A Fast 2D Shape Recovery Approach by Fusing Features and Appearance

Jianke Zhu, *Student Member*, *IEEE*, Michael R. Lyu, *Fellow*, *IEEE*, and Thomas S. Huang, *Life Fellow*, *IEEE*

Abstract—In this paper, we present a fusion approach to solve the nonrigid shape recovery problem, which takes advantage of both the appearance information and the local features. We have two major contributions. First, we propose a novel progressive finite Newton optimization scheme for the feature-based nonrigid surface detection problem, which is reduced to only solving a set of linear equations. The key is to formulate the nonrigid surface detection as an unconstrained quadratic optimization problem that has a closed-form solution for a given set of observations. Second, we propose a deformable Lucas-Kanade algorithm that triangulates the template image into small patches and constrains the deformation through the second-order derivatives of the mesh vertices. We formulate it into a sparse regularized least squares problem, which is able to reduce the computational cost and the memory requirement. The inverse compositional algorithm is applied to efficiently solve the optimization problem. We have conducted extensive experiments for performance evaluation on various environments, whose promising results show that the proposed algorithm is both efficient and effective.

Index Terms—Image processing and computer vision, nonrigid detection, real-time deformable registration, nonrigid augmented reality, medical image registration.

1 Introduction

Recovering nonrigid shapes is an interesting and beneficial research problem for computer vision and image analysis [1], [2], [3]. An effective nonrigid shape recovery technique can be applied in a variety of applications for digital entertainment, medical imaging [2], and augmented reality, such as the retexturing of images and videos [4], [5], [6], [7].

Nonrigid shape recovery can usually be regarded as the problem of recovering the explicit surface with a few deformation parameters. In contrast to nonrigid shape recovery, nonrigid surface detection [8] does not require any initialization or a priori pose information. Furthermore, the goal of nonrigid surface detection is to extract the deformable shape's structure from an input image and find out the correct correspondences from noisy data automatically.

Many applications have been investigated for deformable object tracking [9], [10], [11] and registration, such as face tracking and modeling [12], [13], [14], [15], and also more generic and more deformable objects [2]. The major problem of these methods is that they tend to be computationally expensive and mainly aim at object recognition and image segmentation tasks rather than nonrigid shape recovery. However, a real-time and automated solution has recently been proposed [8], [16], which takes advantage of an iterative semi-implicit optimization scheme.

Since the nonrigid shape is usually highly dynamic and represented by many deformation parameters, the nonrigid shape recovery problem is far more complex than the rigid object detection. Moreover, it requires a sufficient number of correct correspondences in order to obtain high registration accuracy. Therefore, it is difficult to directly employ a robust estimator used in the rigid object pose estimation, such as RANSAC [17] or the Hough transform [18], to remove the spurious matches for nonrigid surface detection. An alternative strategy is to iteratively solve for both the correspondence and the transformation [19], [20]. However, these methods tend to be computationally expensive and few of them can be applied to point sets extracted from real images; an exception is the most recent part-based approach [21].

Most of the current nonrigid shape recovery methods can be divided into two categories. The first is dependent on local feature correspondences [8], [22]. The second is based on the appearance, which directly minimizes the residual image between the synthesized template image and the input image [12], [13], [23]. As for the feature-based methods, it is difficult to guarantee the registration accuracy in regions lacking texture. On the other hand, the appearance-based approach can exploit more of the texture information and therefore achieves better registration accuracy. However, it tends to be computationally expensive and requires good initialization to avoid the local optima, and only a few automated solutions have been proposed in the literature. Since both the feature and appearance-based methods have limitations, there is a need

- *J. Zhu and M.R. Lyu are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong. E-mail: {jkzhu, lyu}@cse.cuhk.edu.hk.*
- *T.S. Huang is with the Beckman Institute, University of Illinois at Urbana-Champaign, 405 N. Mathews Ave., Urbana, IL 61801. E-mail: huang@ifp.uiuc.edu.*

Manuscript received 11 Sept. 2007; revised 9 Feb. 2008; accepted 28 May 2008; published online 4 June 2008.
Recommended for acceptance by P. Torr.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2007-09-0574.
Digital Object Identifier no. 10.1109/TPAMI.2008.151.

0162-8828/09/$25.00 © 2009 IEEE Published by the IEEE Computer Society

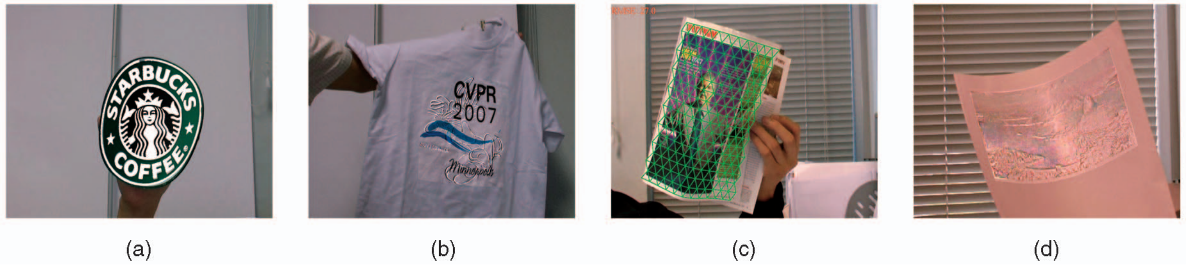


Fig. 1. Recovering nonrigid shapes in real-time video. (a) The contour is overlaid on the Starbucks pad. (b) T-shirt. (c) The cover of a magazine. (d) A piece of paper.

for an automated method that can make use of both the appearance information and the local features.

In this paper, we propose a novel automated approach to efficiently handle the nonrigid shape recovery problem, as shown in Fig. 1. Also, the proposed fusion approach is able to take advantage of both the appearance information and the local features.

Our first major contribution is the proposed progressive finite Newton optimization scheme for nonrigid surface detection, which has the advantage of solving only a fixed number of linear equations. The previous method [8] is currently generally accepted as the most effective method of solving this kind of problem. It employs an implicit iterative scheme for the first-order partial differential equation; however, this requires a large number of iterations to solve the problem and remove the outliers simultaneously. We tackle this critical problem from two angles. First, the nonrigid surface detection is formulated as an unconstrained quadratic optimization problem, which has a closed-form solution for a given set of observations. Thus, it can be efficiently solved through LU factorization. Then, a progressive sample [24] scheme is employed to initialize the optimization scheme, which can decrease the number of trials significantly. Therefore, the present approach requires much fewer iterations than the semi-implicit iterative optimization scheme [16].

Our second contribution is the proposed deformable Lucas-Kanade algorithm, which triangulates the template image into small patches and preserves the regularity of the mesh through the second-order derivatives of the mesh vertices. Moreover, the optimization of our proposed deformable Lucas-Kanade algorithm is formulated into a sparse regularized least squares problem, which is able to reduce the computational cost and the memory requirement. The inverse compositional algorithm [25] is applied to efficiently solve the optimization problem. Furthermore, we solve the optimization for our fusion approach with a modified deformable Lucas-Kanade algorithm.

The rest of this paper is organized as follows: Section 2 reviews the previous approaches employed for the nonrigid surface detection and recovery. In Section 3, we present the proposed fusing features and appearance approach for nonrigid shape recovery. Section 4 provides the details of our experimental implementation and describes our experimental results. We discuss limitations and future work in Section 5. Section 6 sets out our conclusion.

2 Related Work

Considerable research effort has been devoted to nonrigid shape recovery problems in the computer vision and image analysis domain [1], [2], [7], [12]. However, only a few approaches are automatic and can achieve real-time results.

In a recent study, the repeating properties of a near regular texture were exploited to track new texture tiles in video frames [26]. Well-designed markers widely used in motion capture are also applied to recover the structure of a nonrigid surface, such as cloth and paper [6], [7]. As these methods rely on the physical markers, they require the placing of predefined patterns on the target surface. Nevertheless, they are capable of high accuracy.

In fact, a large number of the appearance-based methods [12], [15], [23] can be viewed as extensions of the original Lucas-Kanade algorithm [25], which has been one of the most widely used techniques in computer vision. These approaches directly minimize the residual image between the input image and the synthesized model image [12], [27]. An inverse compositional method [25] has recently been proposed to efficiently solve the optimization problem in the Lucas-Kanade algorithm, reducing the computational cost by precomputing the Hessian matrix. In [28], a feature-driven method is described to make use of the compositional algorithms for the parametric warps. In addition, optical flow information [2], [15] can be incorporated into the optimization scheme to obtain better results. The major limitation of these methods is that they tend to become stuck at a local minimum and hence require good initialization.

On the other hand, feature-based methods [8], [19], [29] try to find out the transformation from the correspondences built by feature matching methods. Thus, these methods can benefit from the recent advances in the feature detection and matching. In [8], [16], Pilet et al. proposed an iterative approach to attack the nonrigid surface detection problem. Physical constraints based on the Finite Element Model [3] are employed for regularization. A semi-implicit iterative scheme is proposed to solve the optimization problem.

The feature matching algorithm plays a very important role for the feature-based nonrigid surface recovery method. Recently, several sophisticated feature descriptors [30], [31] have been proposed to handle the wide-baseline matching problem, including images with large deformation [32]. In addition, machine learning methods such as random classification trees [33] are also employed to find the point correspondences. These methods can take

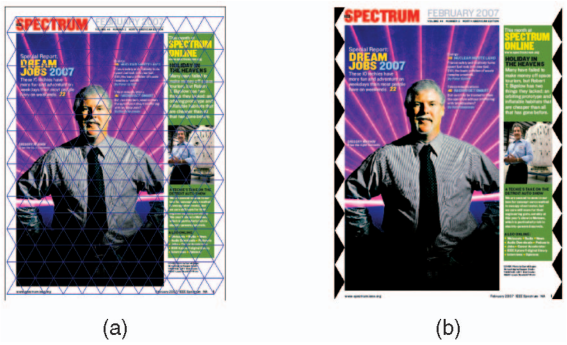


Fig. 2. (a) The mesh model with 216 vertices and 374 triangles. (b) The reference image size of 403 × 516. (a) Model mesh $\mathbf{s}_0$. (b) Reference image.

advantage of shifting part of the computational load from the matching phase to the training phase.

It is more complex to handle a large amount of deformation parameters for detecting the nonrigid surface rather than only a few pose parameters used in rigid object detection. Therefore, there are several challenges when applying conventional robust estimators such as RANSAC and M-estimator for the nonrigid surface detection task. One is the lack of a concise function that can estimate the deformed mesh from the correspondences directly. Obviously, the semi-implicit iterative approach [8] is not efficient enough to deal with this problem. Another challenge is that the RANSAC-based approach requires a large number of trials. To the best of our knowledge, there is still a lack of criteria for selecting the number of samples for each trial in nonrigid surface detection. We tackle the initialization problem through a modified RANSAC method. The key is to draw from progressively larger sets of top-ranked correspondences [24]. Thus, our progressive sample scheme affords large computational savings and the conventional robust estimator can be engaged for initializing the nonrigid surface detection.

3 Fusing Features and Appearance

3.1 Overview

In this section, we describe the fusion approach to dealing with the nonrigid shape recovery, which takes advantage of both the local features and appearance information. For tackling the challenges, a 2D nonrigid shape model is introduced. We formulate the proposed algorithm into an optimization problem that minimizes the correspondence error, the texture difference, and the surface energy. The key of our fusion approach is to solve this problem in the following. First, we present a progressive finite Newton method that employs the feature correspondences to detect the nonrigid surface. Then, we describe a novel deformable Lucas-Kanade algorithm to handle the appearance error. Based on these two algorithms, the optimization scheme for our fusion approach is formulated.

3.2 Mesh Model

The nonrigid shape can be explicitly represented by triangulated meshes. As shown in Fig. 2a, we employ a triangulated 2D mesh with N hexagonally connected vertices, which are formed into a shape vector $\mathbf{s}$ as follows:

$$\begin{aligned}\mathbf{s} &= [\mathbf{s_x} \quad \mathbf{s_y}]^\top \\ &= [x_1 \quad x_2 \quad \dots \quad x_N \quad y_1 \quad y_2 \quad \dots \quad y_N]^\top,\end{aligned}$$

where $\mathbf{s_x}$ and $\mathbf{s_y}$ are the vectors of the coordinates of mesh vertices. Instead of treating the template image as a whole block, as in [34], [35], we employ this 2D deformable mesh model to triangulate it into small patches, as shown in Fig. 2a. Then, the mesh associated with the model image is defined as the reference mesh $\mathbf{s}_0$.

We assume that a point $\mathbf{m}$ lies in a triangle whose three vertices' coordinates are (x_i, y_i), (x_j, y_j), and (x_k, y_k), respectively, and $\{i, j, k\} \in [1, N]$ is the index of each vertex. The piecewise affine warp function is used to map the image points inside the corresponding triangle into the vertices in the mesh. Thus, the mapping function $W(\mathbf{m}, \mathbf{s})$ is defined as follows:

$$W(\mathbf{m}, \mathbf{s}) = \begin{bmatrix} x_i & x_j & x_k \\ y_i & y_j & y_k \end{bmatrix} [\xi_1 \quad \xi_2 \quad \xi_3]^\top, \tag{1}$$

where (ξ_1, ξ_2, ξ_3) are the barycentric coordinates for the point $\mathbf{m}$. Also, the piecewise affine warp $W(\mathbf{m}, \mathbf{s})$ is used to map the input image into the reference frame $\mathbf{s}_0$. Fig. 2b shows an example of the template image in the reference frame.

Based on this triangulated mesh model, we describe in detail the proposed approach to nonrigid shape recovery in the following.

3.3 Fusing Features and Appearance Approach

3.3.1 *Proposed Algorithm*

The aim of our fusion approach is to make use of both the local features and the appearance information:

- **Local feature correspondences.** A set of correspondences M between the model and the input image can be built through a point matching algorithm. Therefore, a pair of matched points is represented in the form of $\mathbf{m} = \{\mathbf{m}_0, \mathbf{m}_1\} \in M$, where $\mathbf{m}_0$ is defined as the 2D coordinates of a feature point in the training image and $\mathbf{m}_1$ is the coordinates of its match in the input image. Then, the correspondence error term $E_c(\mathbf{s})$ is the sum of the weighted square error residuals for the matched points, which is formulated as follows:

$$E_c(\mathbf{s}) = \sum_{\mathbf{m} \in M} \mathcal{V}(\delta, \sigma), \tag{2}$$

  where $\mathcal{V}(\delta, \sigma)$ is a robust estimator, which is described in Section 3.4.
- **Appearance.** In this paper, we try to handle the appearance error under the Lucas-Kanade framework. The objective of the Lucas-Kanade algorithm is to minimize the sum of the squared errors between the template image T and the input image I warped back onto the coordinate frame of the template. Baker and Matthews [25] have proposed an inverse compositional algorithm that switches the role of the template image T and input image I in the

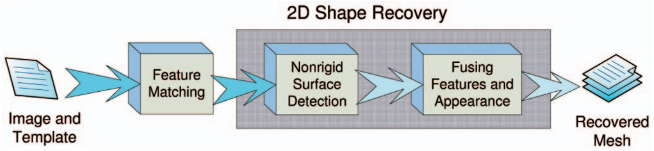


Fig. 3. Overview of our 2D shape recovery algorithm.

computation of the incremental warp. Using this approach, the computational cost can be reduced by precomputing the Hessian matrix. Instead of using the affine transformation or homography, as in [34], [35], we directly employ the parameterization of the mesh model vertices s in this paper. Due to the direct parameterization, $\Delta\mathbf{s}$ is defined as the increments to the mesh vertices. We employ the inverse compositional method to formulate the energy for the appearance E_a. Following the notation in [25], [27], E_a is defined as follows:

$$E_a(\mathbf{s}) = \sum_{\mathbf{x}} [T(W(\mathbf{x};\Delta\mathbf{s})) - I(W(\mathbf{x};\mathbf{s}))]^2. \tag{3}$$

In general, the nonrigid shape recovery problem approximates a 2D mesh with $2N$ free variables, which is usually ill-posed. One effective way to attack this problem is to introduce regularization, which preserves the regularity of a deformable surface. This leads to the following energy function:

$$E(\mathbf{s}) = E_a(\mathbf{s}) + \alpha E_c(\mathbf{s}) + \lambda_r E_r(\mathbf{s}), \tag{4}$$

where α is a weight coefficient and λ_r is a regularization coefficient. The regularization term $E_r(\mathbf{s})$ represents the surface deformation energy. Also, $E_r(\mathbf{s})$, known as internal force in Snakes [36], is composed of the sum of the squared second-order derivatives of the mesh vertex coordinates.

As the mesh is regular, $E_r(\mathbf{s})$ can be formulated through a finite difference:

$$E_r = \mathbf{s}^\top \mathcal{K}\mathbf{s}, \tag{5}$$

where matrix $\mathcal{K} \in R^{2N\times 2N}$ is defined as follows:

$$\mathcal{K} = \begin{bmatrix} K & 0 \\ 0 & K \end{bmatrix},$$

where K is a sparse and banded matrix that is determined by the structure of the explicit mesh model [37].

3.3.2 Optimization Framework

To enable an automated solution, we employ the result of minimizing the feature correspondences error to initialize the optimization for the fusion approach. This is because $E_c(\mathbf{s})$ is independent of the image during the optimization and, so, it can be computed very efficiently. More specifically, the initial result is obtained by the nonrigid surface detection method, which deals with the following energy minimization problem:

$$E_F(\mathbf{s}) = E_c(\mathbf{s}) + \lambda_r E_r(\mathbf{s}). \tag{6}$$

We describe the details of solution for the above optimization problem in Section 3.4.

```
ALGORITHM: PROGRESSIVE NEWTON APPROACH TO NONRIGID SURFACE DETECTION
• Given: mesh model $\mathbf{s}_0$, $\nu$, $\lambda_r$, $\sigma_0$
• Pre-compute: $K$ and $(\xi_1, \xi_2, \xi_3)$ for each keypoint $\mathbf{m}_0$
• Detection: for a given image
      Obtain $M$ by feature matching
      Select active set by modified RANSAC
      While $\sigma > 2$
          Compute $A$ and $\mathbf{b}$
          Solve linear system: Eqn. 16 and Eqn. 17
          Calculate residual error $\delta$ and inlier set $M_1$
          $\sigma = \nu \cdot \sigma$
• Output: mesh vertices $\mathbf{s}$
```

Fig. 4. Progressive Newton approach to nonrigid surface detection.

In this paper, the optimization for our fusion approach is based on the Lucas-Kanade framework. To simplify the formulation, we start from only taking into consideration the texture difference $E_a(\mathbf{s})$. Also, the regularization term $E_r(\mathbf{s})$ is introduced to preserve the surface regularity. Thus, we can obtain the following regularized least squares problem:

$$E_A(\mathbf{s}) = E_a(\mathbf{s}) + \lambda_r E_r(\mathbf{s}). \tag{7}$$

We name this approach the deformable Lucas-Kanade algorithm, which can be used to solve the optimization for our fusion approach with slight modification.

Therefore, the essence of our fusion approach is to first detect the nonrigid shape using feature correspondences and then solve the fusion optimization based on the modified deformable Lucas-Kanade algorithm. We will describe it in detail in the following sections. The overview of our method is shown in Fig. 3, where each step is highlighted using a shaded box.

3.4 Feature-Based Nonrigid Surface Detection

To tackle the nonrigid surface detection problem in (6), we first present our finite Newton formulation. Then, a progressive optimization scheme is proposed to deal with outliers and find out as many correct correspondences as possible. The complete feature-based nonrigid surface detection algorithm is summarized in Fig. 4.

3.4.1 Finite Newton Formulation

In this paper, we employ a robust estimator $\mathcal{V}(\delta,\sigma)$ with compact support size σ. Moreover, δ is the residual error, which is defined as follows:

$$\delta = \mathbf{m}_1 - W_{\mathbf{s}}(\mathbf{m}_0). \tag{8}$$

The robust estimator function $\mathcal{V}(\delta,\sigma)$ that assesses a fixed penalty for residuals larger than a threshold σ is employed in the present work; this approach is relatively insensitive to outliers [38]:

$$\mathcal{V}(\delta,\sigma) = \begin{cases} \frac{\|\delta\|}{\sigma^n}, & M_1 = \{\mathbf{m} \mid \ \|\delta\| \leq \sigma^2\} \\ \sigma^{2-n}, & M_2 = \overline{M_1}, \end{cases} \tag{9}$$

where the set M_1 contains the inlier matches and M_2 is the set of the outliers. In addition, the order n determines the scale of the residual. As shown in Fig. 5, the most correspondences are included when the support σ is large.

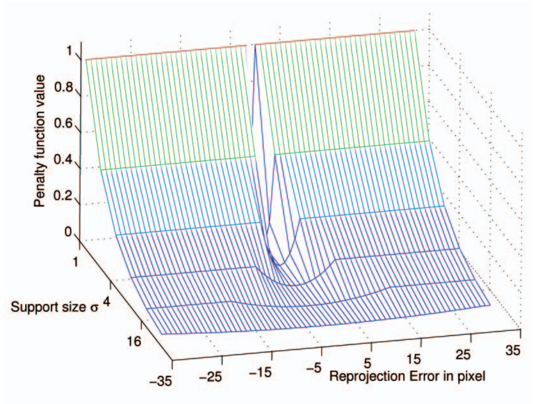


Fig. 5. The robust estimator that assesses a fixed penalty to residuals larger than a threshold σ.

As σ decreases, the robust estimator becomes narrower and more selective.

Since the robust estimator function is not convex, the associated penalty function approximation problem becomes a hard combinational optimization problem. We tackle this problem under the finite Newton optimization framework. An augmented vector $\mathbf{t} \in R^N$ containing the barycentric coordinates is defined as follows:

$$\mathbf{t}_i = \xi_1, \quad \mathbf{t}_j = \xi_2, \quad \mathbf{t}_k = \xi_3,$$

while the remaining elements in the vector $\mathbf{t}$ are all set to zero. Therefore, the residuals for the inlier correspondences can be rewritten as follows:

$$\begin{aligned}\|\delta\| &= (u - \mathbf{t}^\top \mathbf{x})^2 + (v - \mathbf{t}^\top \mathbf{y})^2 \\ &= u^2 + v^2 - 2(u\mathbf{t}^\top \mathbf{x} + v\mathbf{t}^\top \mathbf{y}) + \mathbf{x}^\top \mathbf{t}\mathbf{t}^\top \mathbf{x} + \mathbf{y}^\top \mathbf{t}\mathbf{t}^\top \mathbf{y},\end{aligned}$$

where (u, v) are the coordinates of $\mathbf{m}_1$. Therefore, the error term in (6) turns out to be

$$\begin{aligned}E_c = \sum_{\mathbf{m}\in M_1} \frac{1}{\sigma^n} \Bigg(& u^2 + v^2 - 2\begin{bmatrix} u\mathbf{t} \\ v\mathbf{t} \end{bmatrix}^\top \mathbf{s} \\ & + \mathbf{s}^\top \begin{bmatrix} \mathbf{t}\mathbf{t}^\top & 0 \\ 0 & \mathbf{t}\mathbf{t}^\top \end{bmatrix} \mathbf{s} \Bigg) + q\sigma^{2-n},\end{aligned}$$

where q is the number of outliers.

Let $\mathbf{b} \in R^{2N}$ be defined as follows:

$$\mathbf{b} = \begin{bmatrix} \mathbf{b}_x \\ \mathbf{b}_y \end{bmatrix} = \sum_{\mathbf{m}\in M_1} \frac{1}{\sigma^n} \begin{bmatrix} u\mathbf{t} \\ v\mathbf{t} \end{bmatrix}, \tag{10}$$

and a matrix $A \in R^{N\times N}$ is equal to

$$A = \sum_{\mathbf{m}\in M_1} \frac{1}{\sigma^n} \mathbf{t}\mathbf{t}^\top. \tag{11}$$

Thus, the energy function (6) is formulated into an unconstrained quadratic optimization problem, which can be solved by the modified finite Newton method [39], [40]:

$$\begin{aligned}E_F = \mathbf{s}^\top & \begin{bmatrix} \lambda_r K + A & 0 \\ 0 & \lambda_r K + A \end{bmatrix} \mathbf{s} - 2\mathbf{b}^\top \mathbf{s} \\ & + \sum_{\mathbf{m}\in M_1} \frac{\omega_\mathbf{m}}{\sigma^n}(u^2 + v^2) + q\sigma^{2-n}.\end{aligned}$$

The finite gradient of the energy function E_F with respect to $\mathbf{s}$ can be derived as follows:

$$\nabla = 2\left(\begin{bmatrix} \lambda_r K + A & 0 \\ 0 & \lambda_r K + A \end{bmatrix} \mathbf{s} - \begin{bmatrix} \mathbf{b}_x \\ \mathbf{b}_y \end{bmatrix} \right), \tag{12}$$

and the Hessian [38] can also be computed by

$$H_1 = 2\begin{bmatrix} \lambda_r K + A & 0 \\ 0 & \lambda_r K + A \end{bmatrix}. \tag{13}$$

Thus, the gradient can be rewritten as follows:

$$\nabla = H_1 \mathbf{s} - 2\mathbf{b}. \tag{14}$$

Each Newton step will perform the following operation:

$$\mathbf{s} \leftarrow \mathbf{s} - \gamma H_1^{-1} \nabla, \tag{15}$$

where γ is the step size. We simply set γ equal to one, and no convergence problem occurs in our experiments. Since K is regular, we find that the update of the state vector $\mathbf{s}$ can be computed by the following linear equation:

$$\begin{bmatrix} \lambda_r K + A & 0 \\ 0 & \lambda_r K + A \end{bmatrix} \mathbf{s} = \begin{bmatrix} \mathbf{b}_x \\ \mathbf{b}_y \end{bmatrix}.$$

Moreover, the problem can be further simplified into two linear equations that can be efficiently solved via LU decomposition:

$$\mathbf{s}_\mathbf{x} = (\lambda_r K + A)^{-1} \mathbf{b}_x, \tag{16}$$

$$\mathbf{s}_\mathbf{y} = (\lambda_r K + A)^{-1} \mathbf{b}_y. \tag{17}$$

The overall complexity is thus the complexity of one Newton step. Note that the complexity of one step for the proposed method is same as that in [41].

3.4.2 Progressive Finite Newton Optimization

Generally speaking, the incorrect matches cannot be avoided in the first stage of the matching process where only local image descriptors are compared. We introduce a coarse-to-fine scheme to deal with those outliers. The support σ of robust estimator $\mathcal{V}(\delta, \sigma)$ is progressively decayed at a constant rate ν. Since the derivatives of $\mathcal{V}(\delta, \sigma)$ are inversely proportional to the support σ, the regularization coefficient λ_r is kept constant during the optimization. For each value of σ, the object function E_F is minimized through the finite Newton step and the result is employed as the initial state for the next minimization. The minimization of E_F is directly solved through (16) and (17) for a given initial state and one step is enough to achieve convergence. The optimization procedure stops when σ reaches a value close to the expected precision, which is usually one or two pixels. The algorithm reports a successful detection when the number of inlier matches is above a given threshold. Thus, the whole optimization problem can be solved within a fixed number of steps. This

```
ALGORITHM: DEFORMABLE LUCAS-KANADE ALGORITHM
• Given: $T$, $\mathbf{s}_0$, $\lambda_r$
• Pre-compute: $K$, $\nabla T \frac{\partial W}{\partial \mathbf{s}}$, and the Hessian matrix $H_2$
• Iterate:
        Warp $I$ with $W(\mathbf{x}; \mathbf{s})$ to compute $I(W(\mathbf{x}; \mathbf{s}))$
        Compute the residual image $I(W(\mathbf{x}; \mathbf{s})) - T$
        Compute $\Delta\mathbf{s}$ using Eqn. 19
        Update the shape vector $\mathbf{s} \leftarrow \mathbf{s} - \Delta\mathbf{s}$
• Until $\|\Delta\mathbf{s}\| < threshold$
• Output: mesh vertices $\mathbf{s}$
```

Fig. 6. Deformable Lucas-Kanade algorithm.

is in contrast to the semi-implicit optimization scheme [16], which involves a few iterations for each σ and at least 40 iterations in total to ensure the convergence.

In order to select most of the correspondences into the initial active set and avoid getting stuck at local minima, the initial value of σ is usually set to a sufficiently large value σ_0. However, this requires a fixed initial state. The method is dependent on the object position and needs a few iterations to compensate for the errors generated by the pose variations. In the present work, we solve this problem through a modified RANSAC approach. Taking advantage of our concise finite Newton formulation and closed-form solution, the explicit mesh can be directly estimated from a given set of correspondences. Moreover, we draw from progressively larger sets of top-ranked correspondences, which decreases the number of trials significantly. In the experiments, the sampling process stopped within five trials. In the worst case, such as when an object does not appear in the scene, it still behaves as RANSAC. Therefore, the output of the proposed progressive sample can be employed as the initial state for the finite Newton optimization. Since the result of progressive sample estimation is quite close to the solution, σ is relatively small. Thus, the proposed progressive scheme requires fewer stages and is somewhat invariant to the initial position.

3.5 Deformable Lucas-Kanade Algorithm

Taking consideration of appearance information only, we try to solve the optimization problem in (7) using the inverse compositional method. The deformable Lucas-Kanade algorithm is summarized in Fig. 6.

3.5.1 Deformable Lucas-Kanade Algorithm

The warp update equation can be defined as follows:

$$W(\mathbf{x}) \leftarrow W(\mathbf{x}) \circ W(\mathbf{x}; \Delta\mathbf{s})^{-1}.$$

Performing the first-order Taylor expansion on (7) gives

$$\sum_{\mathbf{x}} \left[T(W(\mathbf{x}; \mathbf{s}_0)) + \nabla T \frac{\partial W}{\partial \mathbf{s}} \Delta\mathbf{s} - I(W(\mathbf{x}; \mathbf{s})) \right]^2 + \lambda_r (\mathbf{s} + \Delta\mathbf{s})^\top \mathcal{K} (\mathbf{s} + \Delta\mathbf{s}), \tag{18}$$

where ∇T is the gradient of the template image evaluated at $W(\mathbf{x}; \mathbf{s}_0)$ and $\frac{\partial W}{\partial \mathbf{s}}$ is the Jacobian of the warp parameters evaluated at $\mathbf{s}$. Note that $\nabla T \frac{\partial W}{\partial \mathbf{s}}$ is the gradient and $\mathbf{s}_0$ is the reference mesh shown in Fig. 2a.

Assuming that $W(\mathbf{x}; \mathbf{s}_0)$ is the identity warp, the gradient of (18) with respect to $\Delta\mathbf{s}$ can be derived as follows:

$$\sum_{\mathbf{x}} \left[\nabla T \frac{\partial W}{\partial \mathbf{s}} \right]^\top \left[T(W(\mathbf{x}; \mathbf{s}_0)) + \nabla T \frac{\partial W}{\partial \mathbf{s}} \Delta\mathbf{s} - I(W(\mathbf{x}; \mathbf{s})) \right] + \lambda_r \mathcal{K} (\mathbf{s} + \Delta\mathbf{s}).$$

As the above gradient vanishes for optimality, this leads to the following closed-form solution:

$$\Delta\mathbf{s} = H_2^{-1} \sum_{\mathbf{x}} \left[\nabla T \frac{\partial W}{\partial \mathbf{s}} \right]^\top [I(W(\mathbf{x}; \mathbf{s})) - T(W(\mathbf{x}; \mathbf{s}_0))] - \lambda_r H_2^{-1} \mathcal{K} \mathbf{s}, \tag{19}$$

where H_2 is the $2N \times 2N$ Hessian matrix:

$$H_2 = \sum_{\mathbf{x}} \left[\nabla T \frac{\partial W}{\partial \mathbf{s}} \right]^\top \left[\nabla T \frac{\partial W}{\partial \mathbf{s}} \right] + \lambda_r \mathcal{K}. \tag{20}$$

Note that the Hessian matrix H_2 is independent of the parameter vector $\mathbf{s}$ and it is kept constant across iterative optimization, can be precomputed, and is also independent of the gradient matrix $\nabla T \frac{\partial W}{\partial \mathbf{s}}$. Therefore, the warp update of the shape parameters $\Delta\mathbf{s}$ can be computed very efficiently.

Since the coordinates of the mesh vertices $\mathbf{s}$ are directly employed as the warp parameter in the deformable Lucas-Kanade algorithm, the computation of the warp inversion becomes much easier than the linear combination model method in Active Appearance Models (AAMs) [27]. Specifically, the shape vector $\mathbf{s}$ is updated by

$$\mathbf{s} \leftarrow \mathbf{s} - \Delta\mathbf{s}. \tag{21}$$

Link to the Lucas-Kanade algorithm. The proposed method can be viewed as a natural extension of the Lucas-Kanade algorithm, which is able to handle the deformations rather than the affine transformation. Since the 2D coordinates of the mesh vertices $\mathbf{s}$ is employed as the parameters in the deformable Lucas-Kanade algorithm, the degree of freedom is increased. This is useful for handling the image alignment when the deformation is large. Furthermore, the efficient optimization methods for the Lucas-Kanade algorithm [25] can also be applied for the proposed method.

Link to Active Appearance Models (AAMs) [27]. The deformable Lucas-Kanade algorithm can be treated as a kind of AAMs. It employs a single training example along with certain physical constraints, while AAMs need to build both texture and shape models to constrain the searching space.

3.5.2 Computing Gradient $\nabla T \frac{\partial W}{\partial \mathbf{s}}$

Recall that the destination of the pixel $\mathbf{x}$ under the piecewise affine warp $W(\mathbf{x}; \mathbf{s})$ depends on the vertices of the mesh $\mathbf{s}$. According to the definition in (1), the Jacobian of the warp $W(\mathbf{x}; \mathbf{s})$ with respect to the mesh vertices $\mathbf{v}(x_i, y_i)$ can be derived as follows:

$$\frac{\partial W}{\partial x_i} = [\xi_1 \quad 0]^\top \qquad \text{and} \qquad \frac{\partial W}{\partial y_i} = [0 \quad \xi_1]^\top.$$

It can easily be found that the nonzero parts of $\frac{\partial W}{\partial x_i}$ and $\frac{\partial W}{\partial y_i}$ are equal. The Jacobians can be illustrated as the images with the same size of reference frame; in fact, each image is the Jacobian with respect to the vertex $\mathbf{v}$. Moreover, it can also be observed that the warp Jacobian is quite sparse, having nonzero values only in the triangles around the vertex $\mathbf{v}$. Next, we compute $\nabla T \frac{\partial W}{\partial \mathbf{s}}$ by multiplying the gradient of the template image with the warp Jacobian matrix.

Remark. Since the dimensionality of the texture is usually very high, the gradient $\nabla T \frac{\partial W}{\partial \mathbf{s}}$ becomes quite a large matrix. Fortunately, both the gradient and the Hessian H_2 are the sparse matrices in the proposed deformable Lucas-Kanade algorithm and this can greatly reduce the computational cost and memory requirement and make the problem tractable. Moreover, this also leads to a sparse regularized least squares problem in (7).

3.5.3 Lighting

In order to minimize the effect of global lighting variation, we apply a scaling a and an offset o to the template image T. Therefore, the energy function of the proposed deformable Lucas-Kanade algorithm can be rewritten as follows:

$$\sum_{\mathbf{x}} [aT(W(\mathbf{x};\Delta\mathbf{s})) + o \cdot \mathbf{1} - I(W(\mathbf{x};\mathbf{s}))]^2 + \lambda_r \mathbf{s}^\top \mathcal{K}\mathbf{s}. \tag{22}$$

Similarly, we can employ an extended inverse compositional algorithm [35], [42] to solve this optimization problem. See the Appendix for the details.

3.6 Fusion Approach Optimization

Based on the deformable Lucas-Kanade algorithm, we describe the optimization scheme for the proposed fusion approach in Section 3.3.

We define a matrix $B \in R^{2N\times 2N}$, which is equal to

$$B = \frac{1}{\sigma^n}\begin{bmatrix} A & 0 \\ 0 & A \end{bmatrix}. \tag{23}$$

Therefore, we can rewrite E_c as follows:

$$E_c = \mathbf{s}^\top B\mathbf{s} - 2\mathbf{b}^\top\mathbf{s} + q\sigma^{2-n} + \sum_{\mathbf{m}\in M_1} \frac{1}{\sigma^n}(u^2+v^2).$$

Performing the first-order Taylor expansion on the energy function (4) gives

$$\begin{aligned}&\sum_{\mathbf{x}}\left[T(W(\mathbf{x};\mathbf{s}_0)) + \nabla T\frac{\partial W}{\partial \mathbf{s}}\Delta\mathbf{s} - I(W(\mathbf{x};\mathbf{s}))\right]^2 \\ &+ \alpha(\mathbf{s}+\Delta\mathbf{s})^\top B(\mathbf{s}+\Delta\mathbf{s}) - 2\alpha\mathbf{b}^\top(\mathbf{s}+\Delta\mathbf{s}) + q\sigma^{2-n} \\ &+ \sum_{\mathbf{m}\in M_1}\frac{1}{\sigma^n}(u^2+v^2) + \lambda_r(\mathbf{s}+\Delta\mathbf{s})^\top\mathcal{K}(\mathbf{s}+\Delta\mathbf{s}).\end{aligned} \tag{24}$$

The solution to this problem is

$$\begin{aligned}\Delta\mathbf{s} = H_3^{-1}\sum_{\mathbf{x}}\left[\nabla T\frac{\partial W}{\partial \mathbf{s}}\right]^\top [I(W(\mathbf{x};\mathbf{s})) - T(W(\mathbf{x};\mathbf{s}_0))] \\ - \alpha H_3^{-1}(B\mathbf{s}-\mathbf{b}) - \lambda_r H_3^{-1}\mathcal{K}\mathbf{s},\end{aligned} \tag{25}$$

where H_3 is the Hessian matrix:

$$H_3 = \sum_{\mathbf{x}}\left[\nabla T\frac{\partial W}{\partial \mathbf{s}}\right]^\top\left[\nabla T\frac{\partial W}{\partial \mathbf{s}}\right] + \alpha B + \lambda_r\mathcal{K}. \tag{26}$$

Again, we can compute the warp update through (21).

In order to reduce the computational cost, we precompute the gradient and part of the Hessian for the deformable Lucas-Kanade algorithm. Since the inlier set is slightly changed in the fusion optimization phase, matrix B can be viewed as a constant. Therefore, we compute the Hessian H_3 once for each input image through (26). The optimization procedure stops when $\|\Delta\mathbf{s}\|$ is close to the given threshold or the number of iterations exceeds the limit.

To tackle the lighting variations, we only need make a slight modification on the method described in Fig. 6. Specifically, we add the initialization step and precompute the matrix B and H_3 for each input image. Furthermore, (25) is employed to compute the update for the shape vector $\mathbf{s}$.

4 Experimental Results

In this section, we discuss the details of our experimental implementation and report the results of performance evaluation on nonrigid shape recovery. First, we perform the various evaluations on the feature-based method. Then, both the deformable Lucas-Kanade algorithm and the fusion approach are tested. Videos illustrating the results can be obtained from the the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2008.151.

4.1 Experimental Setup

In order to register the mesh model conveniently, a model image is acquired when the nonrigid surface contains no deformation. In order to facilitate real-time augmented reality applications, a random-tree-based method [33] is used to build the correspondences between the model image and the input image. We also implemented a semi-implicit iterative method [16], which is regarded as the state-of-the-art approach. All the experiments reported in this paper were carried out on a Pentium 4 3.0 GHz PC with 1 Gbyte RAM and a DV camera was engaged to capture videos.

4.2 Evaluation on Feature-Based Nonrigid Surface Detection

In this section, we show that the proposed feature-based approach is very efficient for real-time tracking. In addition, the same convincing results are obtained for medical image registration, even with missing data.

4.2.1 Parameter Settings

Since the number of free variables for nonrigid surface recovery is usually quite large (even up to 1,000), the sample size of each RANSAC iteration becomes a tricky issue. We compare the performance with different sample sizes. In our experiments, the support σ is empirically set to 30 and λ_r is set to a large value to ensure the regularity of the nonrigid surface. Interestingly, we find that the best sample size is three. This is because the nonrigid surface degenerates into a rigid one and only three points are necessary to determine the position of a rigid surface. Moreover, when the sample size increases, the probability of selecting the inlier data is decreased. Thus, three is the best choice for the sample size.

A set of synthetic data is used to select the parameters, and the reference mesh is manually registered. The

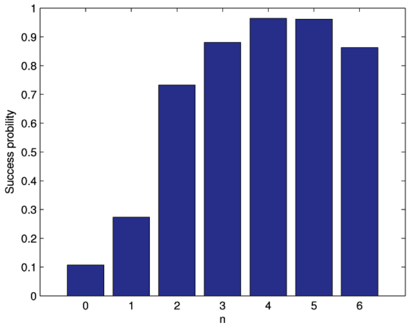


Fig. 7. Probability of success with different orders n of the robust estimator function.

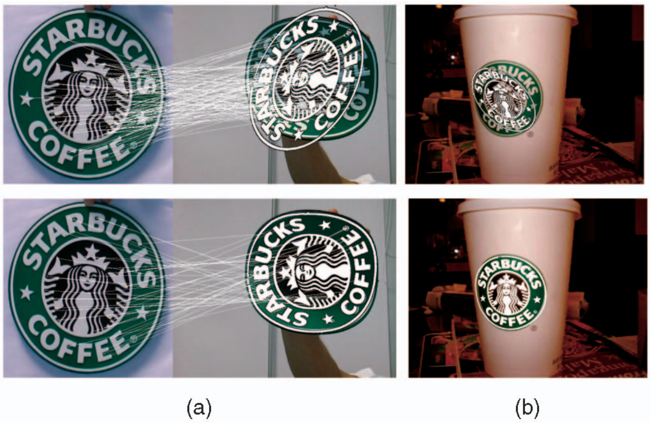


Fig. 8. The first row shows the initialization results using the modified RANSAC method and the second row shows the results. We show the images with inlier matches in (a). (a) Result. (b) Plastic cup.

performance is evaluated by the percentage of mesh vertices within two pixels of those in the reference mesh. The best regularization coefficient is found to be around 3×10^{-4} by grid searching. Similarly, the initial support σ_0 is set to 80 and the decay rate ν is 0.5. Fig. 7 plots the success probability with different orders n of the robust estimator function. Based on these results, n is set to four.

TABLE 1
Computational Time of the Proposed Method at Each Step

Total	Match	Optimization	Iteration	Other
57ms	27ms	14ms	$\sim$ 1.9ms	16ms

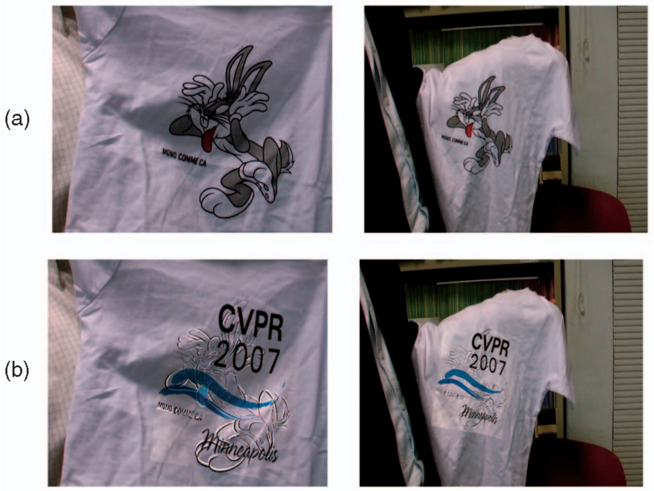


Fig. 9. Retexturing of a shirt print. (a) shows the images captured by a camera. (b) shows the results of replacing the bunny with the CVPR logo.

4.2.2 Computational Efficiency

The complexity of the proposed method is mainly dominated by the order of (16) and (17), which is equal to the number of vertices N in the mesh model. Another important factor is the number of inlier matches, which affects the sparseness of matrix A. This usually differs from one frame to another. For the coffee mat with 120 vertices, as shown in Fig. 8, the proposed method runs at 18 frames per second on real-time video with the size of 720 × 576. As depicted in Table 1, the proposed optimization scheme requires around eight iterations and only takes half of the time of the feature matching algorithm, which is the bottleneck of the whole system. Our implementation[1] of the semi-implicit iterative approach [16] needs around 40 iterations to reach the convergence and runs about 9 frames per second. The improvement is more significant for a high-resolution mesh. Thus, the proposed method requires far less iterations and is efficient for real-time applications. We also conduct the experiments without using the modified RANSAC initialization and start the optimization scheme from a sufficiently large support $\sigma = 1,000$. This requires 11 iterations and the fitting accuracy is worse than the proposed method. In addition, the modified RANSAC initialization can also be used for a semi-implicit method, in which case the number of iterations is reduced to around 25.

4.2.3 Performance of Nonrigid Surface Recovery

We use a coffee mat as the deformable object. As shown in Fig. 8, the proposed method is robust to large deformations and perspective distortion. In practice, the whole process runs at around 18 frames per second. Fig. 9 describes the result of detecting the pattern on a T-shirt, where similar performance is achieved. As another feature-based method, the performance of the proposed method is closely related to the texture of objects. Better results can be obtained for objects with more texture because it is easy to find more correct correspondences than with those lacking texture. Therefore, we present a fusion approach to handle this issue.

4.2.4 Augmented Reality

Once the nonrigid surface is recovered, an immediate application is to retexture an image. In order to obtain realistic results, the texture should be correctly relighted. As suggested in [16], a retextured input image is generated by directly multiplying a blank shaded image, which is the quotient of the input image and the warped reference image. The reference image is acquired when the nonrigid surface is lighted uniformly. Moreover, the quotient image is normalized through multiplying the intensity of white color in the reference image. This relighting procedure is easily done by the GPU and requires only a short OpenGL shading language program and the whole process runs at about 17 frames per second. Fig. 9 shows the results of

[1] We use the same parameter setting as [16]. The convergence condition is set to 0.9995, with at most five iterations for each support value σ.

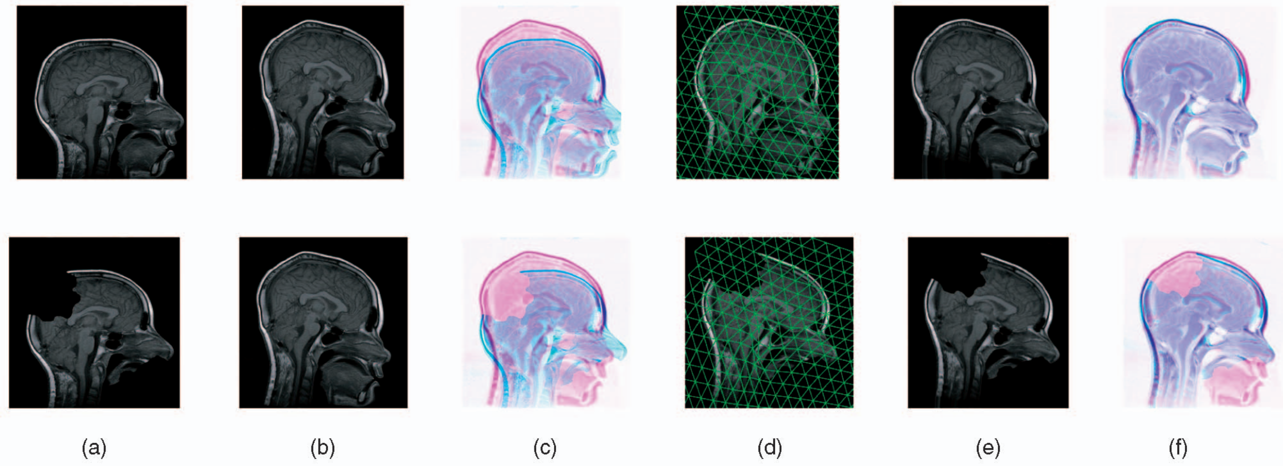


Fig. 10. Applying the proposed method to medical image registration. A pair of sagittal images from two different patients is shown. (a), (b), and (e) are the source, target, and registered source, respectively. (d) is the registered source with the mesh model. (c) and (f) are the overlaid images before and after registration. The second row displays the synthetic example with missing data. (a) Source. (b) Target. (c) Before. (d) Registered. (e) Registered. (f) After.

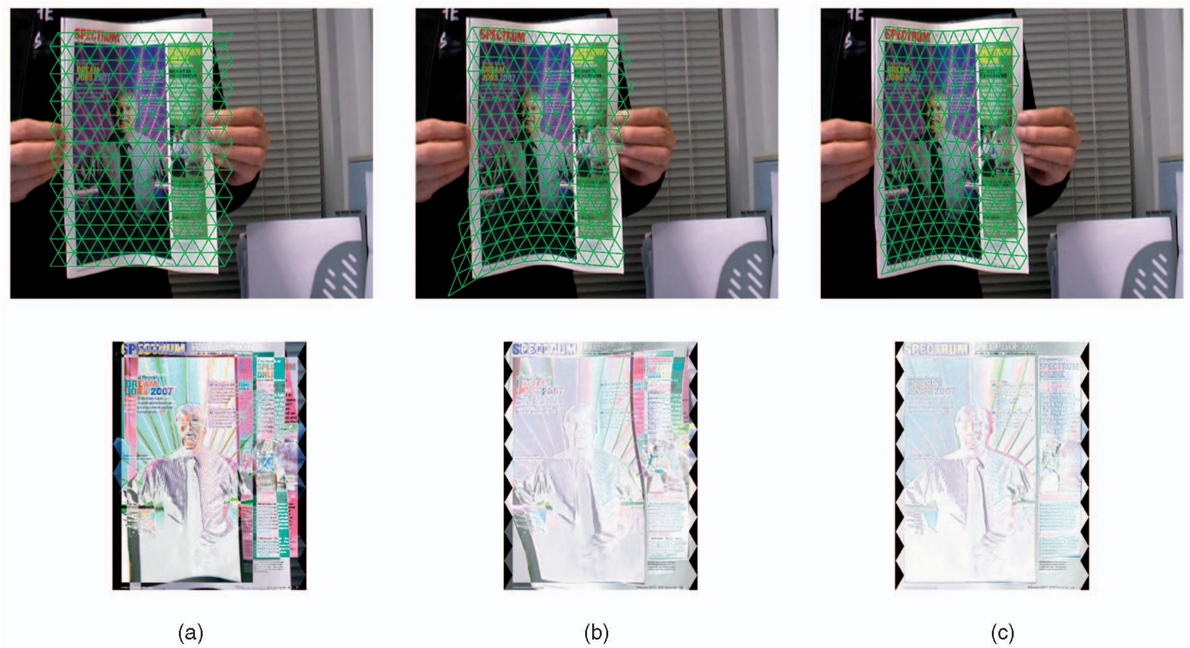


Fig. 11. An example of the deformable Lucas-Kanade fitting to a single image. The first row is the result mesh overlaid on the input image. The second row displays the residual images; the inverted image is used for better illustration. (a) Initialized. (b) After 30 iterations. (c) Converged.

retexturing a T-shirt with a Lambertian surface. It is difficult to estimate a blank shaded image due to dividing near-zero intensity values and the use of an uncontrolled optical sensor. However, the visual effect is that the bunny in the input video is retextured by the CVPR logo.

4.2.5 *Medical Image*

We also evaluate the proposed approach for medical image registration. A pair of sagittal images [43] with the size of 256 × 256 from two different patients are used in the experiments. The source and target images differ in both geometry and intensity. The results are plotted in Fig. 10; it can be seen that the source image is successfully registered. In comparison with the locally affine but globally smooth method [43], which takes about 4 minutes, our proposed method only needs 0.2 second. Moreover, the sparse correspondence-based method can naturally handle the missing data and partial occlusion problem. As shown in the second row of Fig. 10, when the source images come with a region removed, the nonrigid shape can still be recovered.

4.3 Evaluation on the Deformable Lucas-Kanade Algorithm

4.3.1 *Deformable Lucas-Kanade Fitting*

Similarly, the parameters for the deformable Lucas-Kanade algorithm are found by grid searching; and the regularization parameter λ is set to 10^5. Moreover, the texture mapping is efficiently done by OpenGL. In Fig. 11, we include an example of the proposed deformable Lucas-Kanade algorithm fitting to a single image and employ the template image and mesh model, illustrated in Fig. 2.

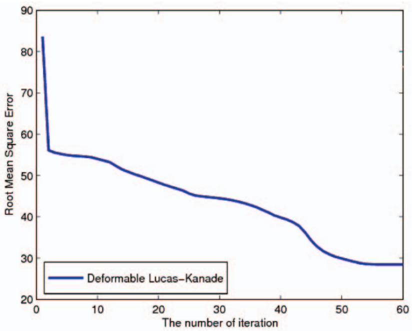


Fig. 12. The RMSE between the template and input images against the number of iterations.

Fig. 11a displays the initial configuration, Fig. 11b shows the result after 30 iterations, and Fig. 11c shows the final converged result after 58 iterations.

We also perform the experiment using the conventional Lucas-Kanade algorithm with the inverse compositional method, using the same initial position as our method. However, this fails to converge in this case due to the large nonaffine deformation. Fig. 12 plots the root-mean-square-error (RMSE) curve for the proposed method. In the case of the deformable Lucas-Kanade algorithm, the RMSE is relatively large (28.5), which is mainly due to the difference between the optical sensor and the printing device. However, we can observe that the mesh was accurately registered on the input image in Fig. 11c. Since the lighting variations are considered in the proposed method, the RMSE dropped rapidly in the first few iterations.

4.3.2 *Computational Efficiency*

The complexity of the proposed method is mainly dominated by the size of the template image and the number of the vertices N in the mesh model. Another factor is the number of inlier feature matches, which affects the sparseness of matrix B. In our experiments, three models are built to perform the evaluation, as summarized in Table 2. The mesh model C_1 is shown in Fig. 2. C_2 is obtained by increasing the edge length, giving fewer mesh vertices than C_1. C_3 is built by reducing both the template image and the mesh size to 75 percent of C_1. We evaluate the computational cost of the proposed method for the nonrigid surface recovery task on real-time videos with the size of 720 × 576. Table 2 summarizes the experimental results on different models. We observe that the dimensionality of the appearance determines the time complexity of the deformable Lucas-Kanade algorithm. Therefore, gray images are easier to track. The number of mesh vertices N has a great influence on the initialization step but a limited impact on the computational time in the optimization.

TABLE 2
Computational Time of the Deformable Lucas-Kanade Algorithm on Different 2D Mesh Models

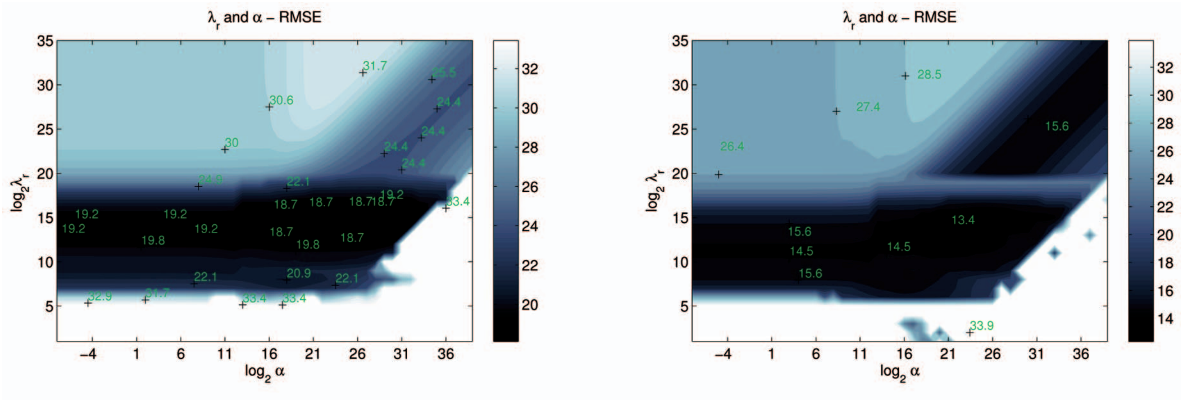
	Vertices	Size	FPS	Initialization	Iteration
C_1 (color)	216	198660	2.4	84ms	$\sim$ 35.7ms
C_1 (gray)	216	198660	4.4	84ms	$\sim$ 16.7ms
C_2 (gray)	96	194670	4.7	68ms	$\sim$ 16.2ms
C_3 (gray)	216	109538	6.1	77ms	$\sim$ 10.2ms

4.4 Fusion Approach

We demonstrate here that the proposed fusion approach is able to be used for nonrigid shape recovery tasks.

4.4.1 *Parameter Settings*

We consider two data sets for searching the parameters. One is the magazine cover, as illustrated in Fig. 15, and the other is a piece of paper in Fig. 16. For each data set, we select 10 testing images containing deformations and then evaluate the proposed fusion approach using different λ_r and α. In the experiment, RMSE is used as the performance measurement. Also, we employ a condition ($\|\Delta \mathbf{s}\| < 2.0$) as the success criteria and set the failure cases with the highest RMSE. Fig. 13 plots the mean RMSE of 10 tests. We can observe that there is a broad area with low RMSE for selecting λ_r and α; the lowest RMSE is found in the middle dark region. Therefore, the local features are useful to improve the fitting accuracy and there is a large range for choosing the weight coefficient α. When α becomes larger, the result is more similar to those from the feature-based method and there is a constant ratio between λ_r and α. It



(a)

(b)

Fig. 13. The Root Mean Square Error (RMSE) with given regularization parameter λ_r and weight coefficient α. Two sets of data are used for evaluation. (a) Magazine cover. (b) Paper.

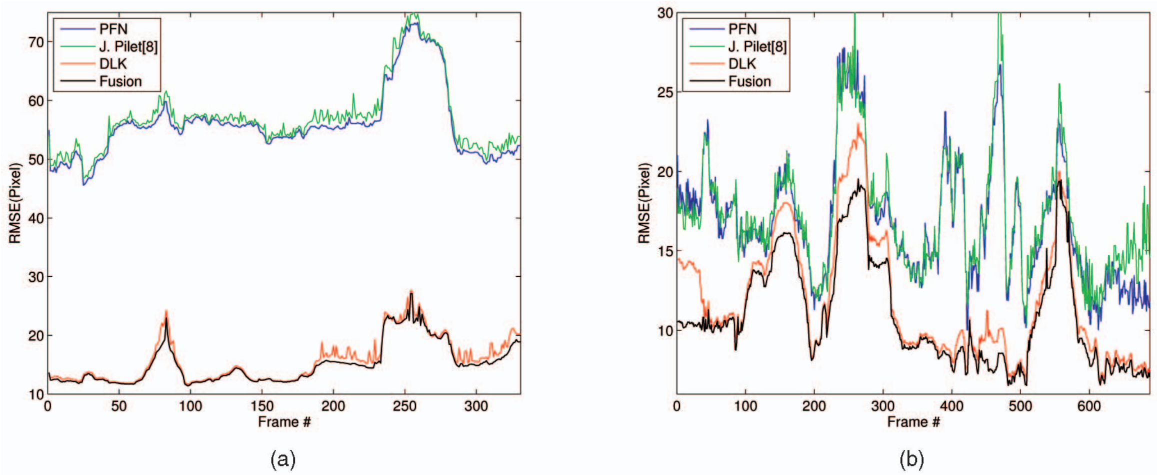


Fig. 14. The Root Mean Square Error (RMSE) comparison of the progressive finite Newton (PFN) method, the semi-implicit method [8], the deformable Lucas-Kanade (DLK) method, and the fusion approach on two videos. (a) As the model image and input video are from different sources, the RMSE for the feature-based method is much larger than that for the fusion method. (b) Both the model image and the input video are captured by the same device and under similar lighting conditions, so the RMSE is relatively low. Sample frames are shown in Figs. 15 and 16. (a) Magazine cover video sequence. (b) Paper video sequence.

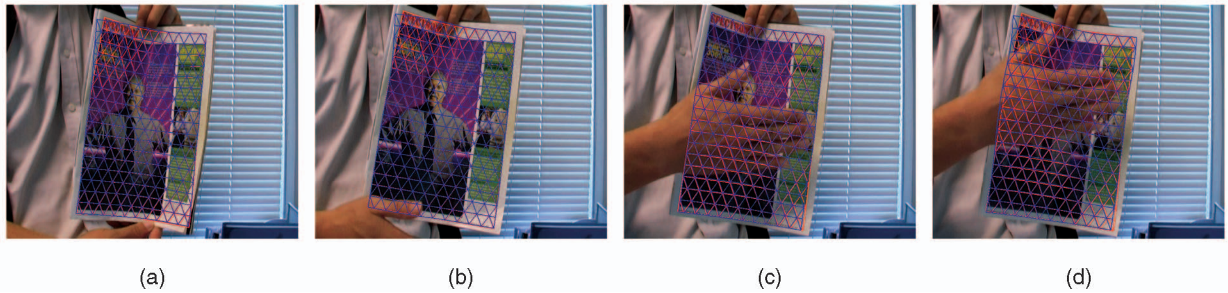


Fig. 15. Comparison of the deformable Lucas-Kanade (DLK) method (blue) and the fusion approach (red) on the magazine video. The magazine cover is occluded by hand in (c) and (d). (a) Frame 80. (b) Frame 229. (c) Frame 249. (d) Frame 271.

can also be found that the optimization seldom converges with small λ_r. Furthermore, as shown in the upper part of each figure, a large λ_r may lead to oversmoothing. We set $\lambda_r = 2 \times 10^4$ and $\alpha = 10^6$ in the following experiments.

4.4.2 Performance Evaluation

Two videos were captured for performance evaluation, which are the magazine cover and a piece of paper. To investigate the occlusion problem, the magazine cover is occluded by hand in some frames. For simplicity, the feature-based method in Section 3.4 is denoted as "PFN." The deformable Lucas-Kanade algorithm in Section 3.5 is denoted as "DLK," which is equivalent to the fusion approach with $\alpha = 0$. The proposed fusion approach is denoted as "Fusion." Fig. 14 shows the results of the comparison between two feature-based methods (PFN and that of Pilet et al. [8]) and two appearance-based methods (DLK and Fusion). From the experimental results, we first observe that the fusion approach consistently obtains the lowest RMSE. Second, we find that PFN is slightly better than our own implementation of the method of Pilet et al. [8]. Further, comparing the two appearance-based methods, DLK may suffer from the drift problem in some frames, as shown in Figs. 15 and 16. For those frames containing small deformations, the two methods obtain very similar results. Also, the proposed fusion approach and DLK method are able to handle the partial occlusion well without other treatments such as the robust loss functions, which is mainly due to our direct parameterizations and the regularization method. In addition, we use a piece of paper to occlude the patterns on the paper, and the results are shown in Fig. 17. Since the inverse compositional optimization starts from a good initialization, the optimization for the fusion approach usually requires around eight iterations.

Fig. 18 illustrates the results of recovering the nonrigid shape from a real-time video. We can observe that the

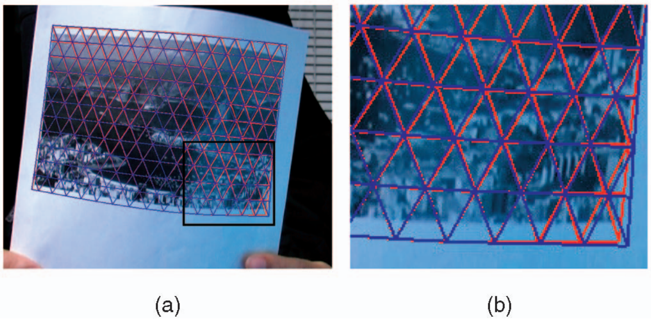


Fig. 16. Comparison of the deformable Lucas-Kanade (DLK) method (blue) and the fusion approach (red) on the paper video. Results are shown at frame 251. (a) Frame 251. (b) Zoomed region.

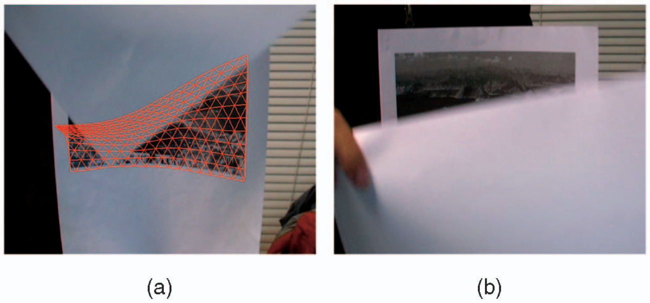


Fig. 17. Pattern occluded by a piece of paper. (a) Detected. (b) Failed.

fusion method is robust to large deformations and perspective distortions. In Fig. 19, we show the results of erasing the patterns on the recovered surfaces using the method in [16]. It can be seen that both the shadows and the specular regions are also correctly estimated. In addition, the artifacts in the resulting images are mainly due to an uncontrolled optical sensor.

5 Discussions and Future Work

5.1 Feature-Based Method

We have proposed a novel scheme for nonrigid surface detection by progressive finite Newton optimization. In comparison with semi-implicit optimization methods [16], the proposed method inherits several advantages. First, we need not solve the optimization iteratively for every σ because it can be solved in one step directly. Second, the iterative method starts from a sufficiently large support value in order to estimate the location and pose of an object, which leads to a large number of iterations. Thus, the proposed method is far more efficient than the semi-implicit method. In our experiments, both methods achieved similar accuracy that is mainly determined by the performance of the feature matching algorithm. Additionally, it is easy to implement the proposed approach, which only involves solving the sparse linear equation and does not require tuning the viscosity parameters and a sophisticated Levenberg-Marquardt optimization algorithm.

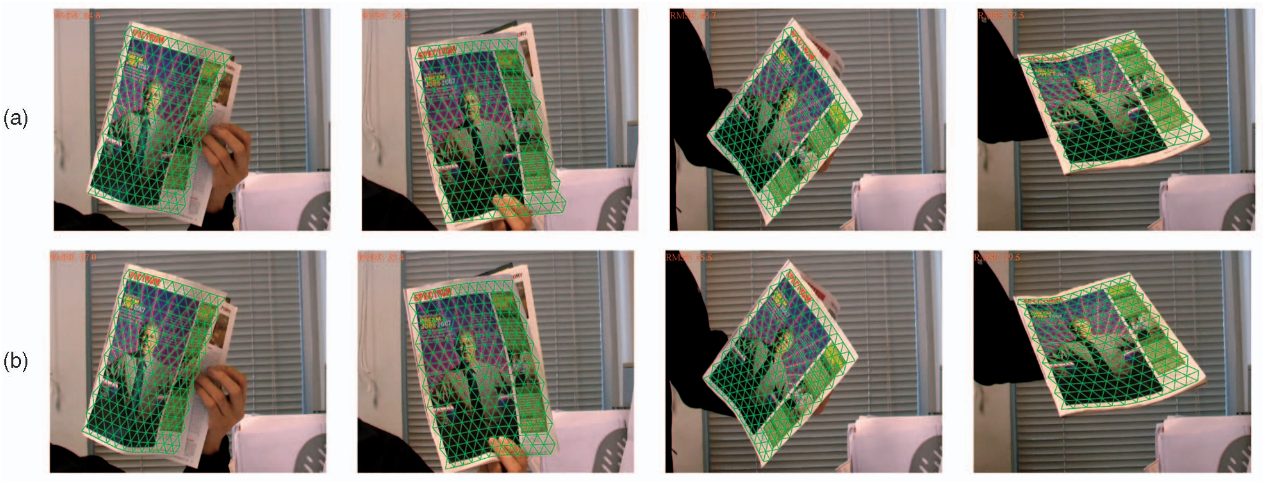


Fig. 18. Recovering the cover of a magazine in a real-time video with the size of 720 × 576. (a) shows the initialization results using feature correspondences only. (b) shows the results with the fusion approach. Moreover, the Root Mean Square Error (RMSE) is shown at the left corner in each image.

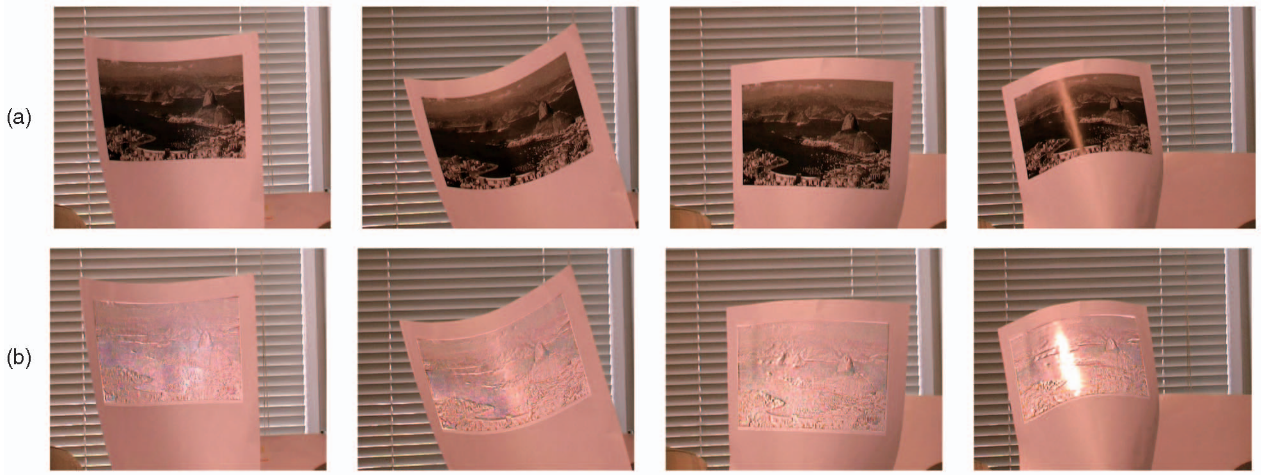


Fig. 19. Diminishing a picture on a piece of paper. (a) shows the 720 × 576 images captured by a DV camera. (b) shows the results of diminishing the texture on the paper.

5.2 Deformable Lucas-Kanade Algorithm

We discuss several major differences of our proposed deformable Lucas-Kanade algorithm compared with the previous work. In contrast to the conventional Lucas-Kanade algorithm for image alignment [25], our proposed deformable Lucas-Kanade algorithm can handle the large deformations rather than the affine transformation or the homography. Differently from the AAMs [12], [27], we do not need a set of representative training examples to build the shape and the texture models. Compared to other deformable template matching methods such as Active blob [23], the proposed deformable Lucas-Kanade algorithm has several advantages. First, our deformable model is more flexible. Second, the optimization of the proposed approach is an efficient sparse problem, which is able to reduce the computational cost by precomputing the gradient and Hessian.

5.3 Fusion Approach

In contrast to the feature-based image alignment methods [44], the fusion approach can deal with large deformations and perspective distortions in which correct feature correspondences are difficult to obtain. Also, the jitter is greatly reduced in the fusion approach. Furthermore, the proposed fusion method can handle the partial occlusion, which is mainly due to the triangulated mesh model and the regularization method. In this paper, however, we have not discussed the illumination issue for the appearance and fusion approaches, which may be tackled by employing an illumination model [14].

5.4 Future Work

Although promising experimental results have validated the efficiency of our methodology, some limitations and future directions should be addressed. First of all, we have focused our attention only on single deformable surface detection, whereas it is also interesting to study the multiple case. Moreover, small errors did occur in the boundary region. In future work, global bundle adjustment will be introduced to improve the performance. Furthermore, a second-order approximation method [45] can be used to compute the warp update. Finally, we may consider extending the proposed scheme to 3D environments, which will be effective to handle the self-occlusion problem.

6 Conclusion

This paper presented a fusion approach to solve the nonrigid shape recovery problem, which takes advantage of both the appearance information and the local features. First, a progressive finite Newton method is proposed to detect the nonrigid surface, which directly solves the unconstrained quadratic optimization problem by an efficient factorization method. Our modified RANSAC scheme takes advantage of our concise formulation and progressive sampling of the top-ranked correspondences and can handle high-dimensional spaces with noisy data. Second, the deformable Lucas-Kanade algorithm can handle image alignment when the deformation is large. We formulated the proposed deformable Lucas-Kanade algorithm into a sparse regularized least squares problem, which can be efficiently solved by the inverse compositional method. Finally, we solve the optimization problem for the fusion approach under the deformable Lucas-Kanade algorithm framework.

We have conducted extensive experimental evaluations on diverse objects with different materials. The proposed progressive finite Newton method is very fast and robust and can handle large deformations and illumination changes. It was tested in several applications such as real-time augmented reality and medical image registration. The promising experimental results showed that the approach is more efficient and effective than previous methods. Furthermore, the fusion approach can improve the accuracy for the nonrigid shape recovery.

Appendix

Optimization with Lighting

Performing the first-order Taylor expansion on (22) gives

$$\sum_{\mathbf{x}} \Bigg[(a+\Delta a)\bigg(T(W(\mathbf{x};\mathbf{s}_0)) + \nabla T \frac{\partial W}{\partial \mathbf{s}} \Delta\mathbf{s}\bigg) + (o+\Delta o)\cdot\mathbf{1} - I(W(\mathbf{x};\mathbf{s}))\Bigg]^2 + \lambda_r(\mathbf{s}+\Delta\mathbf{s})^\top \mathcal{K}(\mathbf{s}+\Delta\mathbf{s}).$$

Let D denote $\nabla T \frac{\partial W}{\partial \mathbf{s}}$; the gradient of the above equation can be derived as follows:

$$\begin{aligned}\frac{\partial E_A}{\partial \Delta\mathbf{s}} &= aD^\top[(a+\Delta a)(T+D\Delta\mathbf{s}) + (o+\Delta o)\cdot\mathbf{1} - I] \\ &\quad + \lambda_r \mathcal{K}(\mathbf{s}+\Delta\mathbf{s}), \\ \frac{\partial E_A}{\partial \Delta a} &= T^\top[(a+\Delta a)(T+D\Delta\mathbf{s}) + (o+\Delta o)\cdot\mathbf{1} - I], \\ \frac{\partial E_A}{\partial \Delta o} &= \mathbf{1}^\top[(a+\Delta a)(T+D\Delta\mathbf{s}) + (o+\Delta o)\cdot\mathbf{1} - I].\end{aligned}$$

The texture difference ΔI is computed by $\Delta I = I - aT - o$. Also, we define $\mathcal{T} = [T \quad \mathbf{1}]$, $\Delta\mathbf{g} = [\Delta a \quad \Delta o]$ and $H_4 = D^\top D + \frac{\lambda_r}{a^2}\mathcal{K}$. Thus, we can obtain the following equation by neglecting second-order terms:

$$\begin{bmatrix} a^2 H_4 & aD^\top\mathcal{T} \\ a\mathcal{T}^\top D & \mathcal{T}^\top\mathcal{T} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{s} \\ \Delta\mathbf{g} \end{bmatrix} = \begin{bmatrix} aD^\top\Delta I - \lambda_r\mathcal{K}\mathbf{s} \\ \mathcal{T}^\top\Delta I \end{bmatrix}.$$

As in [18], [35], we multiply a full-rank matrix $L \in R^{4N\times 4N}$ to the left side of the above equation:

$$L\begin{bmatrix} a^2 H_4 & aD^\top\mathcal{T} \\ a\mathcal{T}^\top D & \mathcal{T}^\top\mathcal{T} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{s} \\ \Delta\mathbf{g} \end{bmatrix} = L\begin{bmatrix} aD^\top\Delta I - \lambda_r\mathcal{K}\mathbf{s} \\ \mathcal{T}^\top\Delta I \end{bmatrix},$$

where L is defined as follows:

$$L = \begin{bmatrix} \operatorname{diag}(\mathbf{1}_{2N}) & 0 \\ -\frac{1}{a}\mathcal{T}^\top D H_4^{-1} & \operatorname{diag}(\mathbf{1}_{2N}) \end{bmatrix}.$$

Simplifying (27), we can obtain $\Delta\mathbf{g}$ by solving the following equation:

$$(Q - G^\top H_4^{-1} G)\Delta\mathbf{g} = \mathcal{T}^\top\Delta I - G^\top H_4^{-1}\bigg(D^\top\Delta I - \frac{\lambda_r}{a}\mathcal{K}\mathbf{s}\bigg),$$

where $G = D^{\top}\mathcal{T}$, and $Q = \mathcal{T}^{\top}\mathcal{T}$. Also, $\Delta\mathbf{s}$ can be computed by

$$\Delta\mathbf{s} = \frac{1}{a}\left[H_4^{-1}\left(D^{\top}\Delta I - \frac{\lambda_r}{a}\mathcal{K}\mathbf{s}\right) - H_4^{-1}G\Delta\mathbf{g}\right].$$

Similarly, we compute the warp update through (21). Note that we can precompute G and Q in order to reduce the computational cost. As the regularization coefficient λ_r can be chosen in a very wide range without significantly affecting the results [8], [16], we treat H_4 as constant (set $a = 1$) and ignore the changes of a during the optimization.

Acknowledgments

The authors appreciate the reviewers for their extensive and informative comments for the improvement of this manuscript. Also, the authors would like to thank Dr. Guangyu Wang and Mr. Xiaopei Liu for their fruitful discussions on the GPU programming. The work was fully supported by two Hong Kong Government grants: the Innovation and Technology Fund (ITS/084/07) and the Research Grants Council Earmarked Grant (CUHK4150/07E). The short version of this paper appeared in our previous work published in the *Proceedings of the 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*.

References

[1] L. Brown, "A Survey of Image Registration Techniques," *ACM Computing Surveys,* vol. 24, no. 4, pp. 325-376, Dec. 1992.

[2] A. Bartoli and A. Zisserman, "Direct Estimation of Non-Rigid Registration," *Proc. 15th British Machine Vision Conf.,* Sept. 2004.

[3] L.V. Tsap, D.B. Goldgof, and S. Sarkar, "Nonrigid Motion Analysis Based on Dynamic Refinement of Finite Element Models," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 5, pp. 526-543, May 2000.

[4] H. Fang and J.C. Hart, "Rototexture: Automated Tools for Texturing Raw Video," *IEEE Trans. Visualization and Computer Graphics,* vol. 12, no. 6, pp. 1580-1589, Nov./Dec. 2006.

[5] H. Fang and J.C. Hart, "Textureshop: Texture Synthesis as a Photograph Editing Tool," *ACM Trans. Graphics,* vol. 23, no. 3, pp. 354-359, 2004.

[6] R. White and D. Forsyth, "Retexturing Single Views Using Texture and Shading," *Proc. Ninth European Conf. Computer Vision,* pp. 70-81, 2006.

[7] R. White and D.A. Forsyth, "Combining Cues: Shape from Shading and Texture," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1809-1816, 2006.

[8] J. Pilet, V. Lepetit, and P. Fua, "Real-Time Non-Rigid Surface Detection," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 822-828, 2005.

[9] M.J. Black and Y. Yacoob, "Tracking and Recognizing Rigid and Non-Rigid Facial Motions Using Local Parametric Models of Image Motion," *Proc. Fifth Int'l Conf. Computer Vision,* pp. 374-381, 1995.

[10] M. Brand and R. Bhotika, "Flexible Flow for 3D Nonrigid Tracking and Shape Recovery," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 315-322, 2001.

[11] L. Torresani, D. Yang, E. Alexander, and C. Bregler, "Tracking and Modeling Non-Rigid Objects with Rank Constraints," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 493-500, 2001.

[12] T. Cootes, G. Edwards, and C. Taylo, "Active Appearance Models," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 6, pp. 981-685, June 2001.

[13] J. Zhu, S.C. Hoi, and M.R. Lyu, "Real-Time Non-Rigid Shape Recovery via Active Appearance Models for Augmented Reality," *Proc. Ninth European Conf. Computer Vision,* pp. 186-197, 2006.

[14] 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, Sept. 2003.

[15] D. DeCarlo and D.N. Metaxas, "Optical Flow Constraints on Deformable Models with Applications to Face Tracking," *Int'l J. Computer Vision,* vol. 38, no. 2, pp. 99-127, 2000.

[16] J. Pilet, V. Lepetit, and P. Fua, "Fast Non-Rigid Surface Detection, Registration and Realistic Augmentation," *Int'l J. Computer Vision,* vol. 76, no. 2, pp. 109-122, 2008.

[17] M.A. Fischler and R.C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Comm. ACM,* vol. 24, no. 6, pp. 381-395, 1981.

[18] R.I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision.* Cambridge Univ. Press, 2000.

[19] S. Belongie, J. Malik, and J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 4, pp. 509-522, Apr. 2002.

[20] H. Chui and A. Rangarajan, "A New Point Matching Algorithm for Non-Rigid Registration," *Computer Vision and Image Understanding,* vol. 89, nos. 2-3, pp. 114-141, 2003.

[21] G. McNeill and S. Vijayakumar, "Part-Based Probabilistic Point Matching Using Equivalence Constraints," *Advances in Neural Information Processing Systems,* vol. 19, pp. 969-976, 2007.

[22] M. Salzmann, J. Pilet, S. Ilic, and P. Fua, "Surface Deformation Models for Non-Rigid 3-D Shape Recovery," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 8, pp. 1481-1487, Aug. 2007.

[23] S. Sclaroff and J. Isidoro, "Active Blobs: Region-Based, Deformable Appearance Models," *Computer Vision and Image Understanding,* vol. 89, nos. 2-3, pp. 197-225, 2003.

[24] O. Chum and J. Matas, "Matching with PROSAC—Progressive Sample Consensus," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 220-226, 2005.

[25] S. Baker and I. Matthews, "Lucas-Kanade 20 Years On: A Unifying Framework," *Int'l J. Computer Vision,* vol. 56, no. 3, pp. 221-255, Mar. 2004.

[26] W.-C. Lin and Y. Liu, "Tracking Dynamic Near-Regular Texture under Occlusion and Rapid Movements," *Proc. Ninth European Conf. Computer Vision,* pp. 44-55, 2006.

[27] I. Matthews and S. Baker, "Active Appearance Models Revisited," *Int'l J. Computer Vision,* vol. 60, no. 2, pp. 135-164, 2004.

[28] V. Gay-Bellile, A. Bartoli, and P. Sayd, "Feature-Driven Direct Non-Rigid Image Registration," *Proc. 18th British Machine Vision Conf.,* 2007.

[29] S. Granger and X. Pennec, "Multi-Scale EM-ICP: A Fast and Robust Approach for Surface Registration," *Proc. Seventh European Conf. Computer Vision,* pp. 418-432, 2002.

[30] D.G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int'l J. Computer Vision,* vol. 60, no. 2, pp. 91-110, 2004.

[31] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 27, no. 10, pp. 1615-1630, Oct. 2005.

[32] H. Ling and D.W. Jacobs, "Deformation Invariant Image Matching," *Proc. 10th IEEE Int'l Conf. Computer Vision,* pp. 1466-1473, 2005.

[33] V. Lepetit and P. Fua, "Keypoint Recognition Using Randomized Trees," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 9, pp. 1465-1479, Sept. 2006.

[34] B.D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proc. Seventh Int'l Joint Conf. Artificial Intelligence,* pp. 674-679, 1981.

[35] A. Bartoli, "Groupwise Geometric and Photometric Direct Image Registration," *Proc. 17th British Machine Vision Conf.,* Sept. 2006.

[36] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models," *Int'l J. Computer Vision,* vol. 1, no. 4, pp. 321-331, Jan. 1988.

[37] P. Fua and Y. Leclerc, "Object-Centered Surface Reconstruction: Combining Multi-Image Stereo and Shading," *Int'l J. Computer Vision,* vol. 16, no. 1, pp. 35-56, Sept. 1995.

[38] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge Univ. Press, 2004.

[39] O.L. Mangasarian, "A Finite Newton Method for Classification," *Optimization Methods and Software,* vol. 17, no. 5, pp. 913-929, 2002.

[40] S.S. Keerthi and D. DeCoste, "A Modified Finite Newton Method for Fast Solution of Large Scale Linear SVMs," *J. Machine Learning Research,* vol. 6, pp. 341-361, 2005.

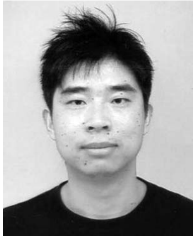
[41] S. Ilic, M. Salzmann, and P. Fua, "Implicit Meshes for Effective Silhouette Handling," *Int'l J. Computer Vision,* vol. 72, no. 2, pp. 159-178, 2007.

[42] A. Bartoli, "Groupwise Geometric and Photometric Direct Image Registration," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, no. 12, pp. 2098-2108, Dec. 2008.

[43] S. Periaswamy and H. Farid, "Medical Image Registration with Partial Data," *Medical Image Analysis,* no. 10, pp. 452-464, 2006.

[44] J. Zhu and M.R. Lyu, "Progressive Finite Newton Approach to Real-Time Nonrigid Surface Detection," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1-8, 2007.

[45] S. Benhimane and E. Malis, "Real-Time Image-Based Tracking of Planes Using Efficient Second-Order Minimization," *Proc. IEEE/RSJ Int'l Conf. Intelligent Robots and Systems,* pp. 943-948, 2004.



Jianke Zhu received the bachelor's degree in mechatronics and computer engineering from the Beijing University of Chemical Technology in 2001 and the master's degree in electrical and electronics engineering from the University of Macau in 2005. He is currently a PhD candidate in the Computer Science and Engineering Department at the Chinese University of Hong Kong. His research interests include computer vision, pattern recognition, and statistical machine learning. He is a student member of the IEEE.

Michael R. Lyu received the BS degree in electrical engineering from National Taiwan University, Taipei, in 1981, the MS degree in computer engineering from the University of California at Santa Barbara in 1985, and the PhD degree in computer science from the University of California, Los Angeles, in 1988. He was with the Jet Propulsion Laboratory from 1988 to 1990, with the Department of Electrical and Computer Engineering, University of Iowa, from 1990 to 1992, with Bell Communications Research (Bellcore), Morristown, New Jersey, from 1992 to 1995, and with Bell Laboratories, Murray Hill, New Jersey, from 1995 to 1997. In 1998, he joined the Chinese University of Hong Kong, where he is currently a professor in the Department of Computer Science and Engineering. He is also the founder and the director of the Video over InternEt and Wireless (VIEW) Technologies Laboratory. His research interests include software reliability engineering, distributed systems, fault-tolerant computing, mobile and sensor networks, Web technologies, multimedia information processing and retrieval, and machine learning. He has published 300 refereed journal and conference papers in these areas. He was the editor of two book volumes: *Software Fault Tolerance* (John Wiley & Sons, 1995) and *The Handbook of Software Reliability Engineering* (IEEE and McGraw-Hill, 1996). He initiated the First International Symposium on Software Reliability Engineering (ISSRE) in 1990. He was the program chair for ISSRE 1996 and the general chair for ISSRE 2001. He was also program cochair of PRDC 1999, WWW10, SRDS 2005, and ICEBE 2007 and a general cochair of PRDC 2005. He was on the editorial board of the *IEEE Transactions on Knowledge and Data Engineering*, *IEEE Transactions on Reliability*, *Journal of Information Science and Engineering*, and *Wiley Software Testing, Verification & Reliability Journal*. He is a fellow of the IEEE and AAAS and a Croucher senior research fellow.

Thomas S. Huang received the BS degree in electrical engineering from the National Taiwan University, Taipei, and the MS and ScD degrees in electrical engineering from the Massachusetts Institute of Technology (MIT), Cambridge. He was on the faculty of the Department of Electrical Engineering at MIT from 1963 to 1973 and on the faculty of the School of Electrical Engineering and the director of its Laboratory for Information and Signal Processing at Purdue University from 1973 to 1980. In 1980, he joined the University of Illinois, Urbana-Champaign, where he is currently the William L. Everitt Distinguished Professor of Electrical and Computer Engineering, a research professor in the Coordinated Science Laboratory, the head of the Image Formation and Processing Group at the Beckman Institute for Advanced Science and Technology, and a cochair of the institute's major research theme—Human Computer Intelligent Interaction. His professional interests lie in the broad area of information technology, especially the transmission and processing of multidimensional signals. He has published 20 books and more than 500 papers in network theory, digital filtering, image processing, and computer vision. He is a member of the National Academy of Engineering, a foreign member of the Chinese Academies of Engineering and Sciences, and a fellow of the International Association of Pattern Recognition, the IEEE, and the Optical Society of American. He has received a Guggenheim Fellowship, an A.V. Humboldt Foundation Senior US Scientist Award, and a fellowship from the Japan Association for the Promotion of Science. He received the IEEE Signal Processing Society's Technical Achievement Award and the Society Award, the IEEE Third Millennium Medal, Honda Lifetime Achievement Award, the IEEE Jack S. Kilby Medal, the King-Sun Fu Prize of International Association of Pattern Recognition, the Pan Wen-Yuan Outstanding Research Award, and the Okawa Prize.