Dynamic Human Body Modeling Using a Single RGB Camera

Haiyu Zhu, Yao Yu *, Yu Zhou * and Sidan Du

School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China; MF1423044@smail.nju.edu.cn (H.Z.); cof128@nju.edu.cn (S.D.)

* Correspondence: allanyu@nju.edu.cn (Y.Y.); nackzhou@nju.edu.cn (Y.Z.); Tel./Fax: +86-25-8968-0827 (Y.Z.)

Academic Editor: Edoardo Charbon
Received: 28 January 2016; Accepted: 16 March 2016; Published: 18 March 2016

Abstract: In this paper, we present a novel automatic pipeline to build personalized parametric models of dynamic people using a single RGB camera. Compared to previous approaches that use monocular RGB images, our system can model a 3D human body automatically and incrementally, taking advantage of human motion. Based on coarse 2D and 3D poses estimated from image sequences, we first perform a kinematic classification of human body parts to refine the poses and obtain reconstructed body parts. Next, a personalized parametric human model is generated by driving a general template to fit the body parts and calculating the non-rigid deformation. Experimental results show that our shape estimation method achieves comparable accuracy with reconstructed models using depth cameras, yet requires neither user interaction nor any dedicated devices, leading to the feasibility of using this method on widely available smart phones.

Keywords: SCAPE; non-rigid reconstruction; single RGB camera; pose estimation; motion classification; structure from motion

1. Introduction

Human body modeling has vast applications, such as computer games, animations and virtual fittings, and has long been an active topic in many research communities, from visual sensing and computer vision to computer graphics. Researchers have developed a number of approaches using depth images [1–5], 3D scanners [6,7] and multi-view images [8–12] to extract 3D human models over the last few decades. These approaches have progressively enabled faster and better production of 3D human models.

1.1. Human Body Modeling Using Depth Sensors

The emergence of depth sensors, such as the Kinect sensor, in recent years has made it possible to perform 3D reconstruction from depth or RGB-depth (RGB-D) images. Newcombe et al. [3] created a system called KinectFusion, which provides a detailed 3D reconstruction of fixed scenes, but this method can be applied only to static objects. Follow-up studies based on KinectFusion specifically focused on scanning humans, where a user rotates in front of the Kinect while maintaining a roughly rigid pose [4,13–15]. Zhang et al. [5] registered several Kinect scans of an object in multiple poses and used these data to train a personalized body model, which was then fitted to dynamic data. Newcombe et al. [2] extended the KinectFusion method to capture dynamic 3D shapes that included partial views of moving people by utilizing a dense volumetric warp-field parametrization. Bogo et al. [16] introduced a multi-resolution body model called Delta, which used a set of variable details for the shapes and locations of body parts along with a displacement map to capture shape details from images. Dou et al. [1] proposed a system that allowed a considerable amount of non-rigid deformation during scanning and achieved high quality results without heavily constraining users or...
camera motion by performing a dense non-rigid bundle adjustment to optimize the final shape and non-rigid parameters for each frame.

1.2. Multi-View Human Body Modeling

Multi-view stereo modeling uses multiple, sparsely-deployed cameras to observe a scene from different viewpoints. Gall et al. [8] recovered not only the movements of skeletons, but also the possible non-rigid temporal deformations from a multi-view image sequence using an unsupervised method that tracked the skeletons and consistently estimated the surface variations over time. Liu et al. [10] split the tracking problem into a multi-view 2D segmentation problem in which the segmentation separates persons by labeling each foreground pixel and a 3D pose and shape estimation problem. Huang et al. [9] introduced the notion of keyframes to 3D human motion tracking and proposed a keyframe-based tracking framework that updates the keyframe pool incrementally. In addition, they presented a new outlier rejection method to improve the method's robustness and significantly limits the impact of missing data and outliers. Robertini et al. [11] proposed a new shape representation that models a mesh surface using a dense collection of 3D Gaussian functions centered at each vertex and formulated a dense photo-consistency-based surface refinement as a global optimization problem for the position of each vertex on the surface. Zollhofer et al. [12] utilized the volumetric fusion framework to create a multi-resolution hierarchy using a stereo camera and then employed real-time, non-rigid reconstruction to produce a deformed mesh at every time step.

1.3. Single-View Human Body Modeling

Guan et al. [17] fit a static model of human body poses and shapes to a single image using cues, such as silhouettes, edges and smooth shading. Hasler et al. [18] presented an approach to estimate a human body pose and shape from a single image using a bilinear statistical model. Although having several images of one subject can improve shape estimation, the human model still suffers from surface contortions. Jain et al. [19] re-projected a parametric 3D model onto an actor’s silhouette to optimize both the shape and pose parameters with the help of a Kanade-Lucas-Tomasi (KLT) tracker along with some manually-introduced features.

Unlike specialized setups or dedicated equipment for 3D reconstruction, using a single RGB camera, such as the integrated camera on mobile phones, has much more potential value for real-life applications. Although considerable attention has been paid to reconstructing human models from images or videos, e.g., acquiring shapes from shading [17], from silhouettes [9,10,17] and from model-based methods [18–20], some of these techniques are limited by assumptions concerning light source directions or intensities, while others may suffer from the insufficient constraint of single-image silhouettes or rely heavily on a good parametric model as the input. Zhou et al. [20] devised a method to compute shape completion and animation for people (SCAPE) parameters to match input images and enable a model-based morphing process for image editing. However, in this approach, users are required to carefully label the body parts and joints manually before computation, to prevent the method from getting stuck in local minima. Yu et al. [21] computed a dense 3D template of the object in a static pose from a short fixed sequence, which they used to subsequently perform non-rigid reconstruction.

Due to the high dynamics of the human body and the ambiguity that results from perspective projection, automatic human modeling using monocular RGB images or videos is still a challenging problem.

To overcome the limitations described earlier, this paper proposes an automatic system that can model dynamic human bodies using a single RGB camera, as demonstrated in Figure 1. Compared to previous works, whose performance degrades when humans are in motion, we investigate whether human motion can be advantageous for improving human model reconstruction quality. The underlying feasibility of this approach arises from the fact that an object in motion may provide more cues for shape recovery than a static object, because self-occluded parts have a higher chance.
of being captured during motion. On the other hand, motion can be considered as an effective clue for human body segmentation, because coincident motion is more reliable evidence than texture, silhouette or location. To achieve this, we begin by designing a kinematic classification to relax the non-rigid recovery problem into a task of piecewise rigid reconstruction. The model for each rigid part is maintained and continuously improved as the image sequence acquisition process progresses. For each update of each part model, we initiate a non-rigid fusion method to merge the parts into a complete human model; thus, the human model is continuously and incrementally being built and improved along with the part models.

Figure 1. A performer is acting freely in front of a single camera, and our system automatically reconstructs a model of the performer. The first row shows the motion of the performer recorded by a common camera, and the second row shows 3D models that are obtained through our system and fitted to the poses in some of the frames from the video sequence.

To the best of our knowledge, our system is the first that can automatically reconstruct a model for dynamic people from image sequences captured by a single RGB camera.

Our paper is organized as follows: Section 2 introduces the proposed approach. Experimental results and the discussion are shown in Section 3. Finally, we conclude our study in Section 4.

2. Building 3D Models

In this section, we supply the details of the pipeline depicted in Figure 2, which consists of the following steps: (i) Rather than manual labeling, we roughly derive 2D and 3D joint locations for each frame by running a software detector; (ii) The obtained 3D pose is refined via a carefully-designed kinematic classifier, and the body image is eventually segmented into rigid parts to relax the non-rigid problem into a rigid reconstruction problem; (iii) We propose a bundle adjustment-based approach to reconstruct the shapes of rigid parts by making full use of their motion flow. The reconstructed shape is continuously improved as more images are processed; (iv) A general template is finally deformed to fit the body parts, and the non-rigid component of shape deformation is derived to form a vivid 3D
human model. As the rigid parts are continuously and incrementally being improved, the final shape of the human is also enhanced, along with the non-rigid deformation in each frame.

![Diagram](Image)

**Figure 2.** Our system’s pipeline begins by taking an image sequence as input, and its output is the reconstructed human model.

### 2.1. Initial Pose Detection

In our framework, we adopt a general SCAPE model [6], illustrated in Figure 3, trained from the database provided by [22], to solve the pose transformation and non-rigid mesh deformation problems.

![Diagram](Image)

**Figure 3.** (a) Our template and (b) the template in a different pose with 16 color-coded parts.

To deform the SCAPE model to the poses of the people in every frame, we need 3D poses of people to derive the pose transformations. To do this, we run a 2D pose detector [23] and a 3D pose parser [24] to roughly obtain the 3D poses for the given images in a frame-by-frame manner. Then, we refine each detected pose using a Kalman filter [25] to make it robust to noise and inaccurate 2D joint estimations. Specifically, we consider pose estimation as a linear dynamic system discretized in the time domain. In Equation (1), we can assume that \( P_i \) (i.e., the pose in the \( i \)-th frame) has evolved from \( P_{i-1} \), because 3D poses of adjacent frames are highly related to each other, as follows:

\[
P_i = F_i P_{i-1} + B_i u_i + w_i
\]

where \( F_i \) is the state-transition model applied to the previous pose \( P_{i-1} \) and \( B_i \) is the control-input model for control vector \( u_i \). The process noise, \( w_i \), is assumed to have a zero-mean multivariate normal distribution with constant covariance. In the \( i \)-th frame, we obtain an estimation \( Z_i \) of the true pose \( P_i \) that satisfies Equation (2):

\[
Z_i = H_i P_i + v_i
\]
where $H_i$ is the observation model and $v_i$ is the observation noise, which is assumed to be zero-mean Gaussian white noise with constant covariance [25]. After obtaining a pose from the pose estimator, we immediately refine it to be more reliable using the Kalman filter.

2.2. Pose Refinement via Kinematic Classification

The poses derived above implicitly provide a classification of rigid body parts for the given images; however, such a classification is usually far removed from the correct poses needed for further computation due to projective ambiguity and self-occlusion during movements. To achieve a better model, the initial roughly-detected poses are refined via a kinematic classifier. Note that non-rigid deformation is temporarily ignored here, but will be discussed later in Section 2.4. In physics, a rigid part is an ideal region that moves consistently; therefore, we can regard the regions in which points have similar motion as rigid body parts. We divide the body into 16 parts according to the skeletons and the SCAPE model.

![Figure 4. Stages in our kinematic classification. (a) The silhouette of our template projected to an image; (b) the silhouette of people in an image; (c) the result before alignment; (d) the aligned result after coherent point drift (CPD) [26]; (e) the classification of people’s silhouette obtained from our template; (f) the kinematic classification result. First, we extract the silhouettes from our template and people; then, we can classify the people’s silhouettes after aligning the silhouette in (b) to (a). Finally, we obtain the kinematic classifications (f) of the people in the images through a fully-connected conditional random field (CRF) model.](image-url)

We model the image as a random field $I$ defined over the variables $\{I_1, \ldots, I_N\}$, where $I_i$ is the color of pixel $i$ at location $p_i$. A random field $X$ is defined over a set of variables $\{X_1, \ldots, X_N\}$ to label each pixel $i$ as $X_i$. We denote $d_i$ as the displacement of pixel $i$, which we obtain by the large displacement optical flow algorithm proposed by Brox and Malik [27]. The reduced Gibbs energy of the conditional random field is then calculated to produce a refined label $X_i$ for each pixel, as shown below:

$$E(I, X) = \sum_i \psi_u(I_i, X_i) + \sum_{i<j} \psi_p(X_i, X_j, I_i, I_j)$$

(3)

where $i$ and $j$ range from one to $N$. The unary potential $\psi_u$ is evaluated according to how well the label $X$ matches the input image $I$, derived as follows:

$$C(I_i, X_i) = \omega_s u_s(p_i, S(X_i)) + \omega_j u_j(p_i, B(X_i)) + \omega_m u_m(d_i, D(X_i))$$

(4)

s.t.

$$\omega_s + \omega_j + \omega_m = 1$$

where $\omega_s$, $\omega_j$ and $\omega_m$ are weighting factors and $S(X_i)$ returns the set of contour points for each labeled $X_i$. Such points are computed by projecting our template to the image. The template silhouette is
aligned to the image silhouette via coherent point drift (CPD) \cite{26}. An example of the process is illustrated in Figure 4. The silhouette energy \( u_s \) is derived from the minimum square distance between the location \( p_i \) and the contour points:

\[
u_s(p_i, S(X_i)) = \exp\left(-\min\left(\frac{|p_i - S(X_i)|^2}{2\theta^2_s}\right)\right) \tag{5}\]

\( B(X_i) \) is the barycenter of the bone assigned to label \( X_i \) in the image; thus, joint energy \( u_j \) is calculated as follows:

\[
u_j(p_i, B(X_i)) = \exp\left(-\frac{|p_i - B(X_i)|^2}{2\theta^2_j}\right) \tag{6}\]

where the parameters \( \theta_s \) and \( \theta_j \) control the connection strength and range of pixels.

To justify the classification, a motion energy term \( u_m \) is proposed to judge how the motion of \( I_i \) matches the motion of the rigid part \( X_i \), where \( D(X_i) \) is the displacement of the barycenter of the body part assigned to label \( X_i \), and the motion energy is evaluated to determine how well the motion of \( I_i \) agrees with the motion of the barycenter of the body part labeled by \( X_i \):

\[
u_m(d, D(X_i)) = \rho_d \left(1 + \frac{d \cdot D(X_i)}{|d||D(X_i)|}\right) \tag{7}\]

The pairwise potentials \( \psi_p \) in Equation (3) are obtained by:

\[
\psi_p(x_i, x_j, I_i, I_j) = \mu(x_i, x_j) \exp\left(-\frac{|p_i - p_j|^2}{2\theta^2_\alpha} - \frac{|X_i - X_j|^2}{2\theta^2_\beta}\right) + \mu(x_i, x_j) \exp\left(u_m(d, d) - 1\right) \tag{8}\]

where \( \theta_\alpha \) and \( \theta_\beta \) are parameters to control the degrees of nearness and similarity. The label compatibility function \( \mu(x_i, x_j) \) is provided by the Potts model, \( \mu(x_i, x_j) = [x_i \neq x_j] \). As a result, \( \psi_p \) tends to make pixels within an area with similar motion more likely to be labeled as the same part.

We use the mean field method with an efficient implementation proposed by Krahenbuhl et al. \cite{28} to optimize Equation (3). To show the importance of the motion term in Equation (4), we assign \( \omega_m = 0 \), and we solve the dense CRF model in the same way. Figure 5 demonstrates that the classification with the motion term (the right figure in Figure 5) is more precise than the classification without the motion term (the middle figure in Figure 5), which results in incorrect classifications, such as the right hand and the fuzzy classification of the right leg. However, with the help of the motion term, our kinematic classification not only provides a clear classification of the right leg, but also distinguishes between the right hand and the chest.

Figure 5. Body part kinematic classification. The left image is the source image recorded by an RGB camera, while the middle image is the classification result without the motion term. The classification result shown in the right-hand image is more precise than the result in the middle image, because of the motion term.
2.3. Dense Reconstruction

By performing kinematic classification in each frame of the sequence, we transform the non-rigid reconstruction into a rigid reconstruction that can be achieved by a structure from motion-based approach. In practice, to enable our system to tackle images with sparsely-detected features and unstable detection for shapes in non-rigid motion, we make use of the dense optical flow approach to perform reconstruction, instead of using sparse features, as demonstrated in Algorithm 1.

**Algorithm 1 Dense Reconstruction of Body Parts**

1: \( i \): the frame number of the input image  
2: \( \mathbf{l}_b^i \): the 2D barycenter of \( \mathbf{B}_b \) in the frame \( f_i \)  
3: \( f_b \): the number of start frame of part \( \mathbf{B}_b \), denoted as ref for simplification  
4: for \( b \leftarrow 1 \) to 16 do  
5: \( \text{if } \| \mathbf{l}_b^i - \mathbf{l}_b^\text{ref}^i \| > 30 \text{ then} \)  
6: \( \mathbf{T}_b^\text{ref} \leftarrow \mathbf{T}_b^\text{ref}^i \)  
7: for \( j \leftarrow f_b + 1 \) to \( i \) do  
8: \( \mathbf{T}_b^j \leftarrow \mathbf{\Delta T}_{\text{ref}^j \rightarrow j}^b \mathbf{T}_b^\text{ref} \)  
9: derive dense optical flow from \( \text{ref} \) to \( f_j \)  
10: end for  
11: \( \mathbf{P}_b \leftarrow \min E_{BA} \)  
12: \( f_b \leftarrow f_i \)  
13: end if  
14: end for

Here, \( f_i \) represents the \( i \)-th frame in the image sequence, and \( \mathbf{B}_b \) represents the \( i \)-th body part of a person. For each part, we regard the motion of parts over a short duration as rigid motion and track the motion throughout the image sequence, by using the large displacement optical flow (LDOF) algorithm [27] to obtain a dense optical flow with sub-pixel accuracy. When the motion of any part, e.g., \( \mathbf{B}_b \), reaches a threshold value (20 pixels), we choose the frames from \( f_b \) (in which the part starts to move) through \( f_i \) (where the motion of this part reaches the threshold) to perform the bundle adjustment. The threshold is set according to the re-projection error obtained through bundle adjustment (Equation (9)). When the threshold is small, the depth of the points is highly uncertain because of the short baseline, which contributes to the large re-projection error. However, if we choose a large threshold, which means that more non-rigid components of the motion will be involved in the bundle adjustment, the re-projection error is also large. To set the threshold to an appropriate value, we increase it gradually, starting from a small value, and choose the value at which the re-projection error reaches a minimum. The start frame index \( f_b \) is then updated to the current frame index \( f_i \). Let \( \pi : \mathbb{R}^3 \rightarrow \mathbb{R}^2 \) be the projection function; then, the cost function of the bundle adjustment is defined as:

\[
E_{BA} = \sum_{j=f_b}^{i} \sum_{k \in \mathbf{B}_b} \| \mathbf{p}_j^k - \pi \left( \mathbf{r}_j^k \mathbf{P}_k + \mathbf{t}_j^k \right) \|^2
\]

where the 3D location of the \( k \)-th point in part \( \mathbf{B}_b \) is denoted as \( \mathbf{P}_k \), whose projection on the \( j \)-th frame is \( \mathbf{p}_j^k \), and \( \mathbf{T}_b^j = \left[ \mathbf{r}_j^b \mathbf{t}_j^b \right] \) is the relative camera pose viewing \( \mathbf{B}_b \) of the \( j \)-th frame. Note that the relative camera pose, denoted as \( \mathbf{T}_b^j \), is not the absolute camera pose \( \mathbf{T}_j^* \); the relative camera pose is used here to measure the relative motion between the camera and each part \( \mathbf{B}_b \) individually, including both the camera motion and the part motion. For instance, taking the motion of \( \mathbf{B}_b \) into consideration, we can obtain the matrix \( \mathbf{\Delta T}_{\text{ref}^j \rightarrow j}^b \) of \( \mathbf{B}_b \), transforming from the reference frame to the \( j \)-th frame by driving the skeleton. Thus, \( \mathbf{T}_j^b = \mathbf{\Delta T}_{\text{ref}^j \rightarrow j}^b \mathbf{T}_b^\text{ref} \).
We utilize sparse bundle adjustment [29] to minimize the cost Equation (9), and all of the points are initially set to a random depth between two and four meters. Finally, we make use of the dense correspondences of pixels to reconstruct the 3D surfaces of parts. An example of dense reconstruction is shown in Figure 6.

![Figure 6. Dense reconstruction of seven body parts.](image)

2.4. Decomposing the Deformation

In this section, we decompose the deformation of the human shape into rigid and non-rigid components. The rigid component of motion consists of both the rigid motion of the body shape and that of clothing. To achieve this, we further process the reconstructed body parts, using the temporal information within each reconstructed part model, to compute the rigid component of the human model and the non-rigid component for the human deformations in each frame.

2.4.1. Building Vertex-Wise Correspondence

The rigid component of the human model is a pose-invariant deformation for a given performer in the image sequence. This means that such a component will implicitly form a coincident deformation along all of the frames. To investigate this pose-invariant deformation, a vertex-wise correspondence must be built. Unfortunately, the reconstructed shapes of body parts contain only the geometry information of the body parts in each frame and do not maintain semantic information for vertex-wise correspondence. Therefore, we use the general template used in the kinematic classification to merge the body part over time, by aligning the template and the recovered models of body parts. Note that the general template can look quite unlike the recovered people in Figure 3, i.e., the difference between the template and the human model affects neither the aforementioned kinematic classification nor the merging process here. Specifically, we drive the template to fit the poses in the reference frames of the body part that are used in the dense reconstruction and vary during the process of reconstruction. Then, the visible points only are transformed to the dense reconstruction by the Z-buffer algorithm and aligned to the dense reconstruction through CPD. Eventually, an instance is obtained. A demonstration of the merging results is shown in Figure 7.

To validate the CPD algorithm in our point registration, we map the Euclidean distances of the correspondent points provided by CPD to the color space. The registration error is shown in Figure 8. As the preceding figure demonstrates, the registration result is acceptable. The maximum error is 0.5 cm, located in a small region of the left arm. The number of points transformed in the template is 2689, and the mean registration error is approximately 0.8 mm. Although the registration result contains some noise points, they have little effect on the following steps.
Figure 7. We obtain the dense reconstruction of upper parts of the body. The left figure is our template driven by the current reference frame before deformation. The right figure is an instance obtained through deforming several body parts of our template.

Figure 8. The error map of deforming some body parts of the template.

2.4.2. Deriving Non-Rigid and Rigid Components

To decompose the rigid component, which is the pose-invariant deformation of a particular human shape, and the non-rigid component that arises from the pose in each frame, we combine all instances to derive deformation matrices. Our model deformations are calculated in a way similar to the SCAPE-based methods [6], where the obtained model consists of a generic template mesh model and a set of instance meshes. We use a $3 \times 3$ matrix $D^i_k$ to represent the deformation of each polygon, which is derived by solving the cost function below:

$$\min_{D^i_k, R^i_k} \sum_k \sum_{l=2,3} \rho \| R^i_k D^i_k \hat{v}_{k,l} - u^i_{k,l} \|^2 + \rho \omega_d \sum_{k_1, k_2} \| D^i_{k_1} - D^i_{k_2} \|^2$$

(10)

where $R^i_k$ is a rigid rotation and $\hat{v}_{k,l} = v_{k,l} - u_{k,1}, l = 2, 3$ are two edges of triangle $k$ in our template mesh. Similarly, $u^i_{k,l}$ are two edges of triangle $k$ in instance $j$. $\rho$ is an indicator function. After the triangle $k$ in instance $j$ has been deformed, $\rho = 1$; otherwise, $\rho = 0$. $\omega_d = 1e^{-3}$ is used to not only
prevent large deformation changes among adjacent triangles, but also to reduce the negative influences caused by the noise points in dense reconstruction. Because the deformation matrix \( D^j_k \) and rotation matrix \( R^j_k \) are both unknown, the optimization is nonlinear. Therefore, we optimize the problem by making an initial guess for \( R^j_k \) and then solve for \( D^j_k \) and \( R^j_k \) individually and iteratively. After \( D^j_k \) is obtained, the rotation matrix \( R^j_k \) can be updated again by a twist vector \( \omega \), \( R^j_k_{\text{new}} \leftarrow (I + [\omega] \times) R^j_k_{\text{old}} \), in which \([\cdot] \times\) denotes the cross product matrix. Thus, when \( D^j_k \) is solved, the twist vector \( \omega \) is solved by minimizing:

\[
\min_{\omega} \sum \sum_{k,l=2,3} \rho \| (I + [\omega] \times) R^j_k \hat{u}_{k,l} - u^j_{k,l} \|^2 + \omega_1 \sum_{b_1,b_2} \| \omega_{b_1} - \omega_{b_2} \|^2
\]

where \( b_1 \) and \( b_2 \) denote two neighboring body parts. After alternatively updating \( D^j_k \) and \( R^j_k \) until they converge to a local minima, a set of matrices \( D^j \) is obtained.

Because our general template contains \( K \) points, the deformation matrices \( D^j \) for each instance can be converted to a vector with size \( 9 \times K \), which can be generated from a simple linear subspace: \( D^j = U \beta^j + \mu \). Here, \( \mu \) is the pose-invariant deformation of each person, which is obtained by calculating the mean value of \( D^j \). When the number of instances is sufficient, we can derive the non-rigid component \( \beta^j \) easily by principal component analysis (PCA).

3. Results and Discussion

In the following experiments, we validate our method on a dataset containing a variety of RGB videos of 30 people performing various types of motion. Our system was developed with MATLAB and executed on an Intel i7 CPU at 3.40 GHz with 24 GB RAM. The images in the dataset were all recorded at a resolution of \( 1920 \times 1080 \) using SONY CX260 under conditions illustrated by Figure 9 and processed off line. Each sequence consists of approximately 450 frames. On average, we obtain an instance from every ten frames. To evaluate the accuracy of our method, we also used KinectFusion and a non-rigid reconstruction algorithm to reconstruct 3D models of the same person in the RGB video for comparison. Furthermore, we also compared our method with the system introduced by Xu et al. [30], which can measure body parameters of dressed people with large-scale motion using a Kinect sensor.

Figure 9. Our setup consists of a common camera, which can record the whole body of the people at an appropriate distance.

3.1. Parameters

Some parameters not stated previously in this paper above were set to fixed values throughout our experiments. These parameter values are listed in Table 1.
Table 1. The parameter settings for our experiments.

| Equations   | Values of Parameters |
|-------------|----------------------|
| Equation (4) | \( \omega_s = 5/16 \) \( \omega_j = 1/16 \) \( \omega_m = 5/8 \) |
| Equations (5–7) | \( \theta_s = 60 \) \( \theta_j = 60 \) \( \rho_d = 0.5 \) |
| Equation (8)  | \( \theta_a = 60 \) \( \theta_B = 25 \) |

To determine the values of the parameters in Table 1, we utilized a method similar to the method described in [28]. Although we need to set the values of six parameters listed in the first two rows in Table 1, \( \rho_d \) is set to 0.5 to normalize the motion energy term in Equation (7); therefore, we only need to choose the appropriate values for \( \omega_s, \omega_j, \theta_s, \theta_j \) because of Equation (4). We manually labeled the different body parts in 20 images of the same person in different poses as the ground truth, then varied the values of \( \omega_s, \omega_j, \theta_s \) and \( \theta_j \) and, finally, analyzed the resulting classification accuracy. The parameters are set to the values that maximize the classification accuracy. The settings for the parameters \( \theta_a \) and \( \theta_B \) were determined in the same manner.

3.2. Qualitative Analysis

We show both the recovered poses and shapes across different bodies with various poses in the example, illustrated in Figure 10. Although we do not employ any optimization-based method for the pose parameters or during local searches in the pose space to improve the resolved poses, our method achieves a fairly accurate computational result for the given image sequence. The second row of Figure 10 demonstrates the effectiveness of kinematic classification. Our method requires no assumptions to reconstruct people. As depicted in the third row of Figure 10, the general template appears quite unlike the reconstructed people, i.e., the proposed system can be utilized in a general application without making any previous body measurements. We maintain the human model and continuously improve the results as more images are processed. The parts of the body improving during the process are highlighted with different colors in the third row. At each processing step, the full body model is also improved, because the enhanced result for the individual parts also contributes more details to the overall model's shape. Reconstructed models are driven individually to fit the poses in the images, to show the effectiveness of both the poses and the recovered shapes, as illustrated in the last row of the Figure 10.

![Figure 10](image-url) Some of the results. The first row shows some representative frames of our dataset. The second row shows the 2D kinematic classification. The third row shows the deformation of our template, the deformed parts are marked in different colors. The last row is the model driven by the current poses.
We make a comparison with a state of the art non-rigid reconstruction algorithm proposed by Yu et al. [21], which needs to compute a dense 3D template from a rigid sequence and then conducts the non-rigid deformation of a dynamic object. Obviously, their method requires more user interaction to ensure that the object to be reconstructed is static at the beginning of the frame sequence. The static object is required to compute a template for every scene. In contrast, our general SCAPE model can be applied to all scenes directly without requiring a rigid sequence. In addition, the examples illustrated in their paper are small objects in a closeup view, e.g., small bobby, toy pig, etc. We tested their algorithm on our dataset. The results are shown in Figure 11. Although two high quality templates are obtained through a dense reconstruction algorithm (see the first and fourth figure in the last row in Figure 11), the non-rigid reconstruction obtained is messy (see the last row of Figure 11), where an increasing number of noise points emerge for large performers’ movement. Because their model dose not include a skeleton-driven method, it is arduous for Yu’s system [21] to tackle the large motions of dynamic people. However, precisely because of the motion, our system polishes the reconstruction quality of two different people using the same template (see the first and fourth figure in the second row demonstrated in Figure 11). In addition, our SCAPE model can automatically make inferences about the body parts that are invisible in any single view; however, Yu’s results are limited to reconstructing only the visible parts of a human body. There is no doubt that our system is more effective at reconstructing 3D models of dynamic people through a single RGB camera, even when the view of the object is relatively distant.

3.3. Quantitative Analysis

To our knowledge, no other method can automatically recover the shapes of dynamic people from a single RGB camera; we show only the quantitative comparison between our system and KinectFusion. The model is recovered using KinectFusion for people in a static pose. The template we use contains 12,500 points and 25,000 faces. To precisely measure our reconstruction error, we align the front part of our model to the KinectFusion results using the rigid iterative closest point (ICP) algorithm and map the Euclidean distance of the closest points to the color space to show the reconstruction error intuitively. As Figure 12 shows, the average reconstruction errors are 0.3 cm and 0.2 cm for the first sequence (the first row in Figure 12) and the second sequence (second row in Figure 12), respectively. The maximum reconstruction error is 1.5 cm for all sequences in our dataset. In the first sequence results, the error occurs mainly around the waist, where the clothing is very loose. There are some different poses in the second sequence (the first row in Figure 10). Note that the watertight model shown in the second row in Figure 12 has less reconstruction errors than the first one due to the tighter clothing, which shows the robustness of our system.

Figure 11. Results from two different motion sequences. The middle row is some of the results from our template corresponding to frames of the motion sequences shown in the first row. The last row shows the template required by Yu’s algorithm [21] and some of the results obtained through their system.
The final model derived from our system can be applied to many areas, particularly for virtual fittings. In our experiment, we reconstructed models for 30 different people and measured the body parameters following the method described by Xu [30]. The mean errors of body parameters are calculated and compared to Xu’s results in Table 2. Although Xu’s method utilizes a depth sensor to improve the precision of body parameter measurement, our system is superior to Xu’s in both convenience and precision: the mean errors of chest girth, neck to hip distance and thigh girth are smaller than Xu’s, even though our method uses only a single RGB camera.

Table 2. Comparison of body parameters with Xu’s.

| Errors            | Arm Length | Chest Girth | Neck to Hip Distance | Hip Girth | Thigh Girth |
|-------------------|------------|-------------|----------------------|-----------|-------------|
| Error of Xu’s (cm)| 1.2        | 3.2         | 4.5                  | 3.1       | 2.1         |
| Error of Ours (cm)| 1.4        | 2.3         | 3.7                  | 3.3       | 1.9         |

To test the influences of the number of frames on the quality of reconstructed models, we evaluate the relationship between the mean errors of reconstructed models and the number of frames in Figure 13. Experimental results show that we can obtain high quality reconstructed models when the input sequence contains more than 400 frames.

Figure 13. The average reconstruction error between the model and KinectFusion. The horizontal ordinate represents the number of frames processed. The vertical coordinate represents the average reconstruction error. The average reconstruction error decreases along with the increase of frames.
3.4. Computation

Due to the large amount of computation required for 3D pose estimation, the dense optical flow descriptor, dense CRF and bundle adjustment, our system is time consuming, currently requiring approximately 8 min per-frame, of which the kinematic classification costs approximately 5 min. In our experiments, we find that the most time-consuming steps are pose estimation and kinematic classification, which together consume approximately 90% of the execution time. The complexity of the pose estimation is $O(N)$, where $N$ is the image size. This complexity is analyzed clearly in [23]. According to the implementation of dense CRF detailed in [28], the main steps involved in solving the dense CRF problem consist of a message passing step, a compatibility transform and a local update. The computational bottleneck is message passing, which also has a complexity of $O(N)$ with a high-dimensional filtering algorithm, where $N$ is the image size. Therefore, our system has a total complexity of $O(Nn)$, where $n$ is the number of frames processed.

Still, even though our system is time consuming, it is the first method that can reconstruct a 3D model of a dynamic person from a single RGB camera. Moreover, our system can be parallelized; we will optimize the performance of our system in future work.

4. Conclusions

In this paper, we presented an automatic system to create dynamic human models from single-camera RGB videos. Our system first refines the coarse 2D and 3D poses using kinematic classification, which is aided by human body motion. From the dense reconstruction of the body parts, a personalized parametric model is constructed incrementally by calculating rigid and non-rigid deformation components. Using that personalized model, our system is ready for testing in some real-life applications, such as trying on and fitting virtually. It should be noted that our method requires no special user interactions or any dedicated devices. Moreover, it is particularly user friendly in that people can employ their existing devices (e.g., smart phones) to create accurate 3D models of the human body.

Regarding future works, we will primarily focus on extending our method to capture more finely detailed models and on improving the speed of our system.

Acknowledgments: This work was partially supported by Grant No. 61100111, 61300157, 61201425, 61271231 from the Natural Science Foundation of China. This work was partially supported by Grant No. BE2015152 from the Natural Science Foundation of Jiangsu Province.

Author Contributions: Yao Yu, Yu Zhou and Sidan Du conceived and designed the experiments; Haiyu Zhu performed the experiments; Haiyu Zhu, Yao Yu and Yu Zhou analyzed the data; Haiyu Zhu contributed analysis tools; Haiyu Zhu wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dou, M.; Taylor, J.; Fuchs, H.; Fitzgibbon, A.; Izadi, S. 3D Scanning Deformable Objects with a Single RGBD Sensor. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 493–501.
2. Newcombe, R.A.; Fox, D.; Seitz, S.M. DynamicFusion: Reconstruction and Tracking of non-Rigid Scenes in Real-Time. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 343–352.
3. Newcombe, R.A.; Izadi, S.; Hilliges, O.; Molyneaux, D.; Kim, D.; Davison, A.J.; Kohi, P.; Shotton, J.; Hodges, S.; Fitzgibbon, A. KinectFusion: Real-time dense surface mapping and tracking. In Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Basel, Switzerland, 26–29 October 2011; pp. 127–136.
4. Zeng, M.; Zheng, J.; Cheng, X.; Liu, X. Templateless quasi-rigid shape modeling with implicit loop-closure. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 25–27 June 2013; pp. 145–152.
5. Zhang, Q.; Fu, B.; Ye, M.; Yang, R. Quality Dynamic Human Body Modeling Using a Single Low-Cost Depth Camera. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 676–683.

6. Anguelov, D.; Srinivasan, P.; Koller, D.; Thrun, S.; Rodgers, J.; Davis, J. SCAPE: Shape completion and animation of people. In Proceedings of the ACM Transactions on Graphics (TOG), New York, NY, USA, 31 July 2005; pp. 408–416.

7. Werghi, N. Segmentation and modeling of full human body shape from 3-D scan data: A survey. IEEE Trans. Syst. Man Cybern. Part C. Appl. Rev. 2007, 37, 1122–1136.

8. Gall, J.; Stoll, C.; de Aguiar, E.; Theobalt, C.; Rosenhahn, B.; Seidel, H.P. Motion capture using joint skeleton tracking and surface estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 1746–1753.

9. Huang, C.H.; Boyer, E.; Navab, N.; Ilic, S. Human shape and pose tracking using keyframes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 24–27 June 2014; pp. 3446–3453.

10. Liu, Y.; Gall, J.; Stoll, C.; Dai, Q.; Seidel, H.P.; Theobalt, C. Markerless motion capture of multiple characters using multiview image segmentation. PAMI 2013, 35, 2720–2735.

11. Robertini, N.; de Aguiar, E.; Helten, T.; Theobalt, C. Efficient Multi-View Performance Capture of Fine-Scale Surface Detail. In Proceedings of the 2014 2nd International Conference on 3D Vision (3DV), Tokyo, Japan, 8–11 December 2014; Volume 1, pp. 5–12.

12. Zollhöfer, M.; Nießner, M.; Izadi, S.; Rehmann, C.; Zach, C.; Fisher, M.; Wu, C.; Fitzgibbon, A.; Loop, C.; Theobalt, C.; et al. Real-time non-Rigid Reconstruction using an RGB-D Camera. ACM Trans. Graph. 2014, 33, 156:1–156:12.

13. Li, H.; Vouga, E.; Gudym, A.; Luo, L.; Barron, J.T.; Gusev, G. 3D self-portraits. ACM Trans. Graph. 2013, 32, 187:1–187:9.

14. Tong, J.; Zhou, J.; Liu, L.; Pan, Z.; Yan, H. Scanning 3d full human bodies using kinects. IEEE Trans. Vis. Comput. Graph. 2012, 18, 643–650.

15. Weiss, A.; Hirshberg, D.; Black, M.J. Home 3D body scans from noisy image and range data. In Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, 6–13 November 2011; pp. 1951–1958.

16. Bogo, F.; Black, M.J.; Loper, M.; Romero, J. Detailed Full-Body Reconstructions of Moving People from Monocular RGB-D Sequences. In Proceedings of the the IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, 6–13 November 2011.

17. Guan, P.; Weiss, A.; Bálan, A.O.; Black, M.J. Estimating human shape and pose from a single image. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 1381–1388.

18. Hasler, N.; Ackermann, H.; Rosenhahn, B.; Thorrmählen, T.; Seidel, H.P. Multilinear pose and body shape estimation of dressed subjects from image sets. In Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; pp. 1823–1830.

19. Jain, A.; Thorrmählen, T.; Seidel, H.P.; Theobalt, C. Moviereshape: Tracking and reshaping of humans in videos. In Proceedings of the ACM Transactions on Graphics (TOG), New York, NY, USA, 12 December 2010.

20. Zhou, S.; Fu, H.; Liu, L.; Cohen-Or, D.; Han, X. Parametric reshaping of human bodies in images. In Proceedings of the ACM Transactions on Graphics (TOG), New York, NY, USA, 26 July 2010.

21. Yu, R.; Russell, C.; Campbell, N.; Agapito, L. Direct, Dense, and Deformable: Template-Based non-Rigid 3D Reconstruction from RGB Video. In Proceedings of the IEEE International Conference on Computer Vision (ICCV 2016), Bath, UK, 13 December 2015.

22. Hasler, N.; Stoll, C.; Sunkel, M.; Rosenhahn, B.; Seidel, H.P. A statistical model of human pose and body shape. In Computer Graphics Forum; Wiley Online Library: Munich, Germany, 2009; Volume 28, pp. 337–346.

23. Yang, Y.; Ramanan, D. Articulated pose estimation with flexible mixtures-of-parts. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 20–25 June 2011.

24. Wang, C.; Wang, Y.; Lin, Z.; Yuille, A.L.; Gao, W. Robust estimation of 3d human poses from a single image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 24–27 June 2014; pp. 2369–2376.
25. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Fluids Eng.* 1960, 82, 35–45.
26. Myronenko, A.; Song, X. Point set registration: Coherent point drift. *PAMI* 2010, 32, 2262–2275.
27. Brox, T.; Malik, J. Large displacement optical flow: Descriptor matching in variational motion estimation. *PAMI* 2011, 33, 500–513.
28. Krähenbühl, P.; Koltun, V. Efficient inference in fully connected crfs with gaussian edge potentials. *Adv. Neural Inf. Process. Syst.* 2011, 24, 109–117.
29. Lourakis, M.I.; Argyros, A.A. SBA: A software package for generic sparse bundle adjustment. *TOMS* 2009, 36, 2:1–2:30.
30. Xu, H.; Yu, Y.; Zhou, Y.; Li, Y.; Du, S. Measuring Accurate Body Parameters of Dressed Humans with Large-Scale Motion Using a Kinect Sensor. *Sensors* 2013, 13, 11362–11384.

© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons by Attribution (CC-BY) license (http://creativecommons.org/licenses/by/4.0/).