Human Object Inpainting Using Manifold Learning-Based Posture Sequence Estimation

Chih-Hung Ling, Yu-Ming Liang, Chia-Wen Lin, *Senior Member, IEEE*, Yong-Sheng Chen, *Member, IEEE*, and Hong-Yuan Mark Liao, *Senior Member, IEEE*

***Abstract*—We propose a human object inpainting scheme that divides the process into three steps: 1) human posture synthesis; 2) graphical model construction; and 3) posture sequence estimation. Human posture synthesis is used to enrich the number of postures in the database, after which all the postures are used to build a graphical model that can estimate the motion tendency of an object. We also introduce two constraints to confine the motion continuity property. The first constraint limits the maximum search distance if a trajectory in the graphical model is discontinuous, and the second confines the search direction in order to maintain the tendency of an object's motion. We perform both forward and backward predictions to derive local optimal solutions. Then, to compute an overall best solution, we apply the Markov random field model and take the potential trajectory with the maximum total probability as the final result. The proposed posture sequence estimation model can help identify a set of suitable postures from the posture database to restore damaged/missing postures. It can also make a reconstructed motion sequence look continuous.**

***Index Terms*—Dimensionality reduction, isomap, manifold learning, object completion, video inpainting.**

I. Introduction

VIDEO inpainting is a popular research field in recent years, owing to its powerful capability in video editing and recovering. A number of algorithms for automatic video inpainting have been proposed [1]–[8] in the past few years. Conventional video inpainting methods can be roughly classified into two types: The first type is patch-based [1]–[4], and the other type is template based [5], [6]. In [1], Patwardhan *et al.* proposed a video inpainting technique that makes use of motion information and image inpainting technique together. Motion information is adopted to help find the most suitable patch. In [2], the space–time volume is sliced up into motion manifolds to perform video completion. The proposed manifolds are composed of 2-D patches (one for the spatial dimension and the other for the temporal dimension). These patches cover the entire trajectory of pixels, and the method in [2] applies the approach of Sun *et al.* [27] to inpaint those missing regions. However, these approaches would cause spatial or temporal structure inconsistency artifacts. In [3], Wexler *et al.* adopted a 3-D fix-sized patch as a unit for video inpainting. The value of a missing pixel is estimated by a set of constituent patches, and a multiscale solution is used to speed up the process. In [4], Cheung *et al.* introduced a probabilistic patch model for video inpainting. They use a video epitome method to compress an original video by learning, after that the epitome is used to synthesize data for the damaged areas of a video.

In the template-based video inpainting category, Cheung *et al.* [5] proposed a technique to deal with the problem of missing objects in videos captured by a stationary camera. All available object templates are used to inpaint the foreground. Then, for each missing object, a fixed-size sliding window that covers the missing object and its neighboring templates is used to find the most similar object template. Although the sliding window can help find similar object templates, the inpainting result may be unsatisfactory if the number of postures is insufficient. Furthermore, a good filling position is crucial for an object inpainting process because an inappropriate position may cause visually annoying artifacts. In [6], Jia *et al.* proposed a user-assisted video layer segmentation technique that decomposes an input video into color and illumination videos. A tensor voting technique is then used to address the pertinent spatio–temporal issues in background and foreground. Image repairing is used for background inpainting, and occluded objects are reconstructed by synthesizing other available objects. However, a synthesized object created under this approach does not have a real trajectory; thus, the approach is only suitable for objects with periodic motion.

Although an object can perform a broad variety of movements, the set of typically performed movements is usually located on a latent space that is low dimensional, particularly when the period of object occlusion is not long, where the missing part usually only contains a simple class of movements. Therefore, motion priors can aid in relaxing the ill-posedness of video inpainting by projecting the high-dimensional video data to a low-dimensional manifold learned from training data and then recovering the missing information in the low-dimensional manifold. Ding *et al.* [8] proposed a nonlinear dimension reduction-based video inpainting technique that utilizes local

Manuscript received October 27, 2010; revised February 14, 2011 and May 15, 2011; accepted May 16, 2011. Date of publication May 31, 2011; date of current version October 19, 2011. This work was supported in part by the National Science Council of Taiwan under Grant NSC98-2221-E-007-080-MY3 and in part by the Taiwan E-learning and Digital Archives Program sponsored by the National Science Council of Taiwan under Grant NSC100-2631-H-001-020 and Grant NSC100-2631-H-001-013. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. James E. Fowler.

C.-H. Ling and Y.-S. Chen are with the Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan.

Y.-M. Liang is with the Department of Computer Science and Information Engineering, Aletheia University, Taipei 251, Taiwan.

C.-W. Lin is with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu 300, Taiwan (e-mail: cwlin@ee.nthu.edu.tw).

H.-Y. M. Liao is with the Institute of Information Science, Academia Sinica, Taipei 115, Taiwan, and also with the Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan.

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

Digital Object Identifier 10.1109/TIP.2011.2158228

linear embedding [9] to transform data observed in frames into embedded features in a low-dimensional manifold. Then, the embedded features are organized to form a Hankel matrix, and missing data can be determined by minimizing the rank of the matrix. Finally, the radial basis function (RBF) is used for inverse mapping. Again, the drawback of this method is that it causes blurring and ghost image artifacts if the object's motion is not periodic.

Motion prior models derived from training data have been also successfully applied in applications of marker-free human motion capture and analysis [10]–[12]. Generally, two main classes of motion priors can be identified [12]. The first class utilizes an explicit motion model to guide motion analysis and tracking of body parts. For example, the method proposed in [13] utilizes variable length Markov models (VLMMs) to characterize both the short-term dynamics and long-term history of video data. Similar to the approach in [8] and this paper, the second class learns a low-dimensional posture manifold and performs analysis and tracking in the low-dimensional manifold [14], [15]. The inverse mapping from the low-dimensional manifold to the high-dimensional full body configuration can be accomplished via RBF or locally linear coordination [16]. Although the basic components for dimensionality reduction and inverse mapping are similar, as motion analysis is aimed at tracking of human motion, the key component of object inpainting-recovering missing trajectories in the learned low-dimensional manifold was usually not addressed in these motion analysis works.

Our literature survey shows that most video inpainting algorithms generate artifacts if the object to be inpainted is completely occluded or its motion is not periodic. To void generating such artifacts, a posture sequence estimation process of good accuracy is required for object inpainting. To this end, Xu *et al.* [17] proposed a method for animating animal motions. The model rearranges available animal templates to form a new animal motion sequence by minimizing a predefined energy function. In this paper, rather than using an optimization approach, which is time consuming, we propose a posture sequence estimation method that maintains the continuity of the local motion of an object. The proposed framework consists of three steps: 1) human posture synthesis; 2) graphical model construction; and 3) posture sequence estimation. Human posture synthesis is used to enrich the number of postures in the database, after which all the postures are used to build a graphical model that can predict motion tendency. We also propose two constraints to confine the motion continuity property. The first constraint limits the maximum search distance if a trajectory in a graphical model is discontinuous, and the second confines the search direction in order to maintain the tendency of an object's motion. We perform both forward and backward prediction to derive local optimal solutions. Finally, we apply the Markov random field (MRF) model to compute an overall best solution, and the potential trajectory with the maximum total probability is taken as the final result. The proposed posture sequence estimation model can help identify a set of suitable postures from the posture database to restore damaged/missing postures. It can also make a reconstructed motion look continuous. The advantage of this posture sequence estimation strategy is that it can handle cases such as nonperiodic motion or complete occlusion. These capabilities are powerful because conventional model-based motion prediction methods [10], [18], [19] must use a training process to achieve the same goal.

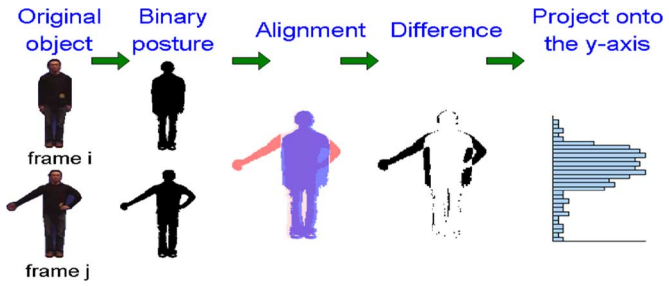


Fig. 1. Projecting posture differences onto the y-axis.

The remainder of this paper is organized as follows: In Section II, we explain how to perform object inpainting based on the proposed posture sequence estimation method. In Section III, we discuss the results of experiments conducted to evaluate the method. Section IV contains some concluding remarks.

II. Human Object Inpainting Using Posture Sequence Estimation

Here, we explain how to perform human object inpainting based on the proposed posture sequence estimation method. As mentioned earlier, the method includes three steps: 1) human posture synthesis; 2) graphical model construction; and 3) posture sequence estimation. We discuss the steps in detail in the following sections.

A. Human Posture Synthesis

The problem of an insufficient number of postures will affect the visual quality of any video sequence generated by a posture-prediction-based approach. To solve the shortage-of-posture problem, we utilize our previous posture synthesis method [7] that was mainly designed for generating synthetic human postures to increase the number of postures. The human posture creation process combines the constituent parts of different available postures to enrich the contents of a posture database. Specifically, the first step performs appropriate segmentation of the postures in the database. To improve the segmentation of a posture, we need to know the amount and speed that each part of the posture moves. For a part that significantly moves and faster, more intermediate postures must be generated to interpolate the gap caused by missing frames. Taking any two postures from the posture database, we use a bounding rectangle to bind each posture; then, we align the two postures, as indicated by the middle part in Fig. 1. Finally, we take the difference between the two postures and project the difference onto the y-axis, as shown on the right-hand side of Fig. 1.

To detect which parts of a human body significantly move, it is necessary to calculate the differences between a posture and all the other postures in the database. All the posture differences are projected onto the y-axis such that the accumulated y-axis component will be like the distribution shown on the right-hand

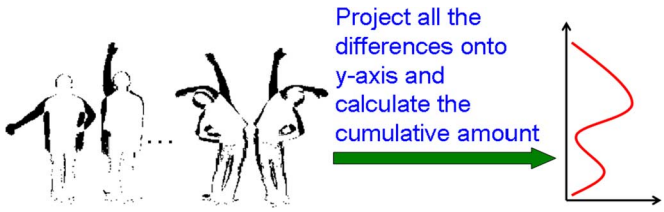


Fig. 2. Projecting all the differences between any two postures onto the y-axis and calculating the cumulative amount.

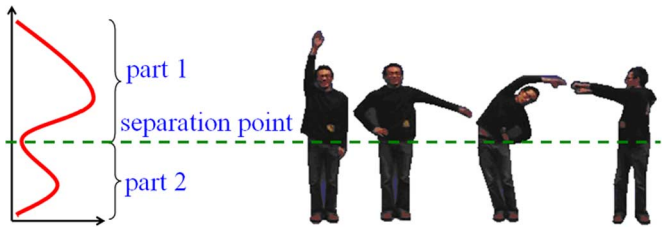


Fig. 3. Constituent components of a posture are partitioned based on the local variance. The dashed line that separates the postures into constituent components can be determined based on the distribution of the local variance shown on the left-hand side of the figure.

Fig. 4. New posture can be synthesized by combining different components (e.g., the torso and the legs).

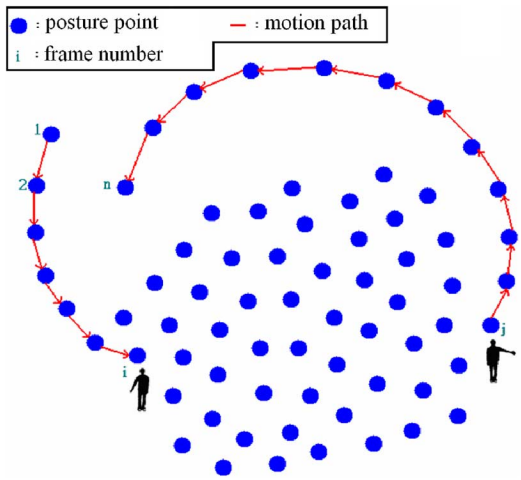


Fig. 5. Graphical model of an object's motion in a low-dimensional manifold. The blue points represent the feature points of the postures, and the red lines connect two feature points whose corresponding postures appear in adjacent frames. In this example, occlusion occurs between frames $i+1$ and $j-1$; hence, we try to find a motion path with l internal points that can be used to link points x_i and x_j.

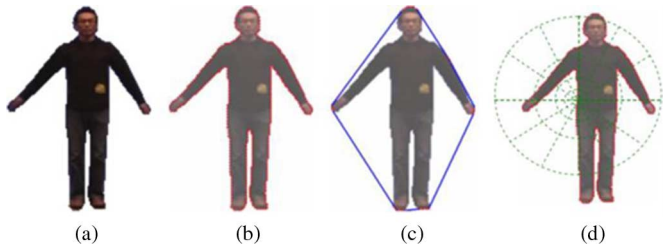


Fig. 6. Extracting the local context of a posture: (a) the object's original posture; (b) the object's silhouette described by a set of feature points; (c) using the convex hull to extract critical reference points; and (d) a shape context mask on a feature point.

side of Fig. 2. Then, from the peaks and valleys of the projected distribution, it is possible to properly segment a posture, as shown by the posture sequence in Fig. 3. From the segmented parts derived from many postures, new postures can be synthesized by combining constituent parts, as shown in Fig. 4.

Note, for the sake of simplicity, in Fig. 1, we assume that the object moves along the direction parallel to the image plane (i.e., the horizontal direction). If the object moves along another direction, the posture difference should be projected to the axis that is orthogonal to the direction of object movements (e.g., the x-axis for vertical movement). The proposed synthesis method is of low complexity and can only synthesize object postures that can be explicitly decomposed into two or more constituent parts. For coping with sophisticated cases in body part localization, one can refer to [20] and [21]. Moreover, the proposed posture synthesis step is to offer more postures with a limited set of configurations of body parts in the posture database to increase the spatio–temporal continuity of a reconstructed trajectory in the low-dimensional manifold, rather than synthesizing arbitrary objects.

B. Graphical Model Construction

After creating synthetic postures, the posture database will contain a lot of postures that can be used to build a graphical model of an object's motion, as shown in Fig. 5. The model provides a simple representation of an object's motion. To obtain such a model, all postures (both synthesized and existing postures) must be projected onto a feature space. Then, we link the postures that appear in adjacent frames in the constructed feature space. After applying the above procedure, we can obtain a graphical representation of the object's motion. To model the distribution of the postures in the feature space, we need to know the distances between distinct postures. We use a shape context descriptor that we developed in a previous work [22], which is a modified version of the descriptor proposed in [23], to compile a detailed description of each posture. The value of the shape context is calculated along the silhouette of the posture. In the posture sequence estimation stage, the values of the shape contexts will be used to compare the degree of similarity between two distinct postures.

To calculate the value of a shape context, the silhouette of a posture must be represented as a set of sampled points $P = \{p_1, p_2, \ldots, p_n\}$, as shown in Fig. 6(b). A convex hull is used to select some critical reference points among the sampled points [see Fig. 6(c)]. Then, for each critical reference point $p_i \in P$, a corresponding local histogram of feature points in N_{bin} bins in a circle of radius r is computed in a log-polar space to represent the local shape context of p_i [see Fig. 6(d)]. The cost of

matching two sampled points that belong to different postures is defined as follows:

$$D(p_i, q_j) = \frac{1}{2}\sum_{k=1}^{N_{\text{bin}}} \frac{\left[h_{p_i}(k) - h_{q_j}(k)\right]^2}{h_{p_i}(k) + h_{q_j}(k)} \tag{1}$$

where $h_{p_i}(k)$ and $h_{q_j}(k)$ denote the kth bin of the two sampled points p_i and q_j, respectively. The value of N_{bin} is empirically set to be 60 for all sequences, and the value of r is determined by an algorithm described in [22]. The best match between two different postures can be accomplished by minimizing the following total matching cost:

$$H(\pi) = \sum_j D\left(p_j, q_{\pi(j)}\right) \tag{2}$$

where π is a permutation of $1, 2, \ldots, n$. Because of the one-to-one matching requirement, shape matching can be considered as an assignment problem that can be solved by a bipartite graph matching method. Therefore, the shape context distance between shapes P and Q can be computed as follows:

$$D_{\text{sc}}(P, Q) = \frac{1}{N_P}\sum_i D\left(p_i, q_{\pi(i)}\right) + \frac{1}{N_Q}\sum_j D\left(p_j, q_{\pi(j)}\right) \tag{3}$$

where N_P and N_Q are the numbers of sample points on shapes P and Q, respectively.

By using the context descriptor proposed in [22], we can calculate the degree of similarity between two distinct postures. Then, based on the similarity scores of the postures, we cluster all the postures in the database by using a nonlinear dimension reduction method called isometric feature mapping (Isomap) [24]. In our application, existing and synthesized postures are regarded as input data points for Isomap, and the distance between two data points is equivalent to the degree of similarity between two corresponding postures. We modify the Isomap algorithm to fit our requirements as follows.

Step 1) Construct a neighborhood graph: If x_i is one of the K-nearest neighbors (K-NN) of x_j, define a graph G that connects data points x_i and x_j. The length of the edge between x_i and x_j is used to measure the degree of similarity between postures o_i and o_j.

Step 2) Compute the shortest paths: Find the shortest path between each pair of feature points in G. Matrix $D_G = (d_G(x_i, x_j))$ contains all the shortest paths between all pairs of data points in G.

Step 3) Construct a d-dimensional embedding: Find eigenvector λ of matrix $\Gamma(D_G)$ (Operator Γ is defined as $\Gamma(D_G) = a_{ij} + a_{**} - a_{*j} - a_{i*}$, where $a_{ij} = -(1/2)(d_G(x_i, x_j))^2$, $a_{i*} = (1/n)\sum_j a_{ij}$, $a_{*j} = (1/n)\sum_i a_{ij}$, and $a_{**} = (1/n^2)\sum\sum a_{ij}$). Then, to derive the final result, we apply classical multidimensional scaling [25] to the matrix of graph distances D_G.

A special feature of Isomap is that it can preserve the distances between data points in each local region during dimension reduction. We exploit this characteristic to preserve the similarity information between postures in each local region of a graphical model and utilize the information to check the motion continuity property between adjacent postures.

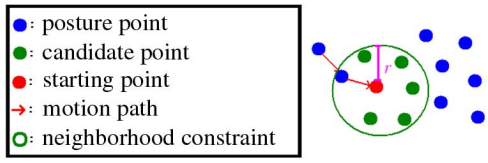


Fig. 7. Neighborhood constraint.

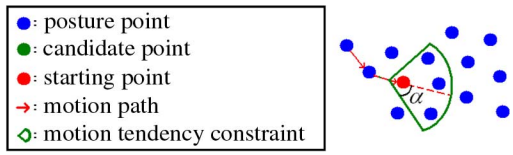


Fig. 8. Motion tendency constraint.

C. Posture Sequence Estimation

Based on the graphical model of an object's motion shown in Fig. 5, we obtain suitable postures to replace damaged/missing postures by finding an approximate path that links data points x_i and x_j in a low-dimensional manifold. Intuitively, a motion path can be reconstructed by taking the shortest path between two nodes or by an optimization process [17], but these two approaches cannot guarantee the smoothness of a recovered motion. To resolve the problem, we propose using two constraints to regulate the motion continuity property in the local region of a graphical model. Specifically, we need a strategy to select a certain number of data points that satisfy the continuous motion constraint. The first constraint limits the search range to within a reasonable neighborhood, as shown in Fig. 7. Therefore, we need to define the search range of the complete trajectory of an object's motion. In the manifold domain, such trajectories are comprised of a number of linked data points (see Fig. 5). To determine the distance between any two consecutive data points on a trajectory, we calculate the shape context difference between their corresponding postures. Then, the maximum distance among all the measured distances is taken as the search range to satisfy the first constraint. Since the search range is circular, we calculate the radius as follows:

$$r = \max_{\forall e_{ij} \text{ on a complete trajectory}} e_{ij} \tag{4}$$

where e_{ij} represents the distance between x_i and x_j on an object's motion trajectory.

The second constraint is introduced to maintain the tendency of an object's motion in each local region. It can be realized by checking the tendency of an object's motion trajectory in a graphical model. In a low-dimensional manifold, a motion trajectory does not significantly change direction in a neighborhood region. Based on this observation, a variance constraint of motion tendency is designed to ensure that the variance of motion tendency stays within a reasonable range (see Fig. 8). In the manifold domain, the complete trajectory of an object's motion is comprised of a number of linked segments, as shown by the red lines in Fig. 5. For the segments indicated by the lines,

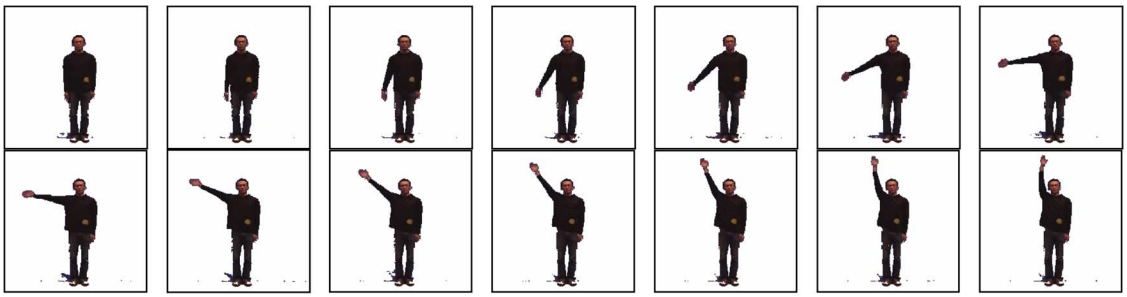


Fig. 9. Some snapshots extracted from test sequence 1.

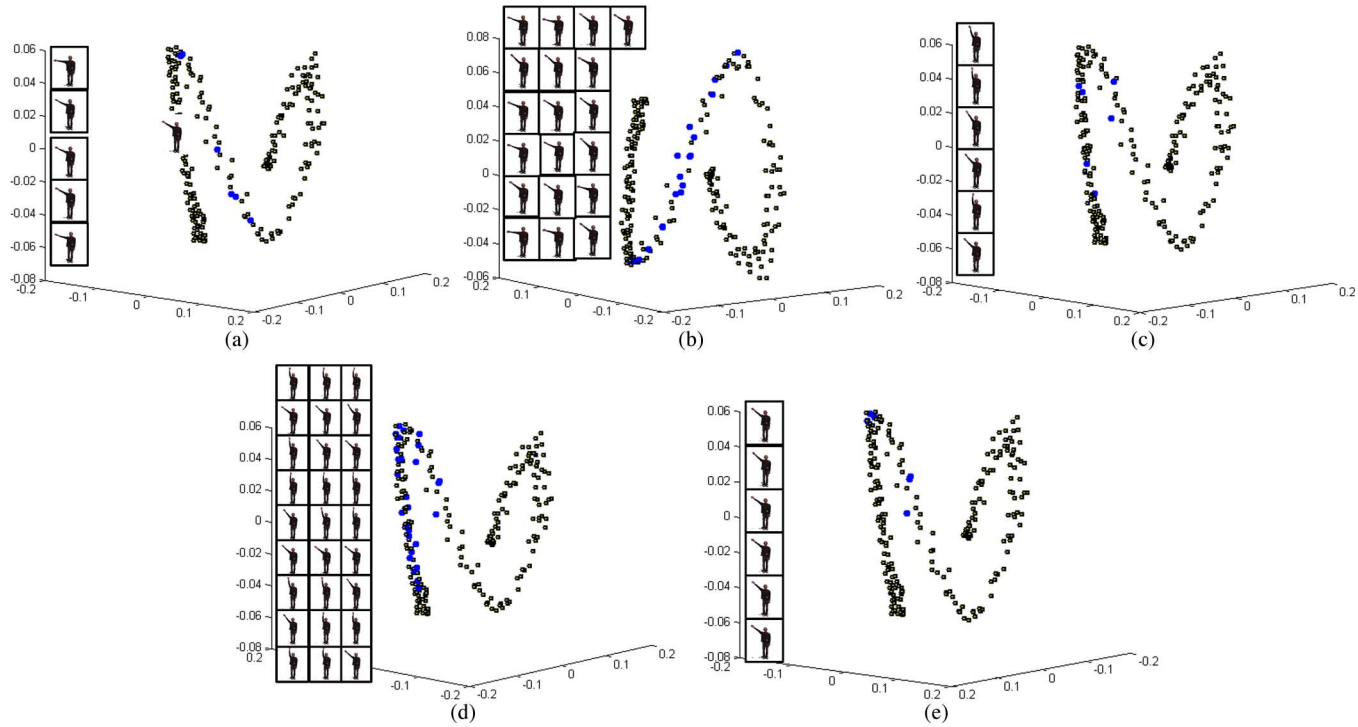


Fig. 10. (a)–(b) Some forward prediction steps, (c)–(d) some backward prediction steps, and (e) the combined results of a two-way prediction at time t.

we compute the change in direction between any two consecutive segments based on the inner product of their corresponding vectors. Among all the computed direction changes, the largest direction change is taken as the maximum allowable angle for direction change. This angle, which is the basis for executing the second constraint, is calculated by

$$\alpha = \max_{\forall \theta_{ijk} \text{ on a complete trajectory}} \theta_{ijk} \tag{5}$$

where θ_{ijk} represents the angle between vectors $\overrightarrow{x_i x_j}$ and $\overrightarrow{x_j x_k}$ on an object's motion trajectory.

The above constraints are designed to maintain the local continuity of an object's motion. To maintain the global motion continuity, we propose a two-way (forward–backward) prediction mechanism. We use three time instants, i.e., $t-1$, t, and $t+1$, to explain how the proposed mechanism operates. In the forward operation, we make a forward prediction on each data point at time $t-1$. The motion tendency constraint and the search range constraint are applied to determine m probable data points at next time instant t. Selected data points m will be used to predict the candidate data points at time $t+1$. We apply the same strategy in the reverse direction and collect related information from $t+1$ to t and from t to $t-1$. Then, we combine the results from the bidirectional processing to obtain the final results for time t. To illustrate the two-way prediction process further, we use a test sequence containing 245 frames. Some snapshots extracted from test sequence 1 are shown in Fig. 9. The candidate points chosen at time instant 19 $(t-1)$ are indicated by the blue dots in Fig. 10(a), and their corresponding postures are shown on the left-hand side of the figure. Those candidate points are used to perform forward prediction. The predicted candidate points at time instant 20 are shown in Fig. 10(b). We apply the same procedure in the reverse direction and generate results from $t=21$ to $t=20$ [shown in Fig. 10(c) and (d)]. The two sets of results are then combined to form the final results, as shown in Fig. 10(e). Table I provides detailed information about the aforementioned processes, including the distance and angle information calculated during the forward and backward prediction steps.

TABLE I
DETAILED INFORMATION DERIVED DURING THE FORWARD–BACKWARD PREDICTION PROCESS

Forward prediction from time instant 19 to time instant 20 (D: distance; A: angle)													
T:19													
T:20	D:0.026 A:13.92	D:0.046 A:31.96	D:0.033 A:40.45	D:0.046 A:14.75	D:0.049 A:27.74	D:0.029 A:5.957	D:0.044 A:8.784	D:0.042 A:19.02	D:0.040 A:28.32	D:0.049 A:19.01	D:0.043 A:44.11	D:0.032 A:31.16	D:0.028 A:15.99
T:19													
T:20	D:0.041 A:37.53	D:0.038 A:8.025	D:0.041 A:8.587	D:0.024 A:8.547	D:0.040 A:6.434	D:0.045 A:24.20	D:0.034 A:4.064	D:0.032 A:38.05	D:0.033 A:35.21	D:0.048 A:22.38	D:0.043 A:12.17	D:0.049 A:22.67	D:0.035 A:24.53
Backward prediction from time instant 21 to time instant 20 (D: distance; A: angle)													
T:21													
T:20	D:0.031 A:26.12	D:0.041 A:2.623	D:0.039 A:19.61	D:0.049 A:4.843	D:0.031 A:15.55	D:0.041 A:26.61	D:0.043 A:29.67	D:0.046 A:18.70	D:0.043 A:13.51	D:0.028 A:13.04	D:0.045 A:4.683	D:0.048 A:5.593	D:0.037 A:13.18
T:21													
T:20	D:0.045 A:42.77	D:0.041 A:43.72	D:0.047 A:31.33	D:0.031 A:49.49	D:0.043 A:48.23	D:0.023 A:1.048	D:0.049 A:33.80	D:0.047 A:23.55	D:0.049 A:9.623	D:0.032 A:14.78	D:0.048 A:5.354	D:0.041 A:5.704	D:0.035 A:0.914

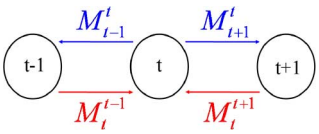


Fig. 11. Example of the MRF process.

Since the motion continuity constraint is only effective on local regions, we use the MRF model to derive global motion continuity. MRF provides a convenient and accurate way to model context-dependent entities, such as image pixels and correlated features. The above modeling can be achieved by characterizing the mutual influences that relate such entities. To predict an object's motion, instead of following the Markov assumption, we assign one node of the Markov network to each time state. Then, the constructed network can reflect statistical dependences. Given a set of data points located at the intervening nodes, every node of a Markov network is statistically independent of other nodes in the network. Since our Markov network does not contain loops, the aforementioned Markov assumption results in simple "message-passing" rules for computing the probability during inference. The data point estimated at node j is

$$c_j^* = \arg\max_{c_j} p(c_j) M_j^{j-1} M_j^{j+1} \tag{6}$$

where c_j denotes the candidate point associated with node j, $p(c_j)$ is the self-probability of candidate point c_j, and M_j^{j+1} is the message derived from node $j-1$ to node j. M_j^{j+1} can be calculated as follows:

$$M_j^{j+1} = \max_{[c_k]} \Psi(c_j, c_{j+1}, c_{j+2}) p(c_{j+1}) \tilde{M}_{j+1}^{j} \tilde{M}_{j+1}^{j+2} \tag{7}$$

where $\tilde{M}_{j+1}^{j}$ is the previous message, which is used to generate M_j^{j+1} by executing (7). M_j^{j+1} includes the probability information of all the candidate data points of node k. The initial $\tilde{M}_{j+1}^{j}$ message is set as a column vector with the initial probability of all the elements associated with node j. Function $\Psi(c_j, c_{j+1}, c_{j+2})$ is defined as follows:

$$\Psi(c_j, c_{j+1}, c_{j+2}) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(\theta-\mu)^2}{2\sigma^2}\right) \tag{8}$$

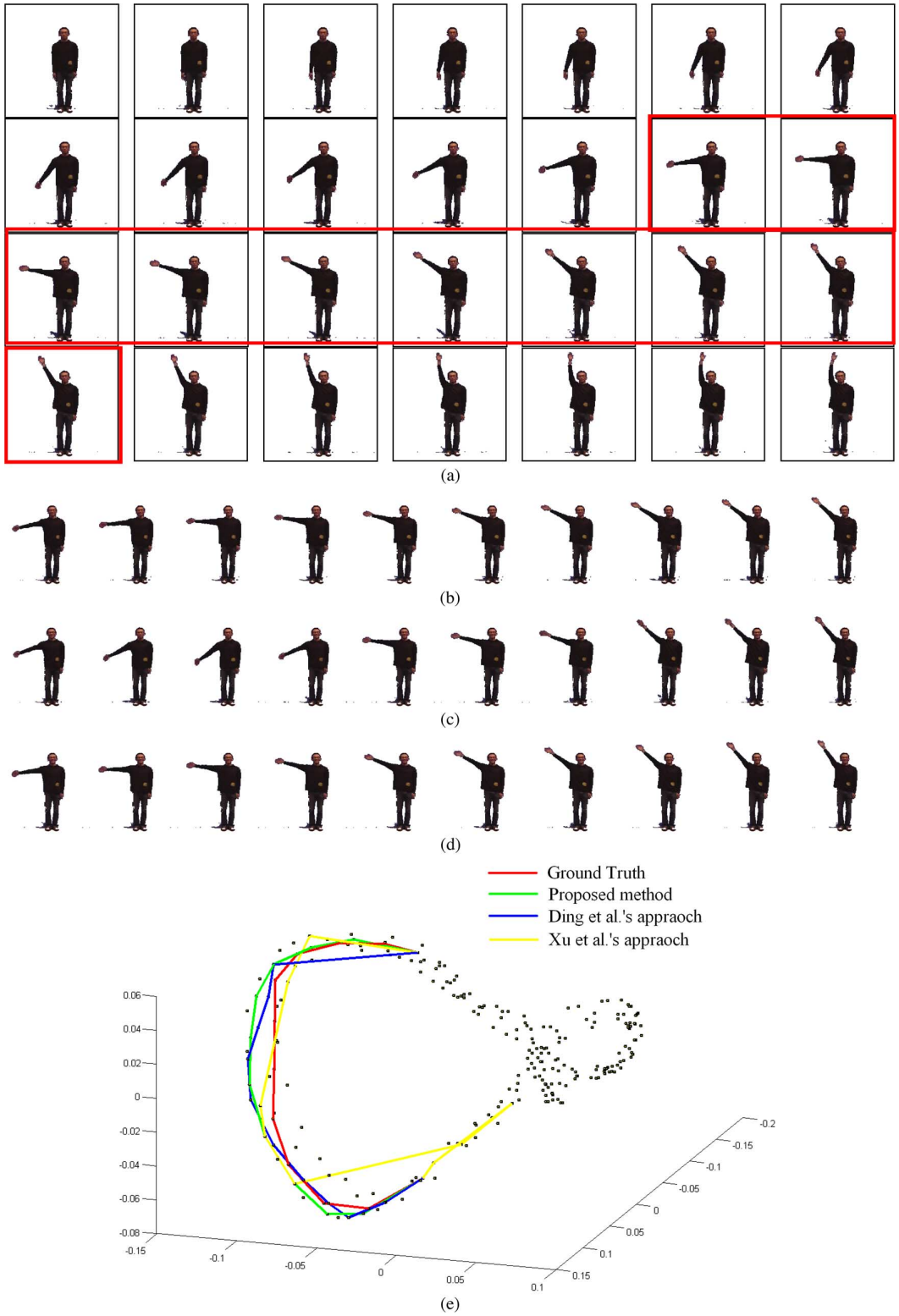


Fig. 12. Experiments on test sequence 1: (a) partial sequence of test sequence 1 in which the red rectangle indicates missing frames; (b) frames reconstructed by the approach in [8]; (c) frames reconstructed by the approach in [17]; (d) frames reconstructed by the proposed approach; and (e) the corresponding trajectory information of predicted object motion generated by the three approaches.

where θ is the angle between vectors $\overrightarrow{c_j c_{j+1}}$ and $\overrightarrow{c_{j+1} c_{j+2}}$ and μ and σ are the mean and standard deviation, respectively, of all angles in a complete trajectory of an object's motion.

To better explain how (6)–(8) find an optimal c_t^*, we use the three nodes shown in Fig. 11 as an example.

Initially, node t receives two messages in the form of a column vector with the initial probabilities of the elements associated with nodes $t-1$ and $t+1$. It then sends the two messages, i.e., M_{t-1}^t and M_{t+1}^t, to nodes $t-1$ and $t+1$, respectively. The messages contain the probability information

TABLE II
COMPARISON OF THE GROUND-TRUTH POSTURES AND THE RECONSTRUCTED MISSING POSTURES (THE PARTS IN BLACK, RED, AND GRAY REPRESENT THE GROUND-TRUTH POSTURES, THE RECONSTRUCTED POSTURES, AND THE PERFECTLY MATCHED PORTIONS, RESPECTIVELY)

Ground-truth											Average
Ding *et al.* [8]											91.4%
	94.7%	93.0%	91.5%	90.7%	91.1%	90.8%	90.6%	90.6%	90.8%	90.6%	
Xu *et al.* [17]											89.1%
	89.8%	87.8%	85.6%	85.8%	89.5%	90.4%	88.0%	93.1%	90.6%	91.2%	
Ours											96.3%
	98.5%	97.7%	96.7%	96.8%	96.5%	95.5%	96.3%	96.4%	92.7%	96.4%	

of all the candidate data points associated with node t. Before the information is sent, it is reordered to form a column vector. On receipt of the information, nodes $t-1$ and $t+1$ respond by sending messages M_t^{t-1} and M_t^{t+1}, respectively, to node t. When each candidate point of node t receives message M_t^{t-1}, it finds a matching point in node $t-1$ as follows:

$$\widehat{p}(c_t) = \arg\max_{c_{t+1}} \Psi(c_{t-1}, c_t, c_{t+1}) p(c_{t-1}) p(c_t) p(c_{t+1}) \quad (9)$$

where $\widehat{p}(c_t)$ is the new self-probability of candidate point c_t, $p(c_t)$ is the previous self-probability of candidate point c_t, and $p(ct-1)$ and $p(ct+1)$ are the probabilities propagated by messages M_t^{t-1} and M_t^{t+1}, respectively. After normalizing the probability value of each candidate point calculated by (9), we obtain a new probability value for each candidate point. Then, node t sends the updated message M_{t+1}^t with the new probability to node $t+1$. Similarly, if node t receives an updated message from node $t+1$, the probability values of all the candidate points of node t are recomputed and sent to node $t-1$. Freeman *et al.* [28] showed that after, at most, one global iteration of (7) on each node of the network, (6) can derive the desired optimal estimate of c_j^* at node j.

III. EXPERIMENTAL RESULTS

To test the effectiveness of the proposed posture sequence estimation method, we performed experiments on eight sequences, wherein part of them were captured with a camcorder and the remaining were grabbed from the Weizmann database [29] and the Internet. In addition to test sequence 1 shown in Fig. 9, we used sequences 2 and 3 to evaluate the proposed method. In the experiments, we first removed several consecutive frames to simulate a real-world situation where objects in a number of consecutive frames were damaged due to packet loss. Then, we applied the proposed posture sequence estimation method to reconstruct the motion of each object. We also compared the performance of our approach with that of the approaches in [8] and [17]. For all the test sequences, the

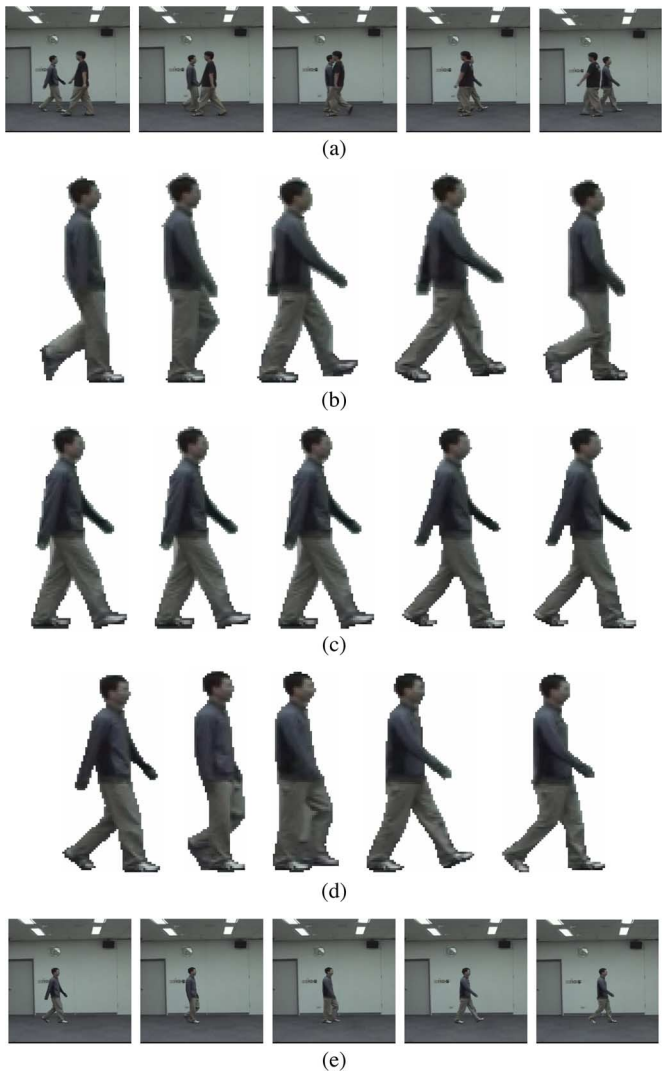


Fig. 13. Experiments on test sequence 2: (a) some snapshots of the occluded object in the test sequence; (b) frames reconstructed by the approach in [8]; (c) frames reconstructed by the approach in [17]; (d) frames reconstructed by the proposed approach; and (e) the inpainting result derived by our approach.

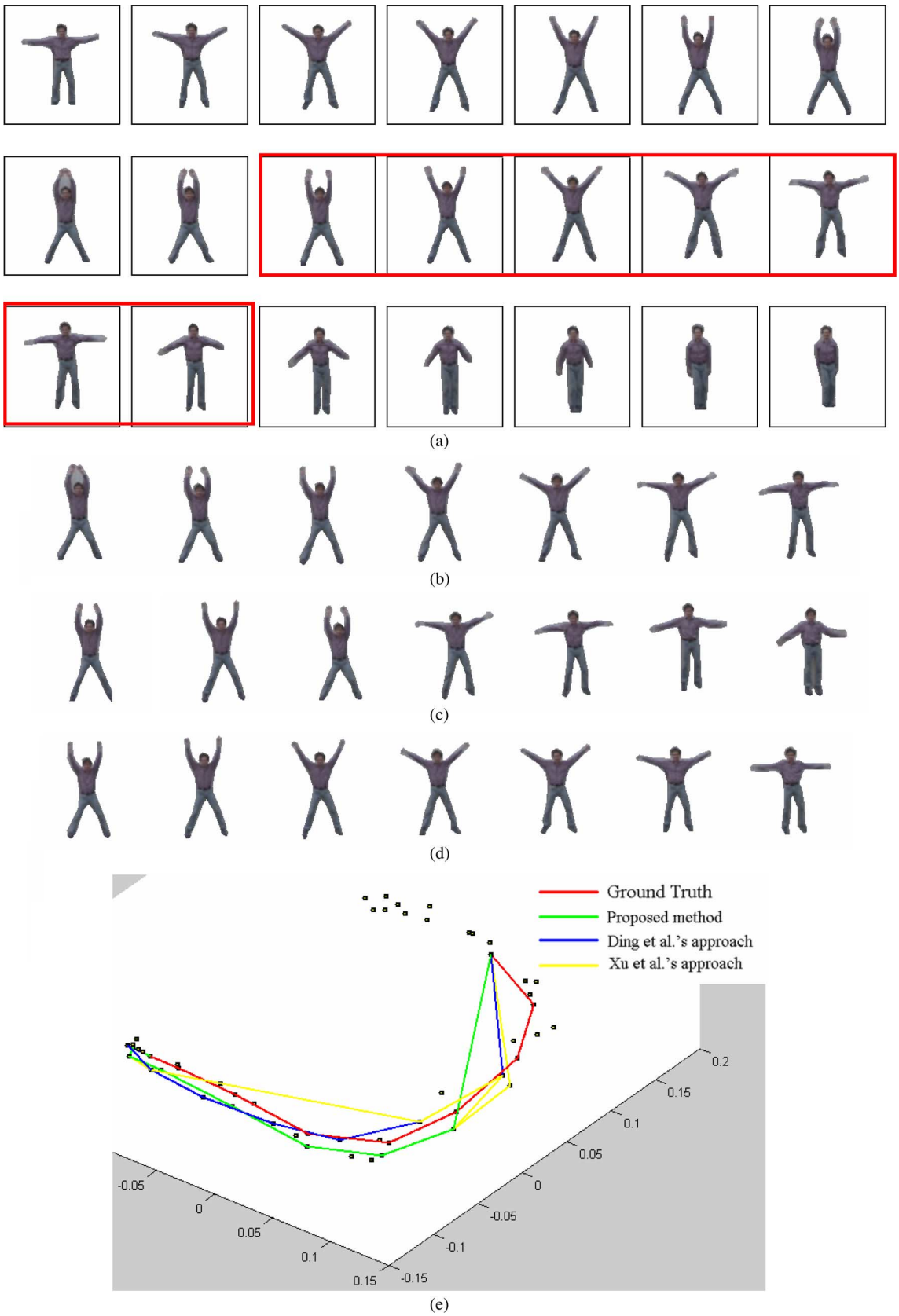


Fig. 14. Experiments on test sequence 3: (a) partial sequence of the test sequence in which the red rectangle indicates the seven missing frames; (b) the frames reconstructed by the approach in [8]; (c) the frames reconstructed by the approach in [17]; (d) the frames reconstructed by the proposed approach; and (e) the corresponding trajectory information of predicted object motion generated by the compared approaches.

proposed method maintained the motion continuity of a reconstructed motion and yielded better results than the compared approaches. For subjective performance comparison, readers can find more test sequences and the complete set of test results,

TABLE III
COMPARISON OF THE GROUND-TRUTH POSTURES AND RECONSTRUCTED MISSING POSTURES (THE PARTS IN BLACK, RED, AND GRAY REPRESENT THE GROUND-TRUTH POSTURES, THE RECONSTRUCTED POSTURES, AND THE PERFECTLY MATCHED PORTIONS, RESPECTIVELY)

Ground-truth								Average
Ding *et al.* [8]								71.3%
	72.7%	76.2%	71.1%	69.2%	72.0%	70.3%	67.7%	
Xu *et al.* [17]								75.7%
	60.6%	94.0%	68.7%	72.8%	73.3%	66.1%	94.3%	
Ours								80.9%
	83.0%	94.0%	81.3%	73.7%	79.7%	77.5%	77.2%	

including the original videos, the videos after object removal, and the inpainted videos, from our project website [31].

In the first experiment, we removed ten of the 245 frames in test sequence 1. Part of the sequence (28 frames) is shown in Fig. 12(a). In the figure, the ten frames that we removed are bounded by the red rectangle. Fig. 12(b)–(d) show the missing sequence that was reconstructed by applying the approaches in [8] and [17] and ours, respectively, and Fig. 12(e) shows the corresponding trajectories reconstructed by the three approaches in the manifold space. Among the trajectories, the red, blue, yellow, and green colors represent the ground-truth trajectory, and the trajectories reconstructed by the approaches in [8] and [17] and the proposed approach, respectively. We observe that the trajectory reconstructed by our approach maintains the best motion continuity, and it is also the smoothest of the three trajectories. Because the proposed posture sequence estimation method is more effective in recovering an object's motion and maintaining motion continuity simultaneously, we conclude that it is more suitable for object inpainting than the compared methods.

Table II details the results of the ground truth and the three compared methods. The top row shows the sequence of missing ground-truth postures, and the second, third, and fourth rows show the missing frames reconstructed by the methods in [8] and [17] and our method, respectively. The black parts of the figures are the ground-truth postures, the gray parts are perfectly matched portions, and the red parts belong to reconstructed postures. We observe that the frames reconstructed by our method are consistently better than those derived by the compared methods.

In the second experiment, we used test sequence 2, which contained 100 frames. In the sequence, two people are walking toward each other, and one person occludes the other in about 20 frames [some of the frames are shown in Fig. 13(a)]. Fig. 13(b)–(d) show the parts of the frames reconstructed by the methods in [8] and [17] and our approach, respectively. From the reconstructed frames, it is apparent that our approach was the most effective in recovering the occluded frames. Using the recovered sequence generated, our approach yielded the best inpainting results among the three compared approaches, as shown in Fig. 13(e).

In the third experiment, we used a video sequence (test sequence 3) from the Weizmann database [29] to evaluate our method. We removed seven of the 55 frames in the sequence. Fig. 14(a) shows part of the sequence (21 frames). The seven frames bounded by the red rectangle were the ones removed before the experiment. Fig. 14(b)–(d) show the missing frames reconstructed by the three approaches, respectively, and Fig. 14(e) shows the trajectories reconstructed by the three approaches in the manifold space.

Table III details the results of the ground-truth method and the three compared methods. The top row shows the sequence of missing ground-truth postures. The second, third, and fourth rows show the missing frames reconstructed by the two methods in [8] and [17] and our method, respectively. The black parts of the figures are the ground-truth postures, the gray parts are perfectly matched portions, and the red portions belong to the reconstructed postures. Note that the first frame reconstructed by the method in [8] covers a broad area (the red area above the head). Only this method may generate such results. In terms of the accuracy of the reconstructed frames, our method reconstructed the most accurate postures overall. However, the method in [17] reconstructed the most accurate posture in the last of the seven missing frames. The match rate was 94.3% compared to that of the ground truth. In contrast, the accuracy of the postures reconstructed by the method in [8] and our method was 67.7% and 77.2%, respectively, compared to that of the ground-truth posture.

As can be observed from the results shown in our demo page [31], since the proposed method uses nonoccluded postures

taken from the same video to completely replace the occluded postures, rather than completing the missing parts of occluded postures, it can avoid the blurring and deformation artifacts, which may be produced by patch-based inpainting approaches. In addition, since in our method the nonoccluded posture sequences for training the MRF models are taken from the same video containing the to-be-inpainted posture sequence, they all have the same frame rate. Therefore, no additional temporal scaling or time warping is required for matching different temporal scales. One shortage of our method is that, since it is an object-based approach, inaccurate object segmentation may lead to visually unpleasant artifacts.

IV. Conclusion

We have proposed a human object inpainting scheme that divides the process into three steps: 1) human posture synthesis; 2) graphical model construction; and 3) posture sequence estimation. In addition, we have defined two constraints on the motion continuity property. The first constraint sets a threshold to limit the maximum search distance, and the second confines the range of the search direction. With the two constraints, the number of possible candidates between any two consecutive postures can be significantly reduced. We then apply the MRF model to perform global matching. The experiment results demonstrate that the proposed approach outperforms two existing state-of-the-art approaches.

References

[1] K. A. Patwardhan, G. Sapiro, and M. Bertalmío, "Video inpainting under constrained camera motion," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 545–553, Feb. 2007.

[2] Y. Shen, F. Lu, X. Cao, and H. Foroosh, "Video completion for perspective camera under constrained motion," in *Proc. IEEE Conf. Pattern Recognit.*, , Hong Kong, Aug. 2006, pp. 63–66.

[3] Y. Wexler, E. Shechtman, and M. Irani, "Space-time completion of video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 3, pp. 463–476, Mar. 2007.

[4] V. Cheung, B. J. Frey, and N. Jojic, "Video epitomes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, San Diego, CA, Jun. 2005, pp. 42–49.

[5] S.-C. S. Cheung, J. Zhao, and M. V. Venkatesh, "Efficient object-based video inpainting," in *Proc. IEEE Conf. Image Process.*, Atlanta, GA, Oct. 2006, pp. 705–708.

[6] J. Jia, Y.-W. Tai, T.-P. Wu, and C.-K. Tang, "Video repairing under variable illumination using cyclic motions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 832–839, May 2006.

[7] C.-H. Ling, C.-W. Lin, C.-W. Su, Y.-S. Chen, and H.-Y. M. Liao, "Virtual contour-guided video object inpainting using posture mapping and retrieval," *IEEE Trans. Multimedia*, vol. 13, no. 2, pp. 292–302, Apr. 2011.

[8] T. Ding, M. Sznaier, and O. I. Camps, "A rank minimization approach to video inpainting," in *Proc. IEEE Conf. Comput. Vis.*, Rio de Janeiro, Brazil, Oct. 2007, pp. 1–8.

[9] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[10] L. Wang, W. Hu, and T. Tan, "Recent developments in human motion analysis," *Pattern Recognit.*, vol. 36, no. 3, pp. 585–601, Mar. 2003.

[11] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," *Comput. Vis. Image Understand.*, vol. 104, no. 2/3, pp. 90–126, Nov./Dec. 2006.

[12] R. Poppe, "Video-based human motion analysis: An overview," *Comput. Vis. Image Understand.*, vol. 108, no. 1/2, pp. 4–18, Oct./Nov. 2007.

[13] F. Caillette, A. Galata, and T. Howard, "Real-time 3-D human body tracking using variable length Markov models," in *Proc. Brit. Mach. Vis. Conf.*, Oxford, U.K., Sep. 2005, pp. 469–478.

[14] A. M. Elgammal and C.-S. Lee, "Inferring 3D body pose from silhouettes using activity manifold learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Washington, DC, Jun. 2004, pp. 681–688.

[15] K. Grauman, S. L. Martin, A. Hertzmann, and Z. Popovic, "Style-based inverse kinematics," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 522–531, Dec. 2004.

[16] Y. W. Teh and S. T. Roweis, "Automatic alignment of local representation," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2002, vol. 15, pp. 841–848.

[17] X. Xu, L. Wan, X. Liu, T.-T. Wong, L. Wang, and C.-S. Leung, "Animating animal motion from still," *ACM Trans. Graph.*, vol. 27, no. 5, pp. 1–8, Dec. 2008.

[18] J. K. Aggarwal and Q. Cai, "Human motion analysis: A review," in *Proc. Nonrigid Articulated Motion Workshop*, Jun. 1997, pp. 90–102.

[19] D. M. Gavrila, "The visual analysis of human movement: A survey," *Comput. Vis. Image Understand.*, vol. 73, no. 1, pp. 82–98, Jan. 1999.

[20] G. Mori, X. Ren, A. A. Efros, and J. Malik, "Recovering human body configuration: Combining segmentation and recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, New York, Jun. 2006, pp. 206–213.

[21] D. Ramanan and C. Sminchisescu, "Training deformable models for localizations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Washington, DC, Jun. 2004, pp. 326–333.

[22] Y.-M. Liang, S.-W. Shih, C.-C. A. Shih, H.-Y. M. Liao, and C.-C. Lin, "Learning atomic human actions using variable-length Markov models," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 1, pp. 268–280, Jan. 2009.

[23] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[24] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.

[25] I. Borg and P. Groenen, *Modern Multidimensional Scaling: Theory and Applications*, 2nd ed. New York: Springer-Verlag, 2005.

[26] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212, Sep. 2004.

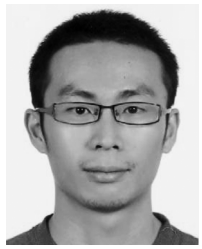
[27] J. Sun, L. Yuan, J. Jia, and H.-Y. Shum, "Image completion with structure propagation," in *Proc. SIGGRAPH*, Los Angeles, CA, 2005, pp. 861–868.

[28] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael, "Learning low-level vision," *Int. J. Comput. Vis.*, vol. 40, no. 1, pp. 25–47, Oct. 2000.

[29] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2247–2253, Dec. 2007.

[30] C.-H. Ling, Y.-M. Liang, C.-W. Lin, Y.-S. Chen, and H.-Y. M. Liao, "Video object inpainting using manifold-based action prediction," in *Proc. IEEE Conf. Image Process.*, , Hong Kong, Sep. 2010, pp. 425–428.

[31] "NTHU Human Object Inpainting Project," NTHU, Hsinchu, Taiwan. [Online]. Available: http://www.ee.nthu.edu.tw/cwlin/inpainting/tip2010_inpainting/object_inpainting.htm



Chih-Hung Ling received the B.S. and M.S. degrees in computer science and information engineering from National Chung Cheng University, Chiayi, Taiwan, in 2003 and 2005, respectively. He is currently working toward the Ph.D. degree in the Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan.

His research interests include computer vision, pattern recognition, and multimedia signal processing.

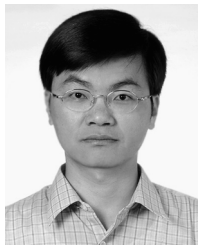
Yu-Ming Liang received the B.S. and M.S. degrees in information and computer education from National Taiwan Normal University, Taipei, Taiwan, in 1999 and 2002, respectively, and the Ph.D. degree from National Chiao Tung University, Hsinchu, Taiwan, in 2009.

From January 2009 to January 2010, he was a Postdoctoral Fellow in the Institute of Information Science, Academia Sinica, Taipei, Taiwan. Since February 2010, he has been with the Department of Computer Science and Information Engineering, Aletheia University, Taipei, Taiwan, as an Assistant Professor. His research interests include computer vision, pattern recognition, and multimedia signal processing.

Chia-Wen Lin (S'94–M'00–SM'04) received the Ph.D. degree in electrical engineering from National Tsing Hua University (NTHU), Hsinchu, Taiwan, in 2000.

He is currently an Associate Professor with the Department of Electrical Engineering, NTHU. He was with the Department of Computer Science and Information Engineering, National Chung Cheng University (CCU), Chiayi, Taiwan, from 2000 to 2007. Prior to joining academia, from 1992 to 2000, he worked with the Information and Communications Research Laboratories, Industrial Technology Research Institute, Hsinchu, where his final post was as a Section Manager. From April 2000 to August 2000, he was a Visiting Scholar in the Information Processing Laboratory, Department of Electrical Engineering, University of Washington, Seattle. He has authored or coauthored over 90 technical papers. He is the holder of more than 20 patents. His research interests include video content analysis and video networking.

Dr. Lin served as the Technical Program Cochair of the IEEE International Conference on Multimedia and Expo (ICME) in 2010 and the Special Session Cochair of the IEEE ICME in 2009. He is an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and the *Journal of Visual Communication and Image Representation*. He has served as a Guest Coeditor of four special issues for the IEEE TRANSACTIONS ON MULTIMEDIA, the *EURASIP Journal on Advances in Signal Processing*, and the *Journal of Visual Communication and Image Representation*. He was a recipient of the 2001 Ph.D. Thesis Award presented by the Ministry of Education, Taiwan, the Young Faculty Award presented by CCU in 2005, and the Young Investigator Award presented by the National Science Council, Taiwan, in 2006. His paper won the Young Investigator Award presented by SPIE VCIP 2005.



Yong-Sheng Chen (M'03) received the B.S. degree in computer and information science from National Chiao Tung University, Hsinchu, Taiwan, in 1993 and the M.S. and Ph.D. degrees in computer science and information engineering from National Taiwan University, Taipei, Taiwan, in 1995 and 2001, respectively.

He is currently an Associate Professor with the Department of Computer Science, National Chiao Tung University. His research interests include biomedical signal processing, medical image processing, and computer vision.

Dr. Chen was the recipient of the Best Paper Award in the 2008 Robot Vision Workshop and the Best Annual Paper Award of the *2008 Journal of Medical and Biological Engineering*.

Hong-Yuan Mark Liao (SM'01) received the B.S. degree in physics from National Tsing Hua University, Hsinchu, Taiwan, in 1981, and the M.S. and Ph.D. degrees in electrical engineering from Northwestern University, Evanston, IL, in 1985 and 1990, respectively.

In July 1991, he joined the Institute of Information Science, Academia Sinica, Taipei, Taiwan, as an Assistant Research Fellow. He was promoted to Associate Research Fellow and then Research Fellow in 1995 and 1998, respectively. Since February 2009, he has been jointly appointed as the Multimedia Information Chair Professor with National Chung Hsing University, Taichung, Taiwan. In August 2010, he was appointed as an Adjunct Chair Professor with Chung Yuan Christian University, Jhongly, Taiwan. He is currently the Division Chair of the Computer Science and Information Engineering Division II, National Science Council of Taiwan. He is also jointly appointed as a Professor with the Department of Computer Science, National Chiao Tung University, Hsinchu. His current research interests include multimedia signal processing, video-based surveillance systems, content-based multimedia retrieval, and multimedia protection.

Dr. Liao started to serve as a member of the Information Forensics and Security Technical Committee of the IEEE Signal Processing Society in January 2010. From 2006 to 2008, he served as the President of the Image Processing and Pattern Recognition Society of Taiwan. From 2004 to 2007, he served as a member of the Multimedia Signal Processing Technical Committee of the IEEE Signal Processing Society. In June 2004, he served as the Conference Cochair for the 5th International Conference on Multimedia and Exposition (ICME) and the Technical Cochair for the 8th ICME held in Beijing, China. In 2011, he will serve as the General Cochair for the 17th International Conference on Multimedia Modeling. He is on the editorial boards of the IEEE SIGNAL PROCESSING MAGAZINE, the IEEE TRANSACTIONS ON IMAGE PROCESSING, and the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY. He served as a Guest Editor of the IEEE TRANSACTIONS ON CIRCUITS and SYSTEMS FOR VIDEO TECHNOLOGY, Special Issue on Video Surveillance (September 2008). He was an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA from 1998 to 2001. He was a recipient of the Young Investigators' Award from Academia Sinica in 1998, the Distinguished Research Award from the National Science Council of Taiwan in 2003, the National Invention Award of Taiwan in 2004, the Distinguished Scholar Research Project Award from the National Science Council of Taiwan, and the Academia Sinica Investigator Award in 2010.