Short Papers

Reconstructing 3D Face Model with Associated Expression Deformation from a Single Face Image via Constructing a Low-Dimensional Expression Deformation Manifold

Shu-Fan Wang, *Student Member*, *IEEE*, and Shang-Hong Lai, *Member*, *IEEE*

Abstract—Facial expression modeling is central to facial expression recognition and expression synthesis for facial animation. In this work, we propose a manifold-based 3D face reconstruction approach to estimating the 3D face model and the associated expression deformation from a single face image. With the proposed robust weighted feature map (RWF), we can obtain the dense correspondences between 3D face models and build a nonlinear 3D expression manifold from a large set of 3D facial expression models. Then a Gaussian mixture model in this manifold is learned to represent the distribution of expression deformation. By combining the merits of morphable neutral face model and the low-dimensional expression manifold, a novel algorithm is developed to reconstruct the 3D face geometry as well as the facial deformation from a single face image in an energy minimization framework. Experimental results on simulated and real images are shown to validate the effectiveness and accuracy of the proposed algorithm.

Index Terms—3D face reconstruction, expression modeling, manifold analysis, surface registration.

1 Introduction

Previous works on 3D head modeling from a single face image utilized prior information on 3D head models. However, it is difficult to accurately reconstruct the 3D face model from a single face image with expression since the facial expression induces 3D face model deformation in a complex manner. The main challenge is the coupling of the neutral 3D face model and the 3D deformation due to expression, which leads to geometry ambiguity between the original neutral face and the expression deformation. In this paper, we propose a 3D face model reconstruction system which includes registration and training of 3D face models with expressional deformations as well as the estimation of the 3D face model and the 3D expressional deformation from a single face image.

1.1 Related Work

In order to analyze the deformation of facial expression, correspondences must be established between 3D face models with different expressions. To this aim, it is strongly demanded to automatically obtain the 3D interframe correspondences between large data sets for accurate facial expression analysis. Many existing methods require considerable amounts of additional efforts to establish interframe correspondences, for instance, markers or manually labeled feature landmark points [1], [2], [3]. The applications of these methods are usually restricted to high-resolutional facial expression analysis because of the small motion between adjacent frames.

• *The authors are with the National Tsing Hua University, Hsinchu 30013, Taiwan, ROC. E-mail: lai@cs.nthu.edu.tw.*

Manuscript received 22 Mar. 2010; revised 30 Nov. 2010; accepted 13 Mar. 2011; published online 27 Apr. 2011.
Recommended for acceptance by S. Li.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2010-03-0210.
Digital Object Identifier no. 10.1109/TPAMI.2011.88.

For the nonrigid registration between 3D face surfaces, Zhang et al. [4] proposed tracking high-resolution 3D face models based on optical flow estimation. This work is based on the matching of texture information and relies on the accuracy of optical flow. Furthermore, Wang et al. [5] proposed a hierarchical tracking algorithm to register high-resolution 3D dynamic facial expressions. Their algorithm is based on using a local deformable generic face, which may suffer from the mesh folding or vertex clustering problems as raised in [6]. Basso et al. [7] registered the facial expression data by using a 3D morphable model (3DMM). In their work, the registration result heavily relies on the accuracy of the prebuilt 3DMM. Later, the property of conformal geometry with its applications were represented and discussed in [8]. Based on the idea of surface parameterization, some automatic methods were proposed to reduce the 3D registration problem to a 2D one [9], [6], and find correspondences between dense 3D point data sets. Furthermore, the work in [6] detected the feature corners with peak curvature values and locally tracked the feature points by applying Laplacian filter to the adjacent frames. However, the appearance and facial geometry are very different from individual to individual; thus it is infeasible to directly apply these methods to register 3D face models across subjects. The appearance should be used selectively and adaptively as complementary information for 3D nonrigid face surface registration. On the other hand, most existing databases do not provide 3D facial expression models captured at video speed. Therefore, the assumption of small and local deformation between 3D face surfaces for most previous 3D nonrigid surface registration methods may not be valid in practice.

Since it is important to obtain the correspondences between models automatically and robustly, we develop a novel method for the nonrigid registration of 3D face models within and across individuals.

Model-based statistical techniques have been widely used for robust human face modeling. Most of the previous 3D face reconstruction techniques require more than one face image to achieve satisfactory 3D human face modeling. Previous work on 3D face reconstruction from a single image is to simplify the problem by using a statistical head model as the prior. For example, Blanz and Vetter [10] proposed an algorithm for 3D face model reconstruction by minimizing the discrepancies between the face image and the corresponding image rendered from a morphable 3D head model under a suitable illumination condition. Later, this technique was successfully applied for face recognition [11] and achieved a high recognition rate.

For 3D facial expression analysis, some methods were proposed to analyze facial expression for different applications with the aid of a 3D face model [12], [13]. Blanz et al. [14] extended their previous works [10], [11] to handle facial expression by collecting 35 3D face scans from an individual with different expressions and reanimated the faces in images by linear-subspace analysis. However, they ignored the variations across individuals and the styles of different individuals cannot be well represented with a linear subspace. The expression and light condition retargeting technique proposed in [15] can transfer the expression and illumination from a 3D scan to a reconstructed neutral 3D face model by using the existing facial expression motion field and estimated spherical harmonic light field. Recently, nonlinear embedding methods, such as ISOMAP [16], locally linear embedding [17], global coordinate of local linear method [18], and neighborhood preserving embedding [19] were proposed to handle high-dimensional nonlinear data, which could be more appropriate to model the facial expression variations. Based on this argument, researchers have developed different methods for facial expression analysis and applied them to different applications [20], [21], [22]. Since the manifold analysis can effectively represent continuous deformation due to facial expression, it can be used as

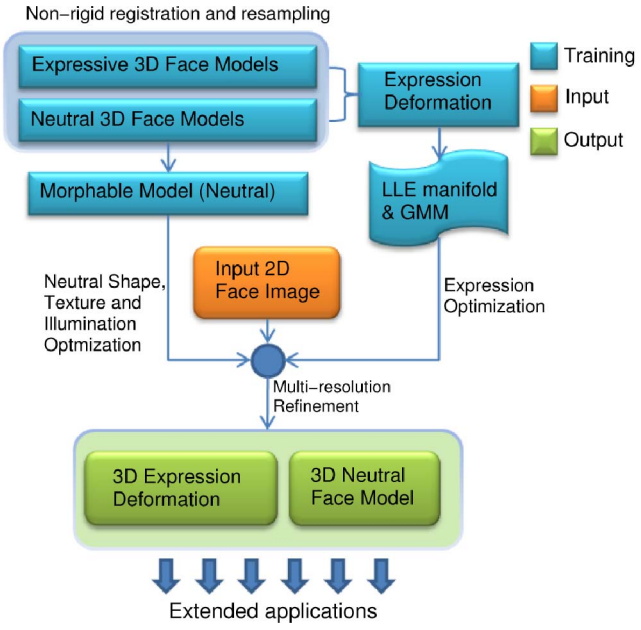


Fig. 1. In the training phase, we build a linear 3D neutral face subspace and a nonlinear 3D expression manifold to represent the shape deformations due to expressions. By combining the merits of morphable neutral face model and the low-dimensional expression manifold, the proposed algorithm reconstructs the 3D face geometry as well as the deformation from a single face image with expression in an multiresolution energy minimization framework.

a prior to transfer expression to a 3D face model by using unified manifold space analysis [5].

1.2 Contribution

In this paper, we propose a system to reconstruct a 3D face model with the associated expressional deformation directly from a single 2D face image with expression based on linear and nonlinear subspace modeling. In summary, the contributions of this proposed approach are listed as follows: First of all, an automatic and robust nonrigid registration algorithm is proposed for registering 3D face models within or across persons. Second, we propose reconstructing the 3D face model with expression deformation from a single image without the assumption on action unit, motion field, or expression style. Third, a probabilistic nonlinear 3D expression manifold is learned from a large set of 3D facial expression models to represent the general deformations due to facial expressions. The manifold reduces the complexity of the reconstruction problem and therefore makes it easier and more robust to track the continuous expression deformation. Finally, we can retarget the reconstructed 3D face model to different expression content or styles for real-world applications.

A global view of our training and reconstruction process is shown in Fig. 1. In Section 2, we first describe how to register the 3D face models within and across persons. In Section 3, we briefly review the morphable model and spherical harmonics for shape and illumination approximation, respectively. In Section 4, we describe how to estimate the probabilistic manifold for 3D facial expression deformations. The main steps of the reconstruction process and parameter estimation are described in Section 5. We demonstrate the effectiveness and accuracy of the proposed algorithm through experiments on simulated and real images in Section 6. The final section concludes this paper.

2 Surface Registration within and across Subjects

In order to establish the morphable model for neutral faces and analyze the deformation of facial expressions, correspondences between 3D models must be determined. The proposed face surface registration algorithm was inspired by the work proposed by Wang et al. [6] which reduced the 3D registration problem to a 2D image registration problem by using a surface parameterization technique. In the following, we will describe how to determine the correspondences between models.

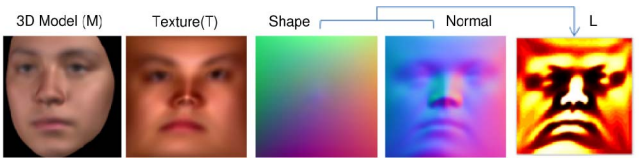


Fig. 2. Local geometry gradient image for intrinsic geometry coding.

2.1 Preprocessing and Coarse Registration

To register 3D face models, we first align the face model to the generic model by the standard Iterative Closest Points (ICP) technique [23] to obtain a coarse registration. We remove the islands and patch the holes on the face surface by using the triangular B-spline technique that was originally proposed by Pfeifle and Seidel [24]. This rigid transformation facilitates the determination of nose tip and orientation in the next step.

2.2 Constrained Surface Parameterization

The parameterization of a surface is a one-to-one mapping from a surface to a planar domain. Based on different metrics, the parameterized domain represents the 3D surface as a simple 2D array of $[x, y, z]$ coordinate values. Similarly, the surface normal and texture color can also be stored as additional 2D images of $[n_x, n_y, n_z]$ and $[c_r, c_g, c_b]$. Geometry images can thus be regarded as two-dimensional arrays storing various surface attributes regularly sampled in the 2D domain.

In this work, we encode the shape, normal vectors, and texture color of a 3D face model into 2D images via Harmonic mapping [6], [25]. A harmonic map is the parameterization of the original 3D surface to the target 2D manifold and preserves the shape conformality well. The harmonic map always exists, one-to-one and onto, and it is robust to the resolution of the 3D faces and to the noise on the surfaces. Moreover, it is invariant to the pose of the source model. Harmonic map can be obtained by minimizing the harmonic energy, and the detail is referred to Pinkall et al.'s paper [26].

In our implementation, boundary condition and nose tip constraint are imposed for the harmonic mapping. We solve the harmonic map by constraining the boundary to a 2D square and the nose tip to the center of the image. Fig. 2 shows an example of the parameterized 2D manifolds for texture, shape, and normal of a 3D face model.

2.3 Robust Weighted Feature Map (RWF)

For surface registration, texture color difference could be the metric when we track the expression of the same individual because the texture is almost the same. However, for different individuals, it is difficult to automatically detect the corresponding features even under the assumption of local deformation [6]. In order to register 3D models of different individuals or with expression variations, we need a general representation to combine the intrinsic texture and geometry characteristics for 3D faces.

Local geometry gradient images [27] is a good choice to represent the intrinsic geometric properties of harmonic maps since it provides several good properties. The value of local gradient at a point in the parametric mesh represents the intrinsic geometry property and is invariant to rotation and translation [27]. For a 3D vertex with normal direction $\mathbf{n}$, we can define an orthogonal local coordinate system formed by a predefined direction and $\mathbf{n}$. Given a parametric shape surface, we transform the gradient into the

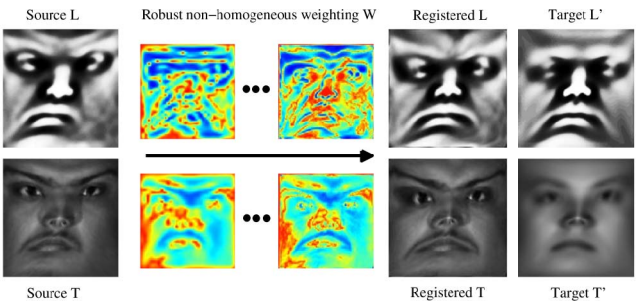


Fig. 3. The robust weighting $\mathbf{W}$ for the texture and local geometry gradient.

corresponding local coordinate system to obtain local gradients of the surface geometry. Fig. 2 shows an example of the local geometry gradient image $\mathbf{L}$ for the generic model.

For a point $\mathbf{p}$ in a parametric surface, the combination of local geometry gradient $\mathbf{L}(\mathbf{p})$ and texture intensity $\mathbf{T}(\mathbf{p})$ form a smooth feature set $\mathbf{RWF}(\mathbf{p}) = [\mathbf{L}(\mathbf{p}), \mathbf{T}(\mathbf{p})]$, which contains the intrinsic geometry and appearance properties. Let $\mathbf{T}'$ and $\mathbf{T}$ represent the parameterized texture intensity images of the target model and the source face model to be registered, and the corresponding local geometry gradient images be denoted by $\mathbf{L}'$ and $\mathbf{L}$, respectively. From robust statistics, the weight associated with the texture for the registration is determined by computing the robust influence function for the discrepancy between $\mathbf{T}(p)$ and $\mathbf{T}'(p)$, given by

$$\mathbf{W}_T(p) = \frac{\sigma_T}{|\mathbf{T}(p) - \mathbf{T}'(p)| + \sigma_T}. \tag{1}$$

Similarly, the weight $\mathbf{W}_L$ for the local geometry gradient is also defined. Note that $\mathbf{T}$, $\mathbf{T}'$, $\mathbf{L}$, and $\mathbf{L}'$ all range from 0 to 1.0, and σ_T and σ_L are the corresponding standard deviations of the differences. In our implementation, the σ_T and σ_L are predefined to 0.1 in the experiments. The values of $\mathbf{W}_T$ and $\mathbf{W}_L$ measure the influences of texture and local geometry gradient information for each pixel in the registration. The robustness function used in (1) maps the difference nonlinearly to $[0, 1]$ and this robust function makes the weighted feature map $\mathbf{RWF}$ more insensitive to noise and outlier. Fig. 3 illustrates an example of registering a 3D face model to a generic model. The weighted feature set $\mathbf{RWF}$ plays an important role because both the geometry and the appearance are important for the guidance of the registration. Notice that the importance of the texture in the mustache area is suppressed during the updating at each iteration since this area can be misleading for the registration.

2.4 RWF-Based Nonrigid Registration

The proposed $\mathbf{RWF}$ provides a unified representation for 3D facial geometry and texture. On the other hand, it is also important to incorporate $\mathbf{RWF}$ into the registration algorithm well with compromise among flexibility, robustness, and efficiency. From these considerations, we extend an intensity-based nonrigid image registration algorithm developed in [28] to an $\mathbf{RWF}$-based surface registration algorithm.

2.4.1 Local Affine Model and Intensity Variations

The source texture $\mathbf{T}$ and target texture $\mathbf{T}'$ can be represented as $f(x, y)$ and $f'(x, y)$. Similarly, the source and target local gradient images $\mathbf{L}$ and $\mathbf{L}'$ can also be denoted as $g(x, y)$ and $g'(x, y)$. In spatial domain, the deformation between images is modeled locally by an affine transform; in order to account for intensity variations, the changes of local contrast and brightness in intensity domain are also incorporated. This affine model for the image registration is similar to that originally proposed in [28]. For each pixel, the affine model can be defined as

$$\begin{aligned} m_7 f(x, y) + m_8 &= f'(m_1 x + m_2 y + m_5, m_3 x + m_4 y + m_6), \\ m_9 g(x, y) + m_{10} &= g'(m_1 x + m_2 y + m_5, m_3 x + m_4 y + m_6), \end{aligned} \tag{2}$$

where $m_1, m_2, \ldots, m_6$ are the parameters of affine transformation which are shared by both the texture image and local geometry images. From the view of 2D image registration, m_7 and m_9 denote the contrast parameter and m_8 and m_{10} denote the change of brightness. Thus, a quadratic energy function can be derived with a first-order truncated Taylor series approximation

$$\begin{aligned} E_f(\mathbf{m}) \approx \sum_{x,y\in\Omega} \big[& m_7 f + m_8 - \big(f' + \big(\mathbf{m}_{125}^T \mathbf{x} - x\big) f'_x \\ & + \big(\mathbf{m}_{346}^T \mathbf{x} - y\big) f'_y\big)\big]^2, \end{aligned} \tag{3}$$

where $\mathbf{m} = (m_1, \ldots, m_{10})^T$, Ω denotes a small neighboring region, $\mathbf{m}_{125} = (m_1, m_2, m_5)^T$, $\mathbf{x} = (x, y, 1)^T$, and f'_x, f'_y denote the spatial derivatives of f'. Based on this formulation, we integrate the idea of $\mathbf{RWF}$ and combine the energy functions E_f and E_g with robust weighting determined by (1), thus leading to the following data energy function:

$$\begin{aligned} E_{data}(\mathbf{m}) \approx \sum_{x,y\in\Omega} \big\{ & \mathbf{W}_T(x, y)\big[k_f - c_f^T \mathbf{m}\big]^2 \\ & + \mathbf{W}_L(x, y)\big[k_g - c_g^T \mathbf{m}\big]^2\big\}, \end{aligned} \tag{4}$$

where

$$\begin{aligned} k_f &= x f'_x + y f'_y - f', \\ k_g &= x g'_x + y g'_y - g', \\ \mathbf{c}_f &= (x f'_x, y f'_x, x f'_y, y f'_y, f'_x, f'_y, -f, -1, 0, 0)^T, \\ \mathbf{c}_g &= (x g'_x, y g'_x, x g'_y, y g'_y, g'_x, g'_y, 0, 0, -g, -1)^T. \end{aligned} \tag{5}$$

2.4.2 Deformation and Intensity Smoothness

To enforce the smoothness over the deformation and intensity variation parameters, similarly to [28], we have the smoothness constraint over all the parameters in $\mathbf{m}$, i.e.,

$$E_{sm}(\mathbf{m}) = \sum_{i=1}^{10} \lambda_i \left[\left(\frac{\partial m_i}{\partial x}\right)^2 + \left(\frac{\partial m_i}{\partial y}\right)^2 \right]. \tag{6}$$

The partial derivatives in (6) can be computed by discrete approximation, and E_{sm} is differentiated with respect to the parameters

$$\frac{dE_{sm}(\mathbf{m})}{d\mathbf{m}} \approx 2\Lambda(\mathbf{m} - \overline{\mathbf{m}}), \tag{7}$$

where Λ is an diagonal matrix formed with predefined constants λs and $\overline{\mathbf{m}}$ denotes the average of $\mathbf{m}$ over a small spatial neighboring region. With the combination of the data term and the smoothness term, the parameters in $\mathbf{m}$ at each pixel can be refined iteratively with the following updating equation:

$$\begin{aligned} \mathbf{m}^{(j+1)} = & \big(\mathbf{W}_T(x, y)\mathbf{c}_f \mathbf{c}_f^T + \mathbf{W}_L(x, y)\mathbf{c}_g \mathbf{c}_g^T + \Lambda\big)^{-1} \\ & \big(\mathbf{W}_T(x, y)\mathbf{c}_f k_f + \mathbf{W}_L(x, y)\mathbf{c}_g k_g + \Lambda\overline{\mathbf{m}}^{(j)}\big). \end{aligned} \tag{8}$$

The proposed method employs a coarse-to-fine optimization strategy by building a Gaussian pyramid for the source and target images. Subsequently, we have dense correspondences between the source and target Harmonic maps. Since we have the parameterized coordinate of the original models, we can obtain the point-wise correspondences between the original source and target models. The correspondences are used to build the morphable model and the expression manifold.

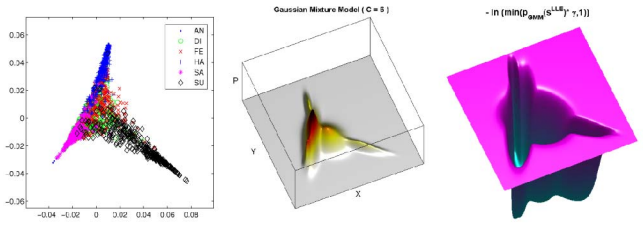


Fig. 4. Low-dimensional manifold representation of expression deformations. (a) The 2D *expression manifold*. (b) The *probability distribution* of the estimated GMM. (c) The modified penalty used as the soft constraint is computed from (b) by using (15).

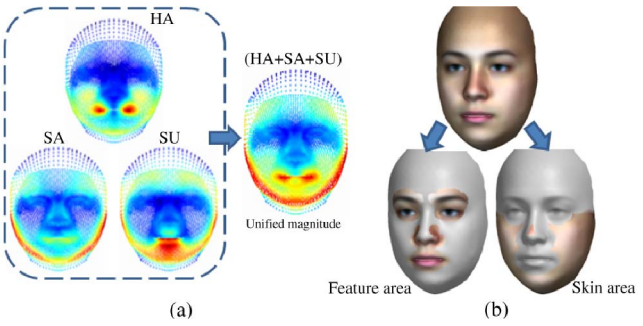


Fig. 5. (a) The *magnitude distribution* of facial deformation under different expressions. (b) The colorful area denotes the corresponding feature and skin area, respectively.

3 Morphable Model and Spherical Harmonics Illumination

The morphable model and spherical harmonic bases are employed in our algorithm to approximate a 3D face model with a small number of parameters and the corresponding face images under different lighting conditions, respectively. In this section, we will briefly describe the morphable model and spherical harmonic lighting bases.

3.1 Morphable Model

The Morphable model provides the prior knowledge of neutral 3D face geometry as well as the associated texture. The geometry can be represented as $S = (x_1, y_1, z_1, \ldots, x_N, y_N, z_N) \in \Re^{3N}$ and the appearance can be represented as a vector $\tau = (r_1, g_1, b_1, \ldots, r_N, g_N, b_N) \in \Re^{3N}$, where N is the total number of vertices in a 3D model. Therefore, the geometry and texture of a 3D face model can be approximated by a mean and a linear combination of several eigenhead basis vectors [10]. In this work, we use the 3D face scans and images from BU-3DFE database [29] as the training faces. Based on the registration algorithm described in Section 2, we can obtain the point-wise correspondences of all the 3D faces for constructing the morphable model.

3.2 Spherical Harmonic Bases

Spherical harmonic bases [30] have been used to approximate the images of a 3D model under a wide variety of lighting conditions. These bases could be determined by the surface normal $\mathbf{n}$ and the albedo $\vec{\lambda}$. The albedo is a measure of how strongly a surface reflects light from light sources. The reflection is related to the surface material on the human face, and the reflection magnitude is positive related to the facial appearance [31]. Therefore, the albedo $\vec{\lambda}$ can be approximated by the texture component of the morphable model $\tau(\vec{\beta})$. As the work described in [31], we can approximate the image of a 3D model under arbitrary illumination conditions with a linear combination of the bases $\mathbf{B}$ as $I_{\text{model}} = \mathbf{B}\ell$, where $\ell \in \Re^9$ is a nine-dimensional weighting vector.

4 Probabilistic Manifold Embedding for Expression Deformations

In this work, we are interested in the expression deformations $\Delta\mathbf{s}_i^{fp}$ obtained by the registered 3D face models with and without expression

$$\Delta\mathbf{s}_i^{fp} = \mathbf{S}_{Ei}^{fp} - \mathbf{S}_{Ni}^{fp}, \tag{9}$$

where $\mathbf{S}_{Ei}^{fp} = \{x_1^E, y_1^E, z_1^E, \ldots x_n^E, y_n^E, z_n^E\} \in \Re^{3n}$ is a set of feature points representing the ith 3D face geometry with facial expression, and similarly, $\mathbf{S}_{Ni}^{fp}$ denotes the set of feature points of the corresponding 3D neutral face. We first conduct an experiment to determine which manifold analysis is suitable for 3D face expression deformation. Due to page limitations, please see the supplementary material, which can be downloaded from https://sites.google.com/site/savan520/home/expression, for more details on the small experiment. In this paper, the coordinate of expression data i for expression j on the manifold is defined as $\mathbf{s}_{ij}^{LLE}$ and the coordinate of zero expression deformation on the manifold is $\mathbf{s}_0^{LLE}$. From our experimental evaluation of several dimension reduction methods, we decide to use the LLE to build a probabilistic model in a low-dimensional manifold for representing the 3D deformation due to facial expression.

4.1 Low-Dimensional Embedding

We employ the LLE [17] to achieve a low-dimensional nonlinear embedding of M expression deformations $\Delta\mathbf{s}_i^{fp}$ obtained by Wang et al. (9). Since we have the correspondences between the models, we can easily select feature points on the generic model and obtain the corresponding features on the other models. The definition of the features is the same as that in BU-3DFE database [29]. The collection of M 3D expression deformations, denoted by $\Delta\mathbf{s}_i^{fp}$ for $i = 1, \ldots, M$, including six different expressions, are embedded to M points in the manifold space. In this work, we settled with the 2D manifold to represent the space of 3D expression variations. Fig. 4a shows the embedded two-dimensional manifold space.

4.2 Probability Distribution in the Expression Manifold

In order to obtain a feasible solution, we need to provide a prior as additional constraints to the 3D face reconstruction system. Since the number of training data is much smaller than the original dimension of the 3D face data, it is not suitable to estimate the prior distribution in the original space. Therefore, we estimate the prior for the 3D face model in the intrinsic low-dimensional space. Gaussian Mixture Model (GMM) is used to approximate the probability distribution of the 3D expression deformation and $p_{GMM}(\mathbf{s}^{LLE})$ represent the probability of the expression $\mathbf{s}^{LLE}$ in the 2D manifold. The colorful dots shown in Fig. 4a indicate the distribution of the manifold and the probability distribution of the estimated GMM in the expression manifold is shown in Fig. 4b.

5 3D Model Reconstruction

In this section, we present a novel algorithm to reconstruct the 3D face model from a single image based on the learned neutral 3D face morphable model and the probabilistic 2D expression manifold model. We apply an iterative scheme to optimize the intra and interdeformation of 3D human face models. To be more robust during the optimization procedure, we first analyze the magnitude of the expression deformation. Take three expressions, for example, the color distribution shown in Fig. 5a depicts the distribution of the corresponding expression deformation and the unified magnitude vector is obtained by calculating the combination of the magnitudes from different expressions. From the above statistics on the deformation magnitudes for different expressions at all locations in the 3D face model, we can determine the

weighting of each node in the 3D face for the morphable model (neutral face) as well as the expression model. This weighting, denoted by w_j^N for the jth node, will be used in the neutral face estimation, which will be described in the following section.

5.1 Initialization

For the initialization of the 3D model, we first estimate the shape parameters by minimizing the geometric distance of the landmark features. The minimization problem is given by

$$\min_{f,\mathbf{R},\mathbf{t},\vec{\alpha}} \sum_{j=1}^{n} w_j^N \|\mathbf{u}_j - (\mathbf{P}f\mathbf{R}\hat{\mathbf{x}}_j(\vec{\alpha}) + \mathbf{t})\|, \tag{10}$$

where $\mathbf{u}_j$ denotes the coordinate of the jth feature point in 2D image, $\mathbf{P}$ is an orthographic projection matrix, f is the scaling factor, $\mathbf{R}$ denotes the 3D rotation matrix, $\mathbf{t}$ is the translation vector, and $\hat{x}_j(\vec{\alpha})$ denotes the jth reconstructed 3D feature point that is determined by the shape parameter vector $\vec{\alpha}$ as $\hat{x}_j = \bar{x}_j + \sum_{l=1}^{m} \alpha_l \mathbf{s}_l^j$.

The function minimization problem given in (10) can be solved by using the Levenberg-Marquart algorithm [32] to find the 3D face shape vector and the pose of the 3D face as the initial solution for the 3D face model. In this step, the 3D neutral face model is initialized and the effect of the deformation from facial expression can be alleviated by using the weighting w_j^N for the neutral face model. Note that the initial $\mathbf{s}^{LLE}$ is set to $\mathbf{s}_0^{LLE}$ on the expression manifold.

5.2 Parameters Optimization

After the initialization step, all of the parameters are iteratively optimized in two steps, which will be described in detail subsequently.

5.2.1 Texture and Illumination Update

To recover the texture and illumination, we need to estimate texture coefficient vector $\vec{\beta}$ and then determine the illumination bases $\mathbf{B}$ and the corresponding spherical harmonic (SH) coefficient vector ℓ. The spherical harmonic bases $\mathbf{B}$ are determined by the surface normal $\mathbf{n}$ and texture intensity $T(\vec{\beta})$. Therefore, with the surface normal $\mathbf{n}$ determined from the current 3D face model, the texture coefficient vector $\vec{\beta}$ and the SH coefficient vector ℓ can be estimated by solving the optimization problem

$$\min_{\beta,\ell} \|\mathbf{I}_{input} - \mathbf{B}(T(\beta), \mathbf{n})\ell\|. \tag{11}$$

According to different reflection properties in the face *feature area* and *skin area*, we define these two areas for more accurate texture and illumination estimation. The face *feature area* and *skin area* are shown in Fig. 5b. Since the texture variation of facial skin is much more consistent than that of facial features, it would be easier to estimate the illumination variation in *skin area*. Therefore, we first estimate the SH coefficient vector ℓ from the *skin area*. Based on the estimated ℓ, the texture coefficient vector $\vec{\beta}$ can be estimated by minimizing the intensity errors for the vertices in the *face feature area* which contains the most texture variations. Since our formulation is only suited for normal lighting conditions, please refer to the previous works in [31] and [33] for strong lighting conditions.

5.2.2 3D Face Shape Update

In this step, the facial deformation is estimated from the photometric approximation with the estimated texture parameters obtained from the previous step. Since we have the statistical models for the facial geometry and deformation, the shape parameters $\vec{\alpha}$, expression parameters $\hat{\mathbf{s}}^{LLE}$, and the head pose vector $\vec{\rho} = \{f, \mathbf{R}, \mathbf{t}\}$ can be estimated by maximizing the posterior probability (MAP) with a given input image $\mathbf{I}_{\text{input}}$ and the texture coefficient vector $\vec{\beta}$. Similarly to [11], we neglect the correlation between these parameters and the posterior probability is written as

$$\begin{aligned} &p(\vec{\alpha}, \vec{\rho}, \hat{\mathbf{s}}^{LLE} | \mathbf{I}_{\text{input}}, \vec{\beta}) \\ &\propto p(\mathbf{I}_{\text{input}} | \vec{\alpha}, \vec{\beta}, \vec{\rho}, \hat{\mathbf{s}}^{LLE}) \cdot p(\vec{\alpha}, \rho, \hat{\mathbf{s}}^{LLE}) \\ &\approx \exp\left(\frac{-\|\mathbf{I}_{\text{input}} - \mathbf{I}_{\exp}(\vec{\alpha}, \vec{\beta}, \vec{\rho}, \hat{\mathbf{s}}^{LLE})\|^2}{2\sigma_I^2}\right) \cdot p(\vec{\alpha}) \cdot p(\vec{\rho}) \cdot p(\hat{\mathbf{s}}^{LLE}), \end{aligned} \tag{12}$$

with

$$\mathbf{I}_{\exp}(\vec{\alpha}, \vec{\beta}, f, \mathbf{R}, \mathbf{t}, \hat{\mathbf{s}}^{LLE}) = \mathbf{I}(f\mathbf{R}(S(\vec{\alpha}) + \psi(\hat{\mathbf{s}}^{LLE})) + \mathbf{t}), \tag{13}$$

where σ_I is the standard deviation of the image synthesis error, and $\psi(\hat{\mathbf{s}}^{LLE}) : \Re^e \rightarrow \Re^{3N}$ is a nonlinear mapping function that maps the estimated $\hat{\mathbf{s}}^{LLE}$ from the LLE manifold with dimension $e = 2$ to the original 3D deformation space with dimension $3N$. Although the LLE manifold is globally nonlinear, it is constructed based on a neighbor-preserving mapping. Therefore, we use the nonlinear mapping function $\psi(\hat{\mathbf{s}}^{LLE}) = \sum_{k \in NB(\hat{\mathbf{s}}^{LLE})} w_k \Delta\mathbf{s}_k$, where $NB(\hat{s}^{LLE})$ is the set of nearest neighbor training data points to $\hat{\mathbf{s}}^{LLE}$ on the expression manifold, $\Delta\mathbf{s}_k$ is the 3D deformation vector for the kth facial expression data in the training data set, and the weight w_k is determined from the neighbors by the same method described in LLE [17]. Maximizing the log-likelihood of the posterior probability in (12) is equivalent to minimizing the following energy function:

$$\begin{aligned} \min\Bigg(&\|\mathbf{I}_{\text{input}} - \mathbf{I}_{\text{input}}(\vec{\alpha}, \vec{\beta}, \rho, \hat{\mathbf{s}}^{LLE})\|^2 \cdot \frac{1}{2\sigma_I^2} \\ &+ \sum_{i=1}^{m} \frac{\alpha_i^2}{2\lambda_i} - \ln p(\rho) - \ln p_{GMM}(\hat{\mathbf{s}}^{LLE}) \Bigg), \end{aligned} \tag{14}$$

where λ_i denotes the ith eigenvalue estimated with PCA for neutral 3D faces. Note that $p(\rho)$ can be simply a constant or modeled with a reasonable prior probability density function to impose constraints on the pose parameters. In real applications, the last term, $-\ln p_{GMM}(\hat{\mathbf{s}}^{LLE})$, in (14) makes the solution close to the ridge of GMM; we slightly modify the last term in (14) as follows:

$$-\ln p_{GMM}(\hat{\mathbf{s}}^{LLE}) \longrightarrow -\ln(\min(p_{GMM}(\hat{\mathbf{s}}^{LLE}) \cdot \gamma, 1.0)), \tag{15}$$

where γ is a predefined constant which can control the bandwidth of the feasible areas. With this modification, the last term is turned into a penalty function (Fig. 4c).

5.3 Algorithm Summary

To reduce the possibility of being trapped into a local minimum solution, we build two morphable models for low and high resolutions and an L-level Gaussian pyramid for each input face image I_{input}. The level of I_{input} is decreased and the resolution of the morphable model is increased along with the iteration. Our algorithm is summarized as follows:

1. Build an L-level Gaussian pyramid for the input image I_{input} and set $\text{level} = \text{L}$. Initially we apply the low-resolution morphable model and set the total number of eigenhead bases M to 5.
2. Initialize the pose and shape coefficients by the feature landmarks. Set the initial $\mathbf{s}^{LLE}$ to the common center of different expressions (Section 5.1).
3. Update the SH coefficient vector and then optimize the texture coefficient vector $\vec{\beta}$ by solving (11) (Section 5.2.1).
4. Update the pose parameters, neutral face shape model parameters $\vec{\alpha}$, and expression parameters $\mathbf{s}^{LLE}$ simultaneously by minimizing the cost function in (14) (Section 5.2.2).
5. Set $level = level - 1$ and $m = m + 5$. Apply the high-resolution morphable model when $level \leq \frac{L}{2}$.
6. Repeat steps 3, 4, and 5 until $level = 0$.

TABLE 1
3D Error of Nonrigid Registrations between Models

3D error	DROP(T)	DROP(L)	CPD[36]	Proposed
Across person non-rigid registrations (neutral)				
Avg	10.0911	3.4215	3.7503	2.6399
Std	7.2039	2.0116	2.0475	1.5186
Within person non-rigid registrations (experssive)				
Avg	4.3899	3.4816	4.4454	2.7470
Std	3.5695	2.0623	2.5372	1.6415

Note that the function minimization problems given in (11) and (14) are solved by using the Levenberg-Marquart algorithm [32] with the gradients computed by finite difference.

6 Experimental Results

In this section, we validate the proposed method by conducting several experiments on different data sets. In the first experiment, we measure the accuracy of the proposed 3D nonrigid registration algorithm presented in Section 2. The 3D face reconstruction accuracy and real image experiments will also be discussed in the following sections. In this paper, we focus on reconstructing 3D face models from a single face image with pose of small out-of-plane rotations ($-40 \leq \theta \leq 40$).

6.1 The Accuracy of Nonrigid Registration

In order to measure the accuracy of 3D nonrigid face surface registration, we conduct several experiments on a large amount of 3D face surfaces from BU-3DFE database [29]. We measure the accuracy of registering the neutral face models to a generic model and the accuracy of registering the expressional face models to the target neutral ones.

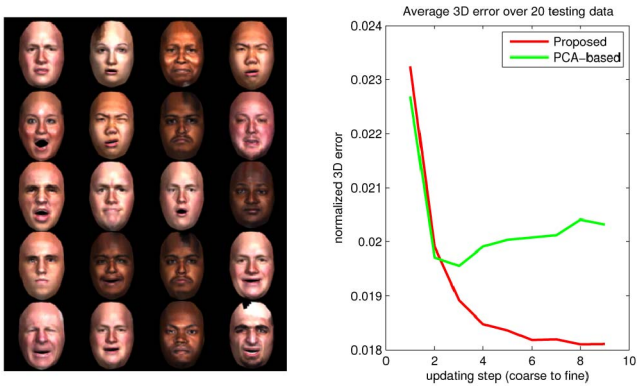
Since we reduce the 3D surface registration problem to a 2D image registration problem via surface parameterization, our results are compared to an image registration method, called DROP [34], [35]. For a fair comparison, we provide the parameterized texture (T) and local geometry gradient image (L) as the input to compare the accuracy of using DROP.

For quantitative assessment of the 3D registration results, we compute the distance between the estimated 3D displacement vectors and the ground-truth at some manually annotated feature points on the source and target 3D face surface models. In this experiment, we also compare our results to a point-based 3D nonrigid registration method called CPD [36], which was shown to be more robust than RPM [37] and ICP [23] under missing or corrupted data. Table 1 shows the statistical measures of the 3D correspondence errors.

6.2 3D Reconstruction Accuracy

To measure the accuracy of the reconstructed 3D face model, we randomly select 20 face models from the BU-3DFE database and these models were not used in the training phase. These 3D face models include different degrees, types, and style of expressions under arbitrary illumination conditions. The testing images are rendered with OpenGL, as depicted in Fig. 6a, and used as the input for the 3D reconstruction.

For quantitative assessment, we calculate the average error from the 3D differences of all vertices and normalize the error by the size, $S_{cube} = \max(width_{cube}, height_{cube}, depth_{cube})$, of the bounding cube containing the 3D face model, i.e., $err = \frac{1}{N \cdot S_{cube}} \sum_{j=1}^{N} \|\hat{v}_j - v_j^g\|$, where $\hat{v}_j$ is the jth 3D vertex of the reconstructed face model, and v_j^g is the jth 3D vertex of the ground truth. In this experiment, the proposed algorithm is also compared with the PCA-based method. For a fair comparison, the PCA model is built from the face models with neutral and six different expressions and the same optimization scheme used in the proposed algorithm is applied to the PCA method. The average errors between the reconstructed 3D model and the ground truth are computed once the face geometry is updated. By optimizing the photometric approximation in the 2D image, the error of our 3D reconstruction is monotonically decreased as the updating step increases. Comparatively, the results of the PCA method shown in Fig. 6 are not stable and the reconstructed 3D face models sometimes are not satisfactory (Fig. 7). The unstable phenomenon of the PCA-based method in Fig. 6b may be caused by the unsuitable linear modeling of the nonlinearly structured expression deformations, which leads to ambiguity in the optimization process.



(a) (b)

Fig. 6. The 3D error analysis for different methods. (a) The input images with known 3D face geometry include different genders, skin color, ages, and expression deformations. (b) The average differences between the reconstructed 3D face model and the ground truth along the updating steps.

6.3 Experiments on Real Images

We used the CMU-PIE data set [38], which contains face images under different poses, illumination, and facial expressions, to test the proposed algorithm. We selected some frames with these two expressions in FG-Net database [39] as the input images in our experiments.

Some of the reconstructed models and the corresponding probability distributions of different expressions are shown in Fig. 7. The bar graphs in Fig. 7 show the estimated probabilities for the expression modeling on the learned manifold, which also demonstrate the accuracy of facial expression estimation. The second and third rows show the reconstructed 3D face model with expressions under a novel view and the synthesized images after expression removal by using the proposed method. The bottom row shows the results of the PCA-based method, which cannot provide satisfactory 3D face model reconstruction.

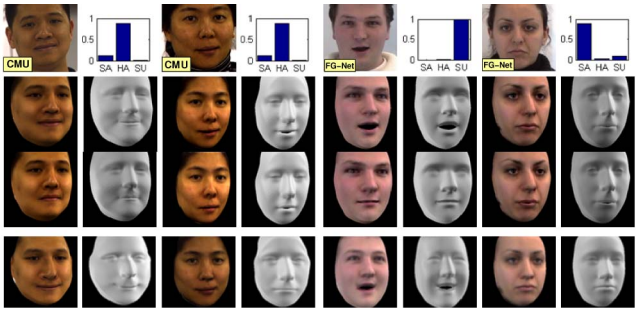


Fig. 7. 3D expression reconstruction from a single real image: Tthe first row shows the *input images* and the bar graphs of the estimated *probabilities* for the expression modeling on the learned manifold. The second and third rows represent our results, including the final *reconstructed expressive face models* and those after *expression removal*. The bottom row shows the results from the *PCA-based* method.

The experiments were all conducted on an X4 3.4G PC with 4,096 MB RAM and the proposed method was implemented in matlab with OpenGL for displaying. In our unoptimized implementation, the reconstruction process of a 3D face took about 5-7 minutes.

7 Conclusion and Discussion

In this paper, we proposed a novel algorithm for reconstructing the 3D face model and the associated 3D expression deformation from a single expressional face image. The nonlinear expression deformation manifold and the linear morphable neutral face model are integrated in an optimization framework to solve this problem. With the combination of spherical harmonics, the texture and illumination condition can be modeled more accurately, thus improving the 3D shape optimization. Based on this framework, we can include more different types of expression deformations into the training data to achieve 3D modeling or retargeting for more general facial expression images.

Since so far we observed that LLE performs well for 3D facial deformations, it is valuable to keep searching methods which can provide better properties for expression modeling. In this paper, the statistical learning approach for 3D face reconstruction makes it difficult to recover the structure of facial expression detail and this is also the limitation of the proposed algorithm. In the future, we plan to investigate better manifold learning techniques for 3D facial expression deformation and further improve the reconstruction of facial expression details for more real-world applications.

Acknowledgments

This research was partially supported by the National Science Council, Taiwan, under grant 98-2221-E-007-089-MY3.

References

[1] B. Allen, B. Curless, and Z. Popovic, "The Space of Human Body Shapes: Reconstruction and Parameterization from Range Scans," *ACM Trans. Graphics,* vol. 22, pp. 587-594, 2003.
[2] B. Guenter, C. Grimm, D. Wood, H. Malvar, and F. Pighin, "Making Faces," *ACM Trans. Graphics,* pp. 55-66, 1998.
[3] J.Y. Noh and U. Neumann, "Abstract Expression Cloning," *Proc. ACM SIGGRAPH,* 2001.
[4] L. Zhang, N. Snavely, B. Curless, and S.M. Seitz, "Spacetime Faces: High-Resolution Capture for Modeling and Animation," *ACM Trans. Graphics,* vol. 23, pp. 548-558, Aug. 2004.
[5] Y. Wang, X. Huang, C.-S. Lee, S. Zhang, Z. Li, D. Samaras, D. Metaxas, A. Elgammal, and P. Huang, "High Resolution Acquisition, Learning and Transfer of Dynamic 3-D Facial Expressions," *Proc. EUROGRAPHICS,* pp. 677-686, 2004.
[6] Y. Wang, M. Gupta, S. Zhang, S. Wang, X. Gu, D. Samaras, and P. Huang, "High Resolution Tracking of Non-Rigid Motion of Densely Sampled 3D Data Using Harmonic Maps," *Int'l J. Computer Vision,* vol. 76, no. 3, pp. 283-300, Mar. 2008.
[7] C. Basso, P. Paysan, and T. Vetter, "Registration of Expressions Data Using a 3D Morphable Model," *Proc. Seventh Int'l Conf. Automatic Face and Gesture Recognition,* pp. 205-210, 2006.
[8] S. Wang, M. Jin, and X.D. Gu, "Conformal Geometry and Its Applications on 3D Shape Matching, Recognition, and Stitching," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 7, pp. 1209-1220, July 2007.
[9] S. Wang, X. Gu, and H. Qin, "Automatic Non-Rigid Registration of 3D Dynamic Data for Facial Expression Synthesis and Transfer," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* June 2008.
[10] V. Blanz and T. Vetter, "A Morphable Model for the Synthesis of 3D-Faces," *SIGGRAPH '99: Proc. 26th Ann. Conf. Computer Graphics and Interactive Techniques,* 1999.
[11] V. Blanz and T. Vetter, "Face Recognition Based on Fitting a 3D Morphable Model," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 25, no. 9, pp. 1063-1074, Sept. 2003.
[12] Z. Wen and T. Huang, "Capturing Subtle Facial Motions in 3D Face Tracking," *Proc. IEEE Int'l Conf. Computer Vision,* 2003.
[13] L. Zalewski and S. Gong, "Synthesis and Recognition of Facial Expressions in Virtual 3D Views," *Proc. IEEE Sixth Int'l Conf. Automatic Face and Gesture Recognition,* 2004.
[14] V. Blanz, C. Basso, T. Poggio, and T. Vetter, "Reanimating Faces in Images and Video," *Proc. 24th Ann. Conf. European Assoc. Computer Graphics,* 2003.
[15] L. Zhang, Y. Wang, S. Wang, D. Samaras, S. Zhang, and P. Huang, "Image-Driven Re-Targeting and Relighting of Facial Expressions," *Proc. Computer Graphics Int'l,* 2005.
[16] J. Tenenbaum, V. de Silva, and J. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction," *Science,* vol. 290, pp. 2319-2323, 2000.
[17] S. Roweis and L. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science,* vol. 290, pp. 2323-2326, 2000.
[18] S. Roweis, L. Saul, and G. Hinton, "Global Coordination of Local Linear Models," *Neural Information Processing Systems,* vol. 14, pp. 889-896, 2001.
[19] X. He, D. Cai, S. Yan, and H.-J. Zhang, "Neighborhood Preserving Embedding," *Proc. IEEE 10th Int'l Conf. Computer Vision,* pp. 1208-1213, 2005.
[20] Y. Chang, C. Hu, and M. Turk, "Manifold of Facial Expression," *Proc. IEEE Int'l Workshop Analysis and Modeling of Faces and Gestures,* 2003.
[21] C. Hu, Y. Chang, and M. Turk, "Probabilistic Expression Analysis on Manifolds," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2004.
[22] C. Hu, Y. Chang, R. Feris, and M. Turk, "Manifold Based Analysis of Facial Expression," *Proc. IEEE Workshop Face Processing in Video,* 2004.
[23] P.J. Besl and H.D. Mckay, "A Method for Registration of 3-D Shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 14, no. 2, pp. 239-256, Feb. 1992.
[24] R. Pfeifle and H.-P. Seidel, "Triangular B-Splines for Blending and Filling of Polygonal Holes," *Proc. Conf. Graphics Interface,* pp. 186-193, 1996.
[25] D. Zhang and M. Hebert, "Harmonic Maps and Their Applications in Surface Matching," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* vol. 2, 1999.
[26] U. Pinkall, S.D. Juni, and K. Polthier, "Computing Discrete Minimal Surfaces and Their Conjugates," *Experimental Math.,* vol. 2, pp. 15-36, 1993.
[27] M.X. Nguyen, X. Yuan, and B. Chen, "Geometry Completion and Detail Generation by Texture Synthesis," *The Visual Computer,* vol. 21, nos. 8-10, pp. 669-678, 2005.
[28] S. Periaswamy and H. Farid, "Elastic Registration in the Presence of Intensity Variations," *IEEE Trans. Medical Imaging,* vol. 22, no. 7, pp. 865-874, July 2003.
[29] L. Yin, X. Wei, Y. Sun, J. Wang, and M.J. Rosato, "A 3D Facial Expression Database for Facial Behavior Research," *Proc. Seventh Int'l Conf. Automatic Face and Gesture Recognition,* pp. 211-216, 2006.
[30] R. Basri and D. Jacobs, "Lambertian Reflectance and Linear Subspaces," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 25, no. 2, pp. 218-233, Feb. 2003.
[31] L. Zhang and D. Samaras, "Face Recognition from a Single Training Image under Arbitrary Unknown Lighting Using Spherical Harmonics," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 3, pp. 351-363, Mar. 2006.
[32] K. Levenberg, "A Method for the Solution of Certain Non-Linear Problems in Least Squares," *Proc. Quarterly of Applied Math.,* vol. 2, no. 2, pp. 164-168, 1944.
[33] Y. Wang, L. Zhang, Z. Liu, G. Hua, Z. Wen, Z. Zhang, and D. Samaras, "Face Relighting from a Single Image under Arbitrary Unknown Lighting Conditions," *IEEE Trans. Pattern Analysis Machine Intelligence,* vol. 31, no. 11, pp. 1968-1984, Nov. 2009.
[34] B. Glocker, N. Komodakis, G. Tziritas, N. Navab, and N. Paragios, "Dense Image Registration through MRFs and Efficient Linear Programming," *Medical Image Analysis,* vol. 12, no. 6, pp. 731-741, 2008.
[35] N. Komodakis, G. Tziritas, and N. Paragios, "Fast, Approximately Optimal Solutions for Single and Dynamic MRFs," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1-8, 2007.
[36] A. Myronenko, X.B. Song, and M.Á. Carreira-Perpiñán, "Non-Rigid Point Set Registration: Coherent Point Drift," *Proc. Neural Information Processing Systems,* pp. 1009-1016, 2006.
[37] H. Chui and A. Rangarajan, "A New Algorithm for Non-Rigid Point Matching," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 44-51, 2000.
[38] T. Sim, S. Baker, and M. Bsat, "The CMU Pose, Illumination, and Expression Database," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 25, no. 12, pp. 1615-1618, Dec. 2003.
[39] F. Wallhoff, *Facial Expressions and Emotion Database,* FG-Net Database, 2006.