839	0.1616	0.0501	0.0836	0.1615
	0.1802	0.0995	0.1649	0.1347	0.0994	0.0569

Note: For each test function, the first row is ARMS, whereas the second row is ASSD.

$$\mathrm{ASSD} = \frac{\sum_{t=2}^{\mathbb{T}} \sqrt{\sum_{i=1}^{m_H} \sum_{j=1}^{n_H} (\Psi_t(i,j) - \Psi_{t-1}(i,j))^2}}{\mathbb{T} - 1}, \tag{21}$$

where RMS_t is obtained by Equation (19) for the t-th frame obtained by MBA or TSSW. In this experiment, $\mathbb{T} = 3$.

As shown in Table IV, the experiments on five test functions and three types of data points demonstrate that TSSW has similar ARMS values as MBA but smaller ASSD values than MBA. ARMS values obtained by TSSW and MBA are partially similar because they use the same feature point constraints. ASSD values obtained by TSSW are largely smaller because TSSW effectively exploits the temporal coherence of the control lattices on two consecutive frames. It illustrates that TSSW has better effectiveness than MBA.

5.2. Facial Attractiveness Enhancement

A training dataset of neutral faces (101 female faces and 94 male faces) are used in this experiment. The 66 facial feature points on each neutral face are obtained by ISDM, which are numbered 1 through 66 in Figure 9(a). Among the 66 facial feature points, 174 edges are constructed by the Delaunay triangulation technique, as shown in Figure 9(b). Then, the lengths of the 174 edges are composed into a 174-dimensional distance vector. Since different faces are usually of different areas, we normalize the 174-dimensional distance vector by the area of the face. We use global symmetrization and overall proportion optimization [Liao et al. 2012]. Denote $\{(V_s^i, b_i)\}_{i=1}^{n_s}$ as the training data of the female or male training dataset, where n_s, V_s^i, and b_i are the number of training faces, the 174-dimensional distance vector, and the corresponding beauty score regarding the i-th training face, respectively.

Given a video, we first select the training dataset (female or male). Then, we calculate the distance vectors of the input video frames, i.e., $\{V_i\}_{i=1}^{\mathbb{T}}$. In this article, we propose a kernel-based similarity. The similarity weight between the distance vector of the input frame V_i and that of the training face V_s^j is formulated by

$$w_{ij} = b_j \cdot \exp\left(-\frac{\left\|V_i - V_s^j\right\|_2^2}{\sigma^2}\right), \tag{22}$$

where σ is the kernel parameter in the weighting computation ($\sigma = 5$ in this experiment). Denote v_{neu} as the distance vector of the input frame with neutral expression.

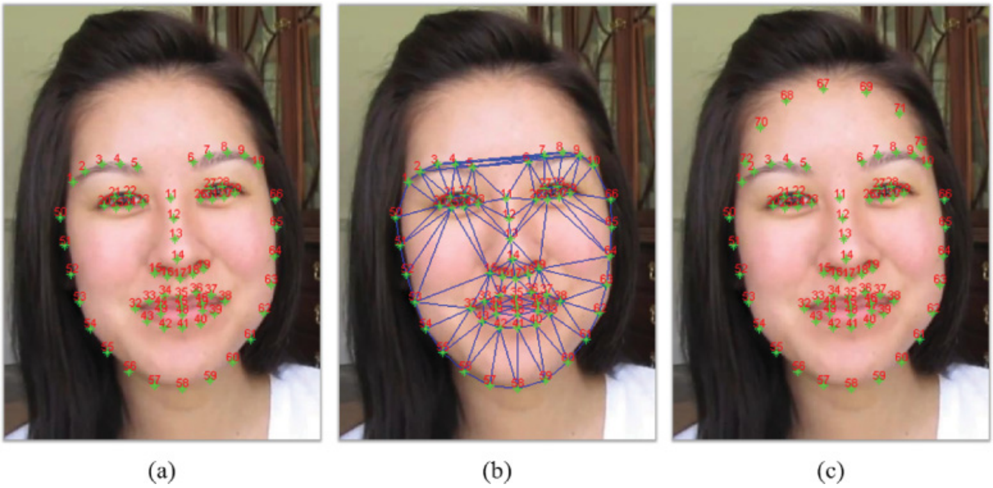


Fig. 9. Facial feature localization. (a) 66 facial feature points. (b) 174 edges among the 66 facial feature points are created by the Delaunay triangulation technique. (c) 73 feature points including an additional 7 forehead points.

For the i-th input frame, the new distance vector V_i' with a higher attractiveness rating is computed by

$$V_i' = \frac{\sum_{j \in \Gamma(v_{neu})} w_{ij} V_s^j}{\sum_{j \in \Gamma(v_{neu})} w_{ij}}, \tag{23}$$

where $\Gamma(v_{neu})$ represents a set of the indexes of the five training faces, whose distance vectors are relatively close to v_{neu}. According to the new distance vectors $\{V_i'\}_{i=1}^{\mathbb{T}}$, the modified localizations of the feature points are obtained by using the Levenberg-Marquardt (LM) algorithm [Marquardt 1963]. Then, the warped video frames are generated by the proposed TSSW method, as shown in Algorithm 2.

Figure 10 shows the warping surfaces in the horizontal (x) and vertical (y) directions and the warped video frames obtained by MBA and TSSW. From Figure 10, we can see that MBA produces large variations of the warping surfaces when a high temporal incoherence of facial feature points on two consecutive frames exists, which suffers from the problem of low temporal coherence in the synthesized result. This is largely because MBA strongly imposes feature point constraints and is very sensitive to the temporal incoherence of facial feature point localizations. Compared to MBA, TSSW achieves highly temporally coherent results. This is because the temporal coherence and spatial smoothness of the control lattices are effectively preserved by minimizing the energy function, i.e., Equation (7). Figure 11 and the online video demonstration[5] show the facial attractiveness enhancement results obtained by different algorithms (i.e., Leyvand et al. [2008]), the method of combining MBA and the proposed kernel-based similarity, the method of combining MLS and the proposed kernel-based similarity, and the method of combining the proposed TSSW and kernel-based similarity. Since the MFFD used in Leyvand et al. [2008], MBA, and MLS methods are single-frame based and do not consider the temporal information between consecutive frames, these methods share the temporal incoherence problem in the synthesized videos. The experimental results on facial attractiveness enhancement demonstrate that TSSW is effectively

[5] https://www.youtube.com/watch?v=8ZYUXlNpeOg&feature=youtube.

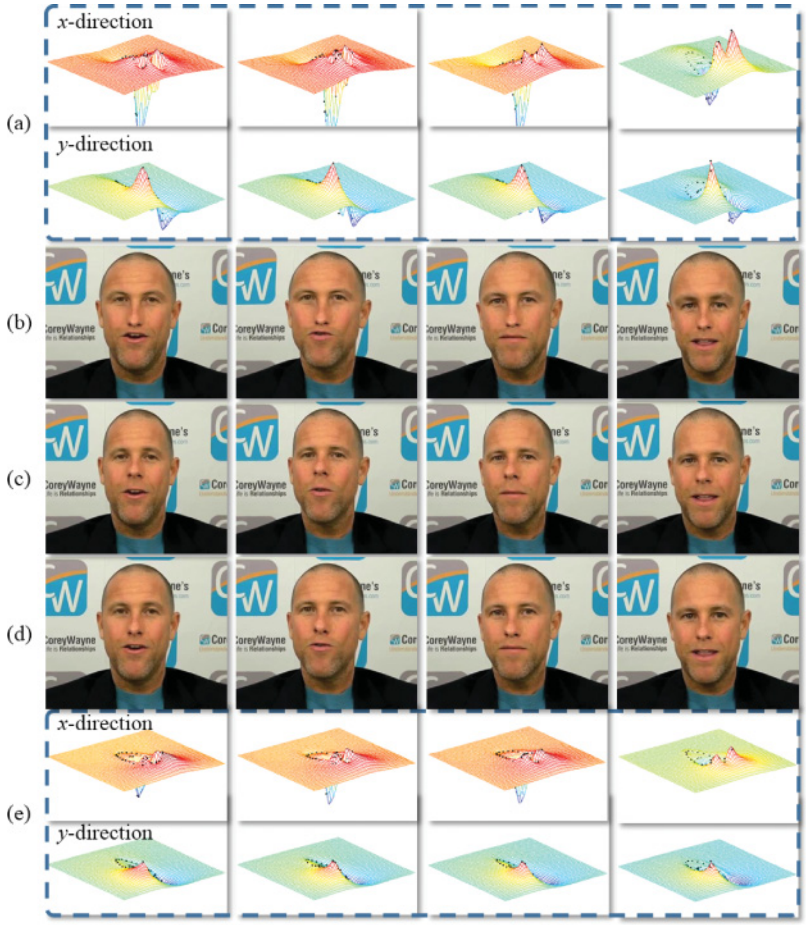


Fig. 10. Facial attractiveness enhancement results and the corresponding warping surfaces. (a) The horizontal and vertical warping surfaces, i.e., $f(\Psi_t^1)$ and $f(\Psi_t^l)$ obtained by MBA. (b) The warped frames obtained by Equation (18) with the corresponding two warping surfaces in (a). (c) The original frames. (d) The warped frames using the warping surfaces in (e). (e) The horizontal and vertical warping surfaces obtained by TSSW.

applied to such a video face editing task and produces more natural-looking results than directly applying the existing warping methods.

5.3. Makeup Transfer

Only modifying the facial feature point localizations may not significantly improve facial attractiveness, because the inherent skin features (e.g., freckles and acne) also affect the ratings of facial attractiveness. Makeup transfer is an effective way to improve detail of the skin.

A template dataset is used to select a template whose skin color is close to that of the input face. To improve the skin detail on the forehead region, we manually add seven forehead points on the first frame, numbered 67 through 73 in Figure 9(c). According to the facial feature points on the input video frames and the selected template, the warped versions of the template are obtained by exploiting the proposed TSSW method.

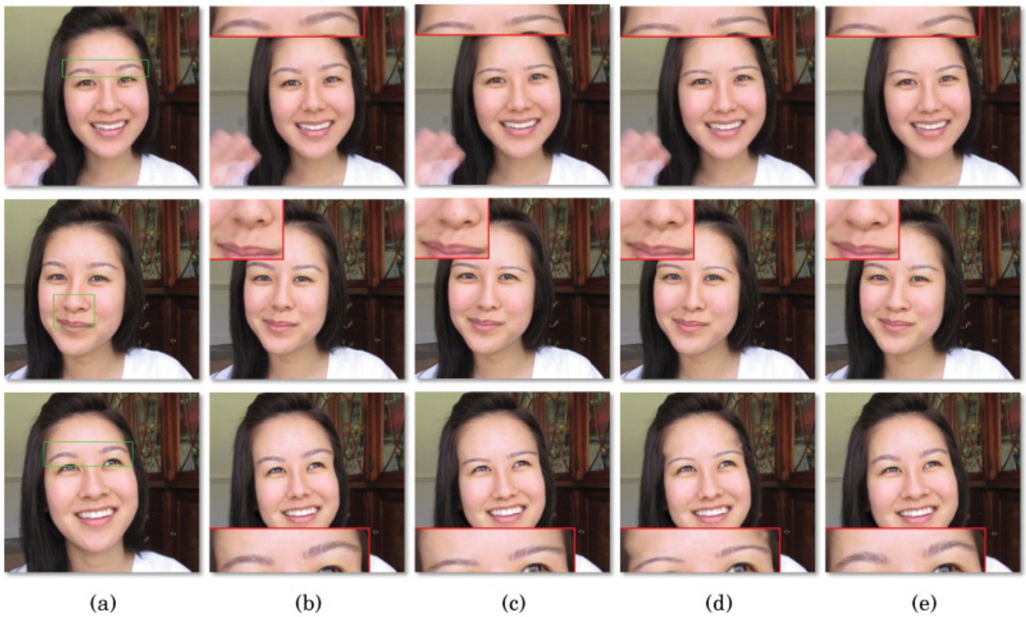


Fig. 11. Comparison of facial attractiveness enhancement results (Video 3). (a) Original frames. (b) Leyvand et al. [2008]. (c) MBA [Lee et al. 1997] + kernel. (d) MLS [Schaefer et al. 2006] + kernel. (e) Ours.

All input video frames and warped versions of the template are separately decomposed into three layers: a structure layer, a skin layer, and a color layer [Guo and Sim 2009]. For each input frame, the information in the skin layer is modified by a weighted addition of that of the warped template and itself. The color layer of the warped template is transferred onto that of the input frame by using the alpha blending technique [Raman and Chaudhuri 2007]. The intrinsic structure layer, the modified skin layer, and the transferred color layer of each input frame are composed together to obtain the synthesized frame. To make the face boundary look natural, we use the Poisson method [Pérez et al. 2003] to improve the synthesized result.

Figure 12 shows the makeup transfer results obtained by different algorithms, i.e., Yang et al. [2012b] and the proposed method without or with the forehead points. The synthesized frames obtained by Yang et al. [2012b] are not plausible, especially for the skin detail on the mouth and nose regions. This is because the estimation of the 3D face model parameters is not accurate for all input frames, which leads to temporal incoherence in the synthesized videos. The proposed method produces natural-looking results by further exploring the additional forehead points. Compared to Yang et al. [2012b], the makeup transfer results obtained by exploiting the proposed TSSW are temporally coherent without using costly 3D face models. Figure 13 shows the beautified results by simultaneously enhancing the facial attractiveness and improving the skin detail of the input face. The makeup transfer and beautified results are shown in the online videos.[6]

5.4. Face Replacement

Given a source video with the desired face and a target video, the facial feature point localizations for the source and target videos are detected and tracked using the

[6] http://youtu.be/xdCF9RyIuOM.

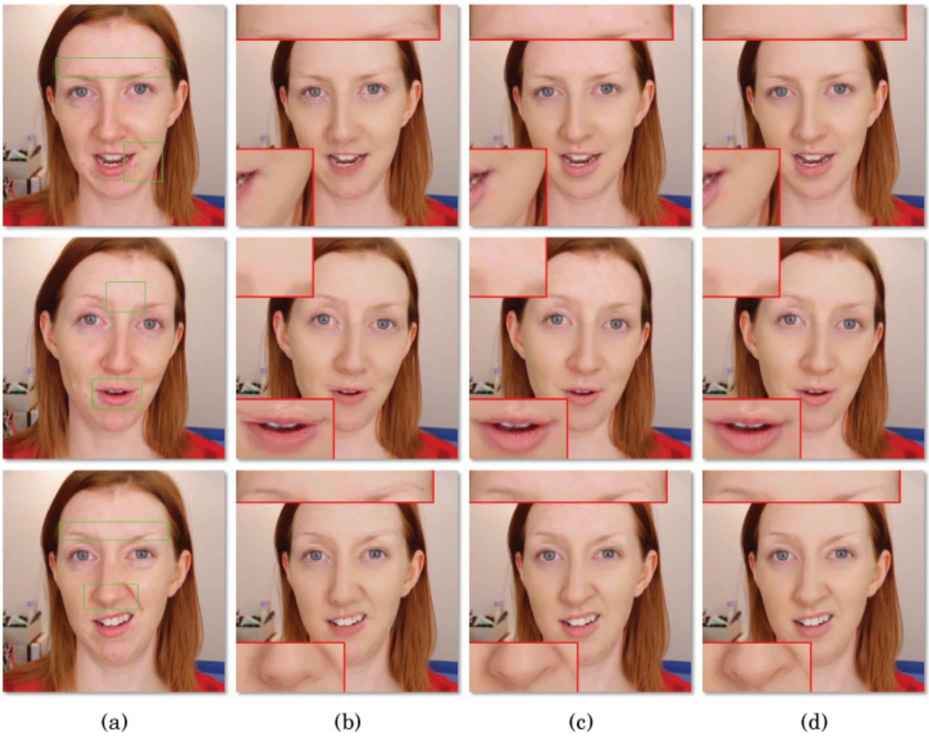


Fig. 12. Comparison of makeup transfer results (Video 6). (a) Original frames. (b) Result of Yang et al. [2012b]. (c) Our result without forehead points. (d) Our result with forehead points.

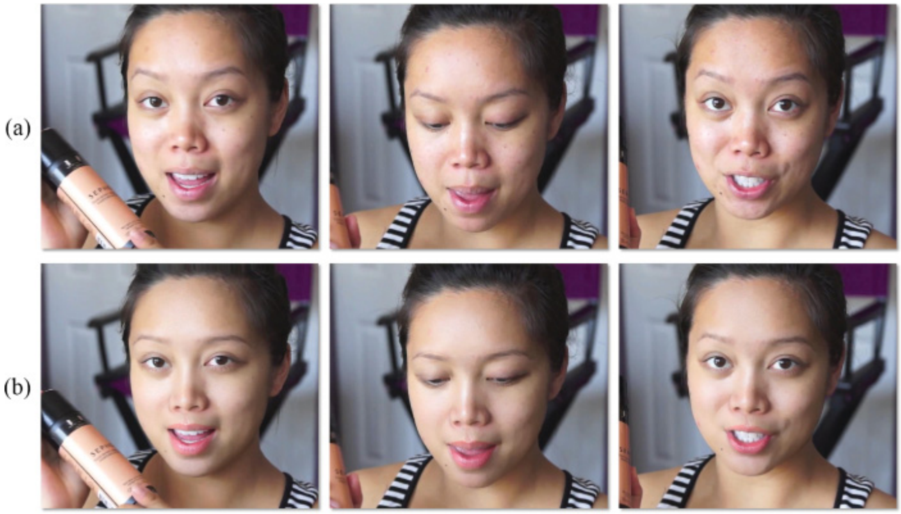


Fig. 13. Beautified results. (a) Original frames. (b) Our result (facial attractiveness enhancement + makeup transfer).

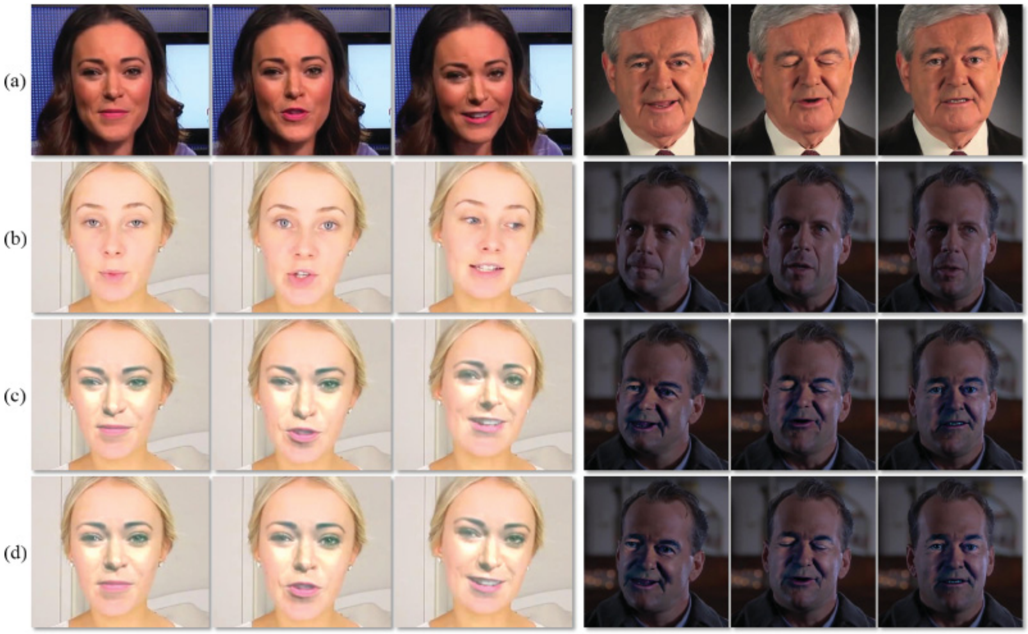


Fig. 14. Comparison of face replacement results on Video 8 and Video 9. (a) Retimed frames after RCTW. (b) Target frames. (c) Dale et al. [2011]. (d) Our result.

proposed ISDM method. The source frames are retimed by using the robust canonical time warping (RCTW) method [Panagakis et al. 2013] to match the facial expressions of the target frames. Then, the facial feature points of each retimed source frame are projected on its corresponding target frame by using the affine transformation. According to the source and target facial feature point pairs, the proposed TSSW method generates the warped versions of the retimed source frames. To produce a more natural-looking composite, we apply the Poisson method to blend the face boundary.

Figure 14 and the online videos[7] show the face replacement results obtained by Dale et al. [2011] and the proposed method. Dale et al. [2011] generated the synthesized frames by deforming the 3D face meshes; however, it produced obvious artifacts along the editing boundary when the shapes of two faces were very different. In addition, the 3D face models used in Dale et al. [2011] require intensive manual interventions on many key frames for estimating the 3D model parameters. Compared to Dale et al. [2011], the proposed method for the face replacement task does not use the costly 3D face models, which reduces manual costs and produces natural-looking and temporally coherent results.

5.5. Expression Manipulation

To manipulate the facial expressions of a face on the input frames, the localizations of the facial feature points (especially on the mouth region) on each video frame need to be adjusted by a fixed manipulation factor c. When $c > 1$, it represents the exaggeration of the facial expressions, whereas it represents the neutralization of the facial expressions for $c < 1$. The adjusted localizations of the facial feature points are then projected on the corresponding video frames. The synthesized frames are obtained by employing the proposed TSSW method.

[7] http://youtu.be/ZL1hncJ9BMA.

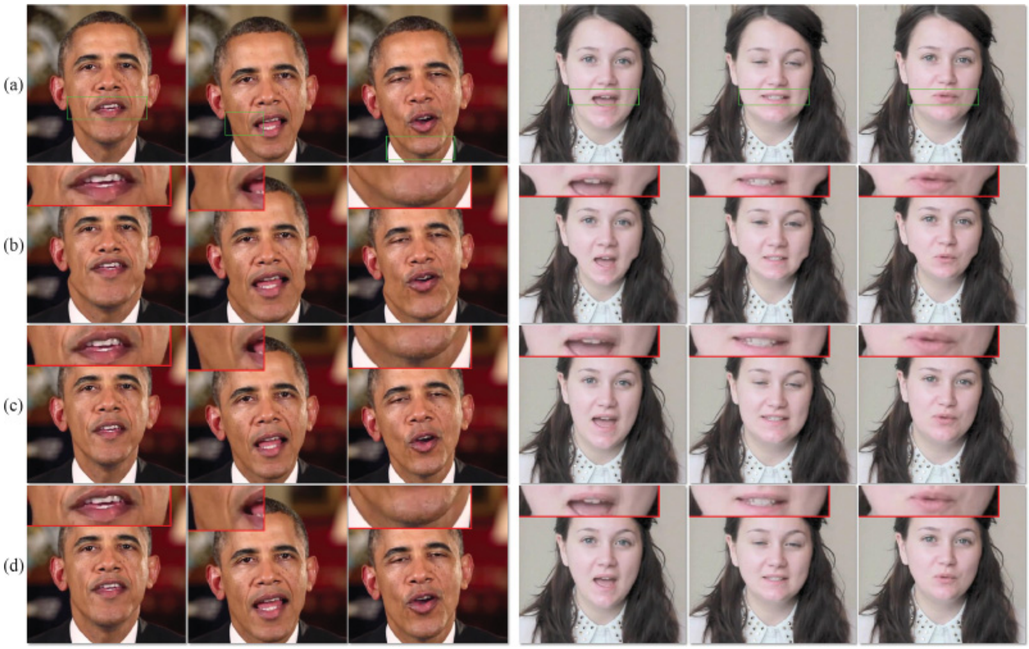


Fig. 15. Comparison of expression manipulation results. Left with manipulation factor $c = 1.15$ (Video 10) and right with $c = 0.85$ (Video 11). (a) Original frames. (b) Result of Ma et al. [2013]. (c) Result of Yang et al. [2012a]. (d) Our result.

Figure 15 and the video demonstration[8] show the exaggerated and neutralized results obtained by Ma et al. [2013], Yang et al. [2012a], and the proposed method. Since the MRLS method used in Ma et al. [2013] is mostly single-frame based, the facial expressions on two consecutive frames are severely deformed, which leads to unnatural results. Although Yang et al. [2012a] can edit facial expressions, the stationary objects on the nonface regions are also deformed across the video sequence, which suffers from the problem of temporal incoherence in the synthesized videos. TSSW for the expression manipulation task achieves better results and preserves temporal coherence in the synthesized videos.

5.6. User Study

Without the reference videos, a subjective evaluation obtained by human observers is probably the best way to validate the effectiveness of video face editing. This is due to the sensitivity of human observers to the visual information in the resultant videos. There exist several complicated approaches based on the subjective results, such as Xu et al. [2008], Nguyen et al. [2011], Reches et al. [2014], and Li et al. [2014]. Following Song et al. [2010], we exploited the paired comparison method and performed a user study on Amazon Mechanical Turk[9] to validate the effectiveness of the proposed approaches. For each video, we invited 100 participants from diverse backgrounds and aged 20 to 45 years. The participants were presented with two synthesized results side by side at one time and then were asked to choose the video that they preferred. During the survey, videos obtained by different methods were randomly ordered to avoid bias.

[8] http://youtu.be/mhzNP3CF0uM.
[9] https://requester.mturk.com/.

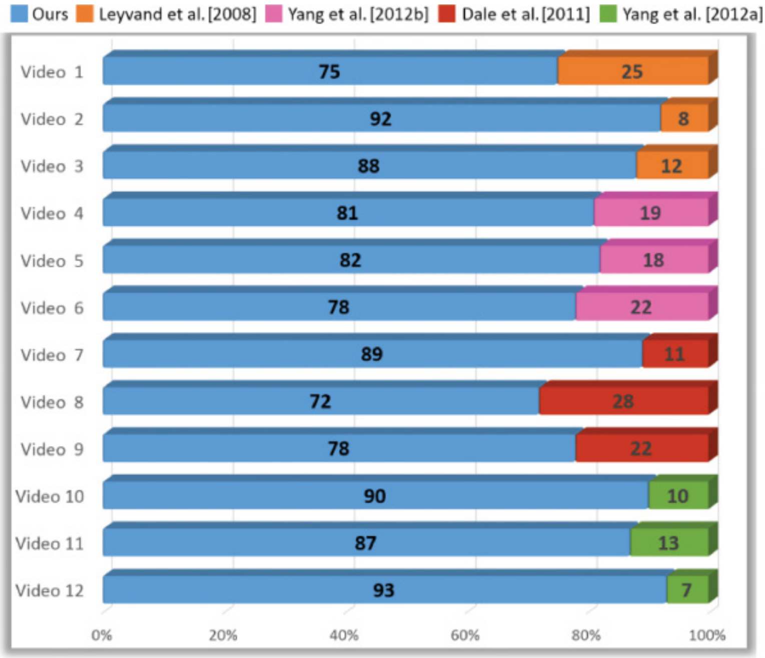


Fig. 16. The stacked bar chart of participants' preferences for our methods compared to Leyvand et al. [2008], Yang et al. [2012b], Dale et al. [2011], and Yang et al. [2012a] among Videos 1 through 12.

In this survey, we mainly compared our approaches to the existing face editing algorithms, i.e., results obtained by Leyvand et al. [2008], Yang et al. [2012b], Dale et al. [2011], and Yang et al. [2012a] for facial attractiveness enhancement, makeup transfer, face replacement, and expression manipulation applications, respectively. For facial attractiveness enhancement, we asked the participants to select the more attractive face (particularly unchanged face geometry throughout the video) in which side of each pair. For makeup transfer, the participants were asked to indicate the face with better skin details in which side of each pair. For replacement, they were asked to tick the more natural-looking face (especially with similar luminance of face region across the video sequence) in each pair. For expression manipulation, we asked them to select the more stable background in each pair. Figure 16 shows the participants' preference among examples from Video 1 through Video 12, which indicates that qualitative assessments obtained by our proposed approaches for the preceding four applications are better than those obtained by existing methods.

6. CONCLUSIONS AND FUTURE WORK

In this article, we have developed a novel and efficient warping method, TSSW, for video face editing by optimizing an energy function containing a data-driven term, a smoothness term, and feature point constraints. TSSW (1) is robust to subtly temporal incoherence of the facial feature point localizations; (2) preserves the temporal coherence and spatial smoothness in the control lattice generation; and (3) is easy to be extended to various video face editing tasks, such as facial expression synthesis or dubbing. The preceding three advantages of TSSW make it practically preferred for video face editing.

There still exist some shortcomings to the proposed method. First, although the SA improves the initialization of SDM, it cannot guarantee to obtain a precise initialization for all input videos. Second, due to the lack of 3D information, our method is suboptimal for large pose variations where complex facial geometries and large dynamic facial components need to be synthesized.

In the future, we plan to extend TSSW in two aspects: (1) exploiting and exploring a more effective tracking method for facial feature point localizations and (2) improving the efficiency of the control lattice estimation.

REFERENCES

K. Anjyo, J. P. Lewis, and F. Pighin. 2014. Scattered data interpolation for computer graphics. In *Proceedings of ACM SIGGRAPH 2014 Courses (SIGGRAPH'14)*. 27:1–27:69.

Nur Arad, Nira Dyn, Daniel Reisfeld, and Yehezkel Yeshurun. 1994. Image warping by radial basis functions: Application to facial expressions. *CVGIP: Graphical Models and Image Processing* 56, 2, 161–172.

Dmitri Bitouk, Neeraj Kumar, Samreen Dhillon, Peter Belhumeur, and Shree K. Nayar. 2008. Face swapping: Automatically replacing faces in photographs. In *Proceedings of the ACM Special Interest Group on Computer Graphics (SIGGRAPH'08)*. 39:1–39:8.

Xudong Cao, Yichen Wei, Fang Wen, and Jian Sun. 2012. Face alignment by explicit shape regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'12)*. 2887–2894.

O. Çeliktutan, S. Ulukaya, and B. Sankur. 2013. A comparative study of face landmarking techniques. *EURASIP Journal on Image and Video Processing* 2013, 1, 13.

Cao Chen, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. 2014. FaceWarehouse: A 3D facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics* 20, 3, 413–425.

Kevin Dale, Kalyan Sunkavalli, Micah K. Johnson, Daniel Vlasic, Wojciech Matusik, and Hanspeter Pfister. 2011. Video face replacement. *ACM Transactions on Graphics* 30, 6, 130:1–130:10.

Douglas DeCarlo and Dimitris Metaxas. 2000. Optical flow constraints on deformable models with applications to face tracking. *International Journal of Computer Vision* 38, 2, 99–127.

P. Dollar, P. Welinder, and P. Perona. 2010. Cascaded pose regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10)*. 1078–1085.

Gianluca Donato and Serge Belongie. 2002. Approximate thin plate spline mappings. In *Proceedings of the European Conference on Computer Vision (ECCV'02)*. 21–31.

Richard Franke. 1982. Scattered data interpolation: Tests of some methods. *Mathematics of Computation* 38, 157, 181.

Dong Guo and T. Sim. 2009. Digital face makeup by example. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'09)*. 73–79.

XiaoHong Han, Long Quan, and Xiaoyan Xiong. 2015. A modified gravitational search algorithm based on sequential quadratic programming and chaotic map for ELD optimization. *Knowledge and Information Systems* 42, 3, 689–708.

Xiaolei Huang, N. Paragios, and D. N. Metaxas. 2006. Shape registration in implicit spaces using information theory and free form deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 8, 1303–1318.

V. Jain and E. G. Learned-Miller. 2010. *FDDB: A Benchmark for Face Detection in Unconstrained Settings*. Technical Report. University of Massachussetts Amherst, Amherst, MA.

Ira Kemelmacher-Shlizerman, Aditya Sankar, Eli Shechtman, and Steven M. Seitz. 2010. Being John Malkovich. In *Proceedings of the European Conference on Computer Vision (ECCV'10)*. 341–353.

Pavel Korshunov and Wei Tsang Ooi. 2011. Video quality for face detection, recognition, and tracking. *ACM Transactions on Multimedia Computing, Communications, and Applications* 7, 3, 14:1–14:21.

Seungyong Lee, George Wolberg, and Sung Yong Shin. 1997. Scattered data interpolation with multilevel B-splines. *IEEE Transactions on Visualization and Computer Graphics* 3, 3, 228–244.

Seungyong Lee, George Wolberg, Kyung Yong Chwa, and Sung Yong Shin. 1996. Image metamorphosis with scattered feature constraints. *IEEE Transactions on Visualization and Computer Graphics* 2, 4, 337–354.

Tommer Leyvand, Daniel Cohen-Or, Gideon Dror, and Dani Lischinski. 2008. Data-driven enhancement of facial attractiveness. *ACM Transactions on Graphics* 27, 3, 38:1–38:9.

Nan Li, William Cushing, Subbarao Kambhampati, and Sungwook Yoon. 2014. Learning probabilistic hierarchical task networks as probabilistic context-free grammars to capture user preferences. *ACM Transactions on Intelligent Systems and Technology* 5, 2, 29:1–29:32.

Xi Li, Weiming Hu, Chunhua Shen, Zhongfei Zhang, Anthony Dick, and Anton Van Den Hengel. 2013. A survey of appearance models in visual object tracking. *ACM Transactions on Intelligent Systems and Technology* 4, 4, 58:1–58:48.

Xiaoyan Li, Tongliang Liu, Jiankang Deng, and Dacheng Tao. 2014. Video Face Editing Using Temporal-Spatial-Smooth Warping. Retrieved January 12, 2016, from https://sites.google.com/site/tsswmethod/.

Qiqi Liao, Xiaogang Jin, and Wenting Zeng. 2012. Enhancing the symmetry and proportion of 3D face geometry. *IEEE Transactions on Visualization and Computer Graphics* 18, 10, 1704–1716.

Shih-Syun Lin, Chao-Hung Lin, I.-Cheng Yeh, Shu-Huai Chang, Chih-Kuo Yeh, and Tong-Yee Lee. 2013. Content-aware video retargeting using object-preserving warping. *IEEE Transactions on Visualization and Computer Graphics* 19, 10, 1677–1686.

Feng Liu, Michael Gleicher, Hailin Jin, and Aseem Agarwala. 2009. Content-preserving warps for 3D video stabilization. *ACM Transactions on Graphics* 28, 3, 44:1–44:9.

Kai Liu, William K. Cheung, and Jiming Liu. 2015. Detecting multiple stochastic network motifs in network data. *Knowledge and Information Systems* 42, 1, 49–74.

T. Liu and D. Tao. 2015. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP, 99, 1.

T. Lyche and K. Morken. 1986. Making the Oslo algorithm more efficient. *SIAM Journal on Numerical Analysis* 23, 3, 663–675.

Jiayi Ma, Ji Zhao, and Jinwen Tian. 2013. Nonrigid image deformation using moving regularized least squares. *IEEE Signal Processing Letters* 20, 10, 988–991.

Donald W. Marquardt. 1963. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics* 11, 2, 431–441.

Markus Mathias, Rodrigo Benenson, Marco Pedersoli, and Luc Van Gool. 2014. Face detection without bells and whistles. In *Computer Vision—ECCV 2014*. Lecture Notes in Computer Science, Vol. 8692. Springer, 720–735.

Hossam E. Abd El Munim, Amal A. Farag, and Aly A. Farag. 2013. Shape representation and registration in vector implicit spaces: Adopting a closed-form solution in the optimization process. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 3, 763–768.

Tuan Anh Nguyen, Minh Binh Do, Alfonso Gerevini, Ivan Serina, Biplav Srivastava, and Subbarao Kambhampati. 2011. Planning with partial preference models. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'11)*. 1772–1777.

G. M. Nielson. 1993. Scattered data modeling. *IEEE Computer Graphics and Applications* 13, 1, 60–70.

Eng-Jon Ong and R. Bowden. 2011. Robust facial feature tracking using shape-constrained multiresolution-selected linear predictors. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 9, 1844–1859.

Y. Panagakis, M. A. Nicolaou, S. Zafeiriou, and M. Pantic. 2013. Robust canonical time warping for the alignment of grossly corrupted sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'13)*. 540–547.

Patrick Pérez, Michel Gangnet, and Andrew Blake. 2003. Poisson image editing. *ACM Transactions on Graphics* 22, 3, 313–318.

S. Raman and S. Chaudhuri. 2007. A matte-less, variational approach to automatic scene compositing. In *Proceedings of the International Conference on Computer Vision (ICCV'07)*. 1–6.

Shulamit Reches, Meir Kalech, and Philip Hendrix. 2014. A framework for effectively choosing between alternative candidate partners. *ACM Transactions on Intelligent Systems and Technology* 5, 2, 30:1–30:28.

Scott Schaefer, Travis McPhail, and Joe Warren. 2006. Image deformation using moving least squares. *ACM Transactions on Graphics* 25, 3, 533–540.

Kristina Scherbaum, Tobias Ritschel, Matthias Hullin, Thorsten Thormählen, Volker Blanz, and Hans-Peter Seidel. 2011. Computer-suggested facial makeup. *Computer Graphics Forum* 30, 2, 485–492.

Lei Shi, Aryya Gangopadhyay, and Vandana P. Janeja. 2015. STenSr: Spatio-temporal tensor streams for anomaly detection and pattern discovery. *Knowledge and Information Systems* 43, 2, 333–353.

Mingli Song, Dacheng Tao, Chun Chen, Xuelong Li, and Chang Wen Chen. 2010. Color to gray: Visual cue preservation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32, 9, 1537–1552.

Yi Sun, Xiaogang Wang, and Xiaoou Tang. 2013. Deep convolutional network cascade for facial point detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'13)*. 3476–3483.

D. Tao, J. Cheng, M. Song, and X. Lin. 2015a. Manifold ranking-based matrix factorization for saliency detection. *IEEE Transactions on Neural Networks and Learning Systems* PP, 99, 1.

D. Tao, X. Lin, L. Jin, and X. Li. 2015b. Principal component 2-D long short-term memory for font recognition on single Chinese characters. *IEEE Transactions on Cybernetics* PP, 99, 1.

Maxime Taron, Nikos Paragios, and Marie-Pierre Jolly. 2009. Registration with uncertainties and statistical modeling of shapes with variable metric kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 1, 99–113.

Wai-Shun Tong, Chi-Keung Tang, Michael S. Brown, and Ying-Qing Xu. 2007. Example-based cosmetic transfer. In *Proceedings of the Pacific Conference on Computer Graphics and Applications (PG'07)*. 211–218.

L. Torresani, V. Kolmogorov, and C. Rother. 2013. A dual decomposition approach to feature correspondence. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 2, 259–271.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research* 11, 3371–3408.

Daniel Vlasic, Matthew Brand, Hanspeter Pfister, and Jovan Popović. 2005. Face transfer with multilinear models. *ACM Transactions on Graphics* 24, 3, 426–433.

Yu-Shuen Wang, Hui-Chih Lin, Olga Sorkine, and Tong-Yee Lee. 2010. Motion-based video retargeting with optimized crop-and-warp. *ACM Transactions on Graphics* 29, 4, 90:1–90:9.

Yu-Shuen Wang, Feng Liu, Pu-Sheng Hsu, and Tong-Yee Lee. 2013. Spatially and temporally optimized video stabilization. *IEEE Transactions on Visualization and Computer Graphics* 19, 8, 1354–1361.

Xuehan Xiong and F. de la Torre. 2013. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'13)*. 532–539.

C. Xu, D. Tao, and C. Xu. 2015. Multi-view intact space learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP, 99, 1.

Shengliang Xu, Shenghua Bao, Ben Fei, Zhong Su, and Yong Yu. 2008. Exploring folksonomy for personalized search. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'08)*. 155–162.

Fei Yang, L. Bourdev, E. Shechtman, Jue Wang, and D. Metaxas. 2012a. Facial expression editing in video using a temporally-smooth factorization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'12)*. 861–868.

Fei Yang, Eli Shechtman, Jue Wang, Lubomir Bourdev, and Dimitris Metaxas. 2012b. Face morphing using 3D-aware appearance optimization. In *Proceedings of the Graphics Interface Conference (GI'12)*. 93–99.

Fei Yang, Jue Wang, Eli Shechtman, Lubomir Bourdev, and Dimitri Metaxas. 2011. Expression flow for 3D-aware face component transfer. *ACM Transactions on Graphics* 30, 4, 60:1–60:10.

Fuzheng Zhang, Nicholas Jing Yuan, Yingzi Wang, and Xing Xie. 2015. Reconstructing individual mobility from smart card transactions: A collaborative space alignment approach. *Knowledge and Information Systems* 44, 2, 299–323.

Jie Zhang, Shiguang Shan, Meina Kan, and Xilin Chen. 2014. Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment. In *Computer Vision—ECCV 2014*. Lecture Notes in Computer Science, Vol. 8690. Springer, 1–16.

Erjin Zhou, Haoqiang Fan, Zhimin Cao, Yuning Jiang, and Qi Yin. 2013. Extensive facial landmark localization with coarse-to-fine convolutional network cascade. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW'13)*. 386–391.

Received August 2014; revised August 2015; accepted August 2015