Learning an Intrinsic-Variable Preserving Manifold for Dynamic Visual Tracking

Hong Qiao, Senior Member, IEEE, Peng Zhang, Bo Zhang, and Suiwu Zheng

Abstract—Manifold learning is a hot topic in the field of computer science, particularly since nonlinear dimensionality reduction based on manifold learning was proposed in Science in 2000. The work has achieved great success. The main purpose of current manifold-learning approaches is to search for independent intrinsic variables underlying high dimensional inputs which lie on a low dimensional manifold. In this paper, a new manifold is built up in the training step of the process, on which the input training samples are set to be close to each other if the values of their intrinsic variables are close to each other. Then, the process of dimensionality reduction is transformed into a procedure of preserving the continuity of the intrinsic variables. By utilizing the new manifold, the dynamic tracking of a human who can move and rotate freely is achieved. From the theoretical point of view, it is the first approach to transfer the manifold-learning framework to dynamic tracking. From the application point of view, a new and low dimensional feature for visual tracking is obtained and successfully applied to the real-time tracking of a free-moving object from a dynamic vision system. Experimental results from a dynamic tracking system which is mounted on a dynamic robot validate the effectiveness of the new algorithm.

Index Terms—Feature extraction, robotic visual tracking, visual tracking.

I. INTRODUCTION

THE WORK in this paper is related to two aspects: One is manifold learning, and the other is visual tracking. The related previous work in these two aspects will be reviewed first, and the work in this paper is then stated.

A. Methods of Manifold Learning

The goal of manifold-learning methods is to extract intrinsic variables from high dimensional inputs which lie on or close to a low dimensional manifold. The intrinsic variables are the intrinsic parameters which cause the change of the input samples. For example, in the process of human tracking, the input samples are the images of a human, and then, the intrinsic variables are the orientation and position of the human with respect to the camera and the light influence on the human appearance. These methods try to obtain a set of low dimensional representations which can preserve the local or global geometric characteristics of the original data manifolds.

According to the geometric characteristics which are kept, recent manifold-learning methods can be categorized as local or global approaches. Within local manifold-learning approaches, locally linear embedding (LLE) [2] methods assume that each data point can be linearly reconstructed from their neighbors, and the algorithm preserves local reconstruction weights. The local tangent space alignment method [4] first constructs local tangent coordinates and then aligns them into a global coordinate system and keeps the local tangent structure. Laplacian Eigenmap (LE) methods [5] preserve the local neighborhood relationship. Locality preserving projections (LPPs) [6], [7] is a linearized version of the LE method in which it is assumed that there exists a linear relationship between the high dimensional inputs and the low dimensional representations. However, LPP is different from the principal component analysis or linear discriminant analysis since LPP keeps the local geometric characteristics of a data manifold. Within global manifold-learning approaches, the isometric feature mapping (Isomap) algorithm [3] preserves the pairwise geodesic distances of high dimensional data points to their low dimensional representations. In [8], a unified framework has been proposed to cast most existing manifold-learning methods into a common graph model. Low dimensional representations can then be obtained from the eigendecomposition of different variations of a data adjacency matrix.

The purpose of most traditional manifold-learning approaches is to extract intrinsic variables which are unknown at the beginning. For example, in [1]–[3], the extracted intrinsic variables are the orientation of a human head, light direction, and so on, which are hidden behind high dimensional input samples. The whole process includes three steps.

1) The manifold is built up where the distance between input samples (for example, the images of a human head) is given by the Euclidean distance between the input samples themselves.
2) In order to preserve the local or global geometric characteristics, an optimization problem is constructed.
3) The low dimensional representations are obtained by solving the optimization problem.

However, in this paper, the purpose of manifold learning is to find out a low dimensional space which keeps the continuity of intrinsic variables and also the mapping relationship between high and low dimensional spaces. Then, this mapping relationship is used in tracking. The whole manifold-learning process in this paper includes four steps.

1) The manifold (graph) is built up in the training step where the distance between input samples (for example, the images of a human head and so on) is given by the distance between the intrinsic variables (for example, position, orientation, natural light, and environment light). It should be noted that the intrinsic variables are known in the training step.
2) The explicit mapping relationship between high and low dimensional spaces is obtained.
3) The low dimensional representations are obtained, where, if two input samples have similar values for their intrinsic variables, then the values of their low dimensional representations are close to each other.
4) In tracking with a complicated background, the candidate images are mapped into low dimensional space with the explicit mapping given in the second step. Note that the intrinsic variables are unknown in tracking. According to the locations of the low dimensional representations of these candidate images, the image of the tracked object is determined.

Note that 1) the tracking and training objects must be in the same class, but they can be different objects, for example, tracking and training objects can be different people, and 2) the candidate images may include some background, which is not in the training samples.

A comparison between traditional manifold-learning methods and the new method proposed in this paper is given in Table I.

B. Application of Manifold-Learning Methods

In recent years, manifold-learning algorithms have been applied to several real-world problems. Wang and Suter [9] utilized LPP to train a dimension-reduced feature space where new coming motion sequence can easily be classified via the utilized LPP to train a dimension-reduced feature space where applied to several real-world problems. Wang and Suter [9]

However, in this paper, the purpose of manifold learning is to find out a low dimensional space which keeps the continuity of intrinsic variables and also the mapping relationship between high and low dimensional spaces. Then, this mapping relationship is used in tracking. The whole manifold-learning process in this paper includes four steps.

1) The manifold (graph) is built up in the training step where the distance between input samples (for example, the images of a human head and so on) is given by the distance between the intrinsic variables (for example, position, orientation, natural light, and environment light). It should be noted that the intrinsic variables are known in the training step.
2) The explicit mapping relationship between high and low dimensional spaces is obtained.
3) The low dimensional representations are obtained, where, if two input samples have similar values for their intrinsic variables, then the values of their low dimensional representations are close to each other.
4) In tracking with a complicated background, the candidate images are mapped into low dimensional space with the explicit mapping given in the second step. Note that the intrinsic variables are unknown in tracking. According to the locations of the low dimensional representations of these candidate images, the image of the tracked object is determined.

Note that 1) the tracking and training objects must be in the same class, but they can be different objects, for example, tracking and training objects can be different people, and 2) the candidate images may include some background, which is not in the training samples.

A comparison between traditional manifold-learning methods and the new method proposed in this paper is given in Table I.

B. Application of Manifold-Learning Methods

In recent years, manifold-learning algorithms have been applied to several real-world problems. Wang and Suter [9] utilized LPP to train a dimension-reduced feature space where new coming motion sequence can easily be classified via the k-nearest neighbor classifier. Chen et al. [10] used Isomap and LLE to resample and interweave face data so that the training samples can distribute more evenly in order to train more accurate classifiers. Bachmann et al. [15] used Isomap to construct manifold coordinates for hyperspectral imageries to achieve classification and compressed expression. In [16], a dynamic appearance manifold of motion objects was learned via LLE and applied to learn simple gestures and facial expressions. Wang et al. [17] used Isomap to learn the underlying structure of visual objects to enhance contour tracking under the framework of Bayesian tracking.

In the aforementioned works, manifold-learning methods have been successfully applied to real-world problems. However, none of them is concerned with directly applying manifold-learning methods to visual tracking with real complicated backgrounds, which is very important in practice.

C. Methods of Robotic Visual Tracking

So far, robotic visual tracking is a very important part in the object-tracking field. This paper focuses on tracking a noncooperative person on a dynamic robotic system using head information. In literature, there exist a variety of head-tracking algorithms. In general, the algorithms can be classified as follows.

1) Deterministic methods. Birchfield [18] alternated two modules to get a robust head-tracking system: One is matching the gradients along the head’s boundary, and the other is matching the color histogram of the head’s interior. Triesch and von der Malsburg [19] integrated motion detection, color, position prediction, shape, and contrast cues together to find the right position of a human’s head. An et al. [20] proposed a multiview head-tracking algorithm with three models: head-shape model, face-skin model, and upper-body-color model. The sum of all three models’ matching scores is used to evaluate the similarity of the tracking result.

2) Statistic methods. Isard and Blake [21] combined the color blob with a contour model in a particular filter tracking frame. Spengler and Schiele [22] proposed a multiple-hypothesis tracking method by integrating multiceps into the CONDENSATION [23] algorithm. Kwolek [24] used color and ellipse fitting in a particle filter for head tracking. The product of the intensity gradients and color histogram similarity is used as the scores of the observation. Wu and Huang [25] proposed a novel switching scheme based on a factorial graphical model where the color and shape distributions are adapted online in a coinference way. Li et al. [26] proposed a multiview adaptive fusion algorithm for human tracking. When fusing the different cues, fuzzy logic is used to adjust each cue weight in the observation according to its associated reliability in the past frame.

Existent methods mainly use physical features for head tracking. As far as we know, no work has considered tracking noncooperative persons by using the low dimensional representations which are obtained by preserving the continuity of intrinsic variables.

D. Work in This Paper

The contributions of this paper can be summarized as follows.

1) From the algorithm-design point of view, the method proposed in this paper is the first one which can be used for tracking a free-moving object from a dynamic vision system based on manifold-learning algorithms. The new method is different from traditional manifold-learning approaches. The new method is different from traditional manifold-learning approaches. The new method is different from traditional manifold-learning approaches.
TABLE I

COMPARISON BETWEEN TRADITIONAL MANIFOLD-LEARNING METHODS AND THE PROPOSED NEW METHOD

<table>
<thead>
<tr>
<th>Traditional manifold learning methods</th>
<th>The proposed new method</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. Purpose</td>
<td>Find out a low dimensional space which keeps the continuity of intrinsic variables and the mapping relationship which is used in tracking.</td>
</tr>
<tr>
<td>b. Process</td>
<td>(1) The manifold is built up in the training step where the distance between input samples is given by the distance between the intrinsic variables which are known in this step. (2) The explicit mapping relationship between high dimensional space and low dimensional space is obtained. (3) The low dimensional representations are obtained. (4) In tracking, the candidate images are mapped into low dimensional space with the explicit mapping given in the second step. The image of tracked object is determined according to the locations of low dimensional representations of the candidate images.</td>
</tr>
<tr>
<td>(1) The manifold is built up where the distance between input samples is given by their pairwise Euclidean distance.</td>
<td>(2) In order to preserve the local or global geometric characteristics, an optimization problem is constructed. (3) The low dimensional representations are obtained by solving the optimization problem.</td>
</tr>
</tbody>
</table>

II. IVPML

In this section, we first give a list of notations used in this paper (Section II-A) and then present details of the Intrinsic Variable Preserving Manifold Learning (IVPML) method.

1) The manifold where data points lie is built up based on intrinsic variables (Section II-B).

2) The manifold-learning algorithm which preserves the continuity of intrinsic variables is presented. Furthermore, a discussion is given on using the manifold-learning results to map the new coming data points to the low dimensional space (Section II-C).

A. Notations

Throughout this paper, all data points are in the form of column vectors. Data sets, as well as manifolds, are denoted by capital curlicue letters. Matrices are expressed using normal capital letters. Data vectors are represented using lowercases, and scalars are given by Greek letters. The norm used in this paper refers to the $L_2$ norm. The main notations used in this paper are shown in Table II.

Remark 2.1: In order to make Table II more clear, we explain some notations in Table II and their relationships as follows.

1) $u_i$ represents the intrinsic variables which are the intrinsic parameters that cause the change of the input samples. For example, in the process of human tracking, the input samples are the images of a human, and then, the intrinsic variables are the orientation and position of the human with respect to the camera and the light influence on the human appearance. In traditional manifold-learning methods, $u_i$ is unknown in the manifold-building step. In our method, $u_i$ is known in the training step and unknown in tracking.

2) $y_i$ is obtained through the dimensionality reduction of $x_i$. In traditional manifold-learning methods, $\{y_i\}$,
Fig. 1. Small change of intrinsic variables can cause the values of each pixel in the image to change greatly. The two samples can be regarded as far from each other by conventional manifold-learning methods. (a) Slight change of light direction. (b) Slight rotation.

\[ i = 1, 2, \ldots, N, \text{ is obtained to discover the geometric structure of } \{ u_i \}, \quad i = 1, 2, \ldots, N. \text{ In our method, } y_i \text{ is obtained to give the low dimensional representation of } x_i \text{ in the space which keeps the continuity of intrinsic variable } u_i. \]

### B. Manifold-Building Process

The current manifold-learning approaches have achieved great success, particularly in finding out intrinsic variables behind complicated high dimensional data and exploring the geometric structure of data inputs. They have many practical applications.

However, the manifold in current methods is built up where the distance between input samples (for example, the images of a human head) is determined by the Euclidean distance between themselves. Therefore, these manifolds may not be used for tracking problems, which are very important topics in computer vision.

In the process of tracking, the values of intrinsic variables behind successive images of the tracked object are close to each other. However, successive images of the tracked object may not be close to each other according to the Euclidean distance. For example, two such examples are shown in Fig. 1,1 where slight change of light direction (intrinsic variable) and slight rotation (intrinsic variable) may cause the value of each pixel in the image to change greatly. Then, the Euclidean distance between the two samples can be large. In this case, these two samples caused by a slight change of intrinsic variables may not be neighbors in the ambient space.

In order to keep the adjacency relationship among intrinsic variables, in this paper, at the first step, we build up a new manifold based on the relationship among the intrinsic variables of input samples.

Assume that \( x_i \) and \( x_j \) are input samples, the intrinsic variable of \( x_i \) is \( u_i \), and the intrinsic variable of \( x_j \) is \( u_j \). Here, \( u_i = (u_i^1, u_i^2, \ldots, u_i^d) \) and \( u_j = (u_j^1, u_j^2, \ldots, u_j^d) \). In order to explain the new manifold-building method clearly, several definitions are given as follows.

**Definition 2.1:** \( x_i \) and \( x_j \) are said to be **adjacent** if \( \| u_i - u_j \|_w < \varepsilon \), where \( \varepsilon \) is a constant and \( \| u_i - u_j \|_w = \sqrt{\sum_{k=1}^d w_k (u_i^k - u_j^k)^2} \) with \( w_k \), \( k = 1, 2, \ldots, d \), being real positive constants. Another way to define adjacency is that \( x_i \) and \( x_j \) are said to be adjacent if \( u_i \) is among the \( k \) nearest neighbors of \( u_j \) or \( u_j \) is among the \( k \) nearest neighbors of \( u_i \).

In general, the distance \( d_{ij} \) between \( x_i \) and \( x_j \) is given by

\[
d_{ij} = \begin{cases} \| u_i - u_j \|_w, & \text{if } x_i \text{ and } x_j \text{ are adjacent} \\ \infty, & \text{otherwise.} \end{cases}
\]

Similarity \( S_{ij} \) between two inputs \( x_i \) and \( x_j \) is then defined as

\[
S_{ij} = \begin{cases} \exp\left\{-d_{ij}/\sigma^2\right\}, & \text{if } d_{ij} \neq 0 \\ 0, & \text{if } d_{ij} = \infty \end{cases}
\]

where \( \sigma \) is a scaling factor which is set according to real problems. Then, variation in intrinsic variables can be reflected in the change of similarities.

### C. Manifold-Learning Process

With adjacency and similarities defined earlier, the next step is to compute the low dimensional representations of input samples. In this paper, we use LPP [6], [7] to achieve this goal and obtain an explicit expression on mapping between the high and low dimensional representations. Some kernel methods [12]–[14] have been used for this purpose. However, the computational cost with kernel methods is high, which prohibits their use in real-time systems. Furthermore, these approaches only work when the mapping relationship between the high and low dimensional spaces matches one kernel function, and input samples must lie on the manifold rather than close to but off the manifold. In order to keep the completeness, we briefly introduce LPP as follows.

Formally, we are trying to find a set of low dimensional representations \( Y \) such that the following optimization problem is satisfied:

\[
\min_{\{y_i\}} \sum_{i,j=1}^{N} \| y_i - y_j \|^2 S_{ij} \quad \text{s.t.} \quad \sum_{i=1}^{N} D_{ii} y_i y_i^T = I_m.
\]

Here, \( S_{ij} \) is defined in (2) and acts as a penalty on pairwise distance \( d_{ij} \) and \( D_{ii} = \sum_{j=1}^{N} S_{ij}, \quad i = 1, 2, \ldots, N \). For two samples \( x_i \) and \( x_j \) which are close to each other on the new manifold, \( S_{ij} \) is large. To minimize the objective function in (3), \( \| y_i - y_j \| \) will be small if the weight \( S_{ij} \) associated with \( \| y_i - y_j \| \) is large. Therefore, \( y_i \) and \( y_j \) are close to each other in the low dimensional space. The purpose of the constraint is to remove an arbitrary scaling factor of the low dimensional representations [6].

By a direct algebraic calculation, it can be seen that

\[
\sum_{i,j=1}^{N} \| y_i - y_j \|^2 S_{ij} = 2 \sum_{i=1}^{N} \left( y_i^T \sum_{j=1}^{N} S_{ij} y_j \right) - \sum_{i,j=1}^{N} y_i^T y_j S_{ij}
\]

\[
= 2 \text{Tr}(YL^TY^T)
\]

where \( L = D - S \) with \( S = (S_{ij}) \) and \( D \) is a diagonal matrix with diagonal entries to be \( D_{ii}, \quad i = 1, 2, \ldots, N \). Now, the
optimization problem can be written as
\[
\min_Y \quad \text{Tr}(YLY^T) \\
\text{s.t.} \quad YDY^T = I_m. \tag{5}
\]

In LPP, it is assumed that there exists a linear projection between \(X\) and \(Y\), i.e., there exists an \(n \times m\) projection matrix \(U\) such that, for any \(x_i, i = 1, 2, \ldots, N\), its low dimensional representation \(y_i\) satisfies
\[
y_i = U^Tx_i. \tag{6}
\]

Substituting (6) into (5) yields
\[
\min_U \quad \text{Tr}(U^TXLY^TU) \\
\text{s.t.} \quad U^TXDY^T = I_m. \tag{7}
\]

Equation (7) is used to find a subspace spanned by \(m\) directions such that similarities in the input samples can be preserved under the linear projection assumption.

From (6), it can be found that the \(m\) directions are just the columns of \(U\). Let \(P = XLY^T\) and \(Q = XDX^T\). Then, (7) becomes
\[
\min_U \quad \text{Tr}(U^TPU) \\
\text{s.t.} \quad U^TQU = I. \tag{8}
\]

The columns of \(U\) are the eigenvectors of the following eigenvalue problem corresponding to the \(m\) smallest eigenvalues:
\[
P\hat{u}_i = \lambda Q\hat{u}_i, \quad \hat{u}_i^T\hat{u}_j = \delta_{ij}, \quad i, j = 1, 2, \ldots, m. \tag{9}
\]

Here, \(\hat{u}_i\) is the \(i\)th column of \(U\).

To avoid unnecessary translations, \(X\) is centered at the origin. Then, the columns of \(U\) are the eigenvectors corresponding to the first to the \(m\)th smallest eigenvalues of (9).

After the projection matrix \(U\) is computed, the low dimensional representation \(y_{new}\) of a new coming sample \(x_{new}\) can be easily obtained as follows:
\[
y_{new} = U^Tx_{new}. \tag{10}
\]

We end this section by combining the new manifold and manifold learning via LPP together to form the IVPML algorithm.

**Algorithm 1: IVPML Algorithm**

Input: Training data set \(X\)

Step 1. Construct the adjacency relationship among data inputs.

Step 2. Use (1) to compute pairwise variable distances.

Step 3. Compute pairwise similarities using (2) and form the similarity matrix \(S\).

Step 4. Form matrices \(P\) and \(Q\) according to Section II-C.

Step 5. Solve the generalized eigenvalue problem (9) and get the projection matrix \(U\).

Step 6. Low dimensional representations are given by \(Y = U^TX\).

III. VISUAL-HEAD-TRACKING ALGORITHM WITH COMPLICATED BACKGROUNDS USING IVPML

In the procedure of visual head tracking with complicated backgrounds, continuous images which contain human heads are data inputs. Tracking using a dynamic platform with complicated backgrounds is a very important problem in pattern recognition. The main difficulties are caused by the following factors.

1) When the dynamic platform, which the vision system is attached to, moves, both the tracked objects (such as a human head) and the background change greatly in the images.

2) Since the human can move arbitrarily, the projection of the human image on the vision system changes continuously. It is hard to find a constant feature suitable for different projections.

In this section, we apply the IVPML algorithm presented in Section II to real tracking with complicated backgrounds. The diagram of the new tracking algorithm based on IVPML is shown in Fig. 2 with the details being stated in the following sections.

A. Input-Sample Collection

In our experiments, the intrinsic variables of a human head during movement are considered to be horizontal rotation \(r_h\) and vertical rotation \(r_v\), where \(r_h \in [0, 2\pi]\) and \(r_v \in [-\pi/4, \pi/3]\). Horizontal rotation is around a vertical upward axis, which is through the geometric center of the head.
rotation is around a horizontal axis which is also through the geometric center of the head and points to the right side of the front face. Therefore, the intrinsic variable \( u = (r_h, r_v) \). Then, head images with continuously altering \( u \) along \( r_h \) and \( r_v \) are collected. The head images are evenly distributed along the \( r_h \) and \( r_v \) axes. Then, the data set containing head images with two intrinsic variables is obtained. The process of training a sample collection with \( r_v = 0 \) is shown in Fig. 3. During the collection procedure, the object person fixes his head orientation such that \( r_v \) is constant. Then, he rotates with a constant speed. The vision system is fixed in front of the person.

After the proper training images are collected, only the smallest rectangle containing the image of a head is kept, and the other part of the image is discarded. A default white background is used in the image-collection step. Second, head images are transformed into gray levels to throw off colors. Third, the head images are resized to fit the window size in the tracking algorithm.

**B. Training Process to Obtain the Mapping Relationship**

After collecting head images, the manifold based on the intrinsic variables \( u = (r_h, r_v) \) is built, and dimensionality reduction is achieved using the IVPML algorithm presented in Section II.

The first step is to construct the adjacency relationship among input samples. Since \( r_h \in [0, 2\pi] \) and \( r_v \in [-\pi/4, \pi/3] \), the human head images are distributed on a manifold which looks like a cylinder surface. \( r_h \) is ordered as \( r_{h1}, r_{h2}, \ldots, r_{hH} \) according to their values from 0 to \( 2\pi \), and \( r_v \) is ordered as \( r_{v1}, r_{v2}, \ldots, r_{vV} \) according to their values from \(-\pi/4\) to \( \pi/3\). In this paper, \( H = 120 \) and \( V = 8 \). We use the first method described in Definition 2.1 to determine the adjacency relationship among head images with \( w_1 = 1, w_2 = 0.04, \) and \( \varepsilon = \pi/50 \), i.e., two head images \( I(r_{h1}, r_{v1}) \) and \( I(r_{h2}, r_{v2}) \) are
adjacent if  \( \sqrt{(r_{hi} - r_{hk})^2 + 0.04(r_{vj} - r_{vl})^2} < \pi / 50 \). Since the intrinsic variables are evenly distributed on a rectangular grid aligned with the \( r_h \) and \( r_v \) axes, the aforementioned definition of adjacency is equivalent to the one that two head images \( I(r_{hi}, r_{vj}) \) and \( I(r_{hk}, r_{vl}) \) are adjacent if and only if \( r_{hk} \) is the smallest one but larger than \( r_{hi} \) or the largest one but smaller than \( r_{hi} \) while \( r_{vl} = r_{vj} \), or \( r_{vl} \) is the smallest one but larger than \( r_{vj} \) or the largest one but smaller than \( r_{vj} \) while \( r_{hk} = r_{hi} \). Therefore, the head images \( I(r_{hi}, r_{vj}), \ i = 1, 2, \ldots, H, \ j = 1, V \) have three adjacent neighbors, and the others have four adjacent neighbors. Then, pairwise distances \( \{d_{ij}\} \) defined by (1) are estimated. Similarities \( \{S_{ij}\} \) are obtained along with the computation of \( \{d_{ij}\} \).

In the second step, the similarity matrix \( S = (S_{ij}) \) is used to train the mapping relationship \( U \) as described in Section II.

C. Tracking Algorithm

The key problem in tracking is to determine where the object should be in the next frame according to the current state of the object and the new frame.

Assume that \((x_t, y_t)\) are the coordinates of the center of the detection window for the human head in Frame \( t \). Then, in Frame \( t + 1 \), positions of candidate windows are acquired by increasing/decreasing \((x_t, y_t)\) with equal intervals.

Assume that \( I_t \) is the detected head image from Frame \( t \) and that \( I_{t+1}^c \) is a candidate head image from Frame \( t + 1 \). Then, their low dimensional representations are computed as follows:

\[
\begin{align*}
y_t &= U^T I_t \\
y_{t+1}^c &= U^T I_{t+1}^c.
\end{align*}
\]

In Frame \( t + 1 \), the image which has the shortest distance to \( I_t \) in the low dimensional space is considered as the optimal head image.

IV. EXPERIMENTAL PROCESS AND RESULTS

In this section, we use the tracking algorithm presented in Section III to obtain both input images for training and the low dimensional representations. We also analyze the stability of the low dimensional representations. The tracking algorithm is then used in the practical tracking process, where a security robot aims to track noncooperative persons with the new tracking algorithm (see Fig. 4).

It demonstrated that, using the IVPML algorithm proposed in this paper, tracking based on a dynamic vision system with
a complicated background is fast and reliable. It should be noted, however, that it is difficult to use conventional manifold-learning methods to achieve this goal.

A. Data Collection and Training

Input samples are collected based on Section III-A, where $H = 120$ and $V = 8$. Some examples are shown in Fig. 5. The manifold is built according to the algorithm described in Section III-B. The structure of the manifold is shown in Fig. 6. Eight sequences of head horizontal rotations are denoted by Seq1, Seq2, Seq3, Seq4, Seq5, Seq6, Seq7, and Seq8 according to $r_v^1 - r_v^8$.

The low dimensional representations obtained by the IVPML algorithm are shown in Fig. 7. It should be noted that eight sequences of horizontal rotations coincide with each other. In order to test the stability of the low dimensional representations, experiments with leave-one-out are carried out. In Fig. 8(a), the left one represents the low dimensional representations obtained by IVPML using data samples without the first horizontal rotation (Seq1). The right one represents the low dimensional representation of Seq1 with the projection matrix obtained by using the seven other sequences [the left one in Fig. 8(a)]. Ideally, the projection of Seq1 should be close to the trained manifold, and the experimental result also supports this. Similarly, Fig. 8(b)–(h) shows the results of IVPML with leaving Seq2–Seq8, respectively.

B. Experiments on a Dynamic Robotic System

We now apply the tracking algorithm described in Section III-C and the low dimensional representations given in Section IV-A to experiments on tracking. Precisely, a head-tracking process with the tracking algorithm is applied to a dynamic robotic system with a complicated background.

The height of the robot is similar to the height of an adult. Its size is of 60 cm by 60 cm by 170 cm, and its weight is 50 kg. The robot can move and rotate on flat surface freely and can avoid obstacles in various environments. The head of the robot can pan and tilt freely. The range of the panning angle is from $-90^\circ$ to $90^\circ$, and the range of the tilting angle is from $-45^\circ$ to $45^\circ$ (see Fig. 4).

A candidate image is shown in Fig. 9 (the image in a box). The candidate images are obtained based on the criterion described in Section III-C. In order to test the performance of the tracking algorithm based on IVPML, eight tracking experiments in three groups are carried out. In each experiment, tracking is tested under different dimensions of $Y$. The dimensionality under which the tracking result is most stable is used. A general description of the experiments is given in Table III. Details of the experiments are stated in the following.

1) Comparing the Performance of Tracking Based on IVPML and Tracking Based on LPP: In the first experiment (Expt1), the performances of tracking based on IVPML and tracking based on standard LPP (pairwise similarities are estimated based on Euclidean distances in $\mathbb{R}^n$) are compared. The low dimensional representations of the tracking results are shown. The low dimensional representations of the tracking inputs and the selected optimal candidates shown in Fig. 11 are marked with blue dots and red down triangles, respectively.
based on IVPML are shown in Fig. 10. The optimal candidate selection criterion is given in Section III-C. The tracking results on a dynamic robotic system are shown in Fig. 11. In order to increase the difficulty, a yellow door is added in the background where the color of the door is similar to the skin color of the tracked person. The person faces the vision system first, then turns right and walks along the right direction, and finally turns left and moves along the left direction. The new tracking algorithm based on IVPML has successfully captured the head during the whole process despite the influence of the background.

The main achievement here is that the algorithm can precisely find the head image at each frame (which includes background) using its low dimensional representations. Dynamic tracking based on LPP with the same tracking principle fails to find the correct head image, and the results are shown in Figs. 12 and 13.

2) Dynamic Tracking Based on IVPML in Complicated Environment: In the second group of experiments, the tracking algorithm based on IVPML is tested in two complicated scenes.

In Expt2-1, there are two main difficulties. One is that the color of the left door frame is black, which is the same with the hair color. The other is that there is obvious change of light condition when the object person walks from the lobby into the corridor and walks back. In the experiment, the person walks to the left direction first and then walks back. Then, he repeats the process. The tracking results based on IVPML are shown in Figs. 16 and 17. The new tracking algorithm has successfully captured the head during the whole process despite the interference from other persons.

In Expt3-1, the condition is set to be the interference from other nonobject persons. The scenes of Expt3-1a and Expt3-1b are the same, which are set in the laboratory with complicated background. In both experiments, the main difficulty is that there are two other persons walking freely behind the object person. The person passes by these two persons in the tracking process. He walks to the left direction and then walks back. Then, he repeats the process. The tracking results based on IVPML are shown in Figs. 16 and 17. The new tracking algorithm has successfully captured the head during the whole process despite the interference from other persons.

3) Dynamic Tracking Based on IVPML Under Different Conditions: In the third group of experiments, the tracking algorithm based on IVPML is tested under different conditions. The tracked object is no longer restricted to the person from whom training samples are collected. Different persons are also taken as tracked objects. Experimental results have shown that the new tracking algorithm can not only successfully capture the head under different conditions but also capture the heads of different persons.

In Expt2-2, the scene is set in a hall. The background is more complicated. There are also two main difficulties. One is the large black area in the background which has similar color with the hair. The other is the obvious change of light condition in the tracking process. The person walks to the left direction first and then walks back. The tracking results based on IVPML are shown in Fig. 15. The new tracking algorithm has successfully captured the head during the whole process despite the interference from the background and the change of light condition.

In Expt3-2, the condition is set to be the sharp change of illumination. In Expt3-2a, the scene is set in a hall. There is sharp change of natural daylight during the tracking process, which has caused great changes on the object’s head appearance. The person first walks to the left direction and then walks back. In Expt3-2b, the scene is set in the laboratory. The person
Fig. 13. (White box) Tracking results on a dynamic robotic system based on LPP. It can be seen that tracking is not stable from the second frame.

Fig. 14. (White box) Tracking results of Expt2-1 on a dynamic robotic system based on IVPML.

Fig. 15. (White box) Tracking results of Expt2-2 on a dynamic robotic system based on IVPML.

Fig. 16. (White box) Tracking results of Expt3-1a on a dynamic robotic system based on IVPML. The new tracking method is used to track a noncooperative person with interference from other persons.
Fig. 17. (White box) Tracking results of Expt3-1b on a dynamic robotic system based on IVPML. The new tracking method is used to track a noncooperative person with interference from other persons.

Fig. 18. (White box) Tracking results of Expt3-2a on a dynamic robotic system based on IVPML. The new tracking method is used to track a noncooperative person with sharp change of illumination.

Fig. 19. (White box) Tracking results of Expt3-2b on a dynamic robotic system based on IVPML. The new tracking method is used to track a noncooperative person with sharp change of illumination.

Fig. 20. (White box) Tracking results of Expt3-3 on a dynamic robotic system based on IVPML. The new tracking method is used to track a noncooperative person with arbitrary rotation.
is different from the one in Expt3-2a. While the person walks to the right direction and walks back, the light in the laboratory is suddenly turned off and then turned on for two times, which has caused sharp changes on the object’s head appearance. The tracking results based on IVPML are shown in Figs. 18 and 19. The new tracking algorithm has successfully captured the head during the whole process despite the sharp change of illumination.

In Expt3-3, the condition is set to be the object’s arbitrary rotation. The scene is set in the laboratory. A full rotation of the object person has been finished during the process. In order to increase the difficulty, the light in the laboratory is suddenly turned off in the tracking process, which has caused a sharp change on the object’s head appearance. The tracking results based on IVPML are shown in Fig. 20. The new tracking algorithm has successfully captured the head during the whole process despite the object’s arbitrary rotation and the sharp change of illumination.

Remark 4.1: The main achievement of this paper is the successful dynamic tracking of a noncooperative person who can move and rotate freely, by directly using a manifold-learning method for extracting the tracking feature. An algorithm based on a specific manifold-building method has also been successfully applied to building tracking for intelligent vehicle systems and aircraft navigation systems in [11]. Tracking multiple objects and tracking with occlusions are not in our consideration. Such issues will be our future work.

V. Conclusion

In this paper, a new manifold is built up in the training step of the process, on which the input training samples are set to be close to each other if the values of their intrinsic variables are close to each other. Based on the new manifold, a manifold-learning method called IVPML has been proposed. Using IVPML, the process of dimensionality reduction is transformed into a procedure of preserving the continuity of intrinsic variables. Furthermore, a novel tracking method with feature extraction by IVPML has been proposed and successfully applied to tracking a noncooperative person (who can move and rotate freely) with a complicated background.

From the algorithm-design point of view, different from traditional manifold-learning methods, the IVPM algorithm is directly applied to dynamic tracking with complicated backgrounds, rather than extracting intrinsic variables underlying high dimensional inputs. As far as we know, the method proposed in this paper is the first one which can be used for tracking a free-moving object from a dynamic vision system based on manifold-learning approaches.

From the application point of view, based on the new manifold, a novel mathematical feature extracted from the space learned through IVPML is used for dynamic tracking, and the dimensionality of the feature is very low. With the new feature, a novel tracking algorithm is proposed to find out dynamic objects in a dynamic environment. It has also been shown that, using standard LPP, it is difficult to track this kind of human movements with a complicated background. Despite the presented eight experiments, the new tracking method is also tested on 37 independent videos which contain different environments and objects. The results showed that the tracking method based on the new feature has successfully tracked dynamic objects in all videos.

Acknowledgment

The authors would like to thank the reviewers for their invaluable comments and suggestions, which have helped improve the presentation of this paper greatly.

References

[21] M. Isard and A. Blake, “ICONDENSATION: Unifying low-level and
high-level tracking in a stochastic framework,” in Proc. Eur. Conf. Com-
[23] M. Isard and A. Blake, “Condensation—Conditional density propagation
p. 192–204.
[25] Y. Wu and T. S. Huang, “Robust visual tracking by integrating multiple
cues based on co-inference learning,” Int. J. Comput. Vis., vol. 58, no. 1,
with multi-cue adaptive fusion,” in Proc. 4th Int. Symp. Neural Netw.:

Hong Qiao (SM’06) received the B.Eng. degree in
hydraulics and control and the M.Eng. degree in
robotics from Xiaojiaotong University, Xian, China,
the M.Phil. degree in robotics control from the
Industrial Control Center, University of Strathclyde,
Strathclyde, U.K., and the Ph.D. degree in robotics
and artificial intelligence from De Montfort Univer-
She was a University Research Fellow with De
Montfort University from 1995 to 1997. She was a
Research Assistant Professor from 1997 to 2000
and an Assistant Professor from 2000 to 2002 with the Department of Manufact-
uring Engineering and Engineering Management, City University of Hong
Kong, Kowloon, Hong Kong. Since January 2002, she has been a Lecturer
with the School of Informatics, University of Manchester, Manchester, U.K.
Currently, she is also a Professor with the Laboratory of Complex Systems and
Intelligent Science, Institute of Automation, Chinese Academy of Sciences,
Beijing, China. She first proposed the concept of “the attractive region in
strategy investigation,” which has successfully been applied by herself in robot
assembly, robot grasping, and part recognition. The work has been reported in
Advanced Manufacturing Alert (Wiley, 1999). Her current research interests in-
clude information-based strategy investigation, robotics and intelligent agents,
animation, machine learning (neural networks and support vector machines),
and pattern recognition.
Dr. Qiao is a member of the Program Committee of the IEEE International
Conference on Robotics and Automation from 2001 to 2004. She is currently
the Associate Editor of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND
CYBERNETICS—PART B and the IEEE TRANSACTIONS ON AUTOMATION
SCIENCE AND ENGINEERING.

Bo Zhang received the B.Sc. degree in mathematics
from Shandong University, Jinan, China, the M.Sc.
degree in mathematics from Xi’an Jiaotong Uni-
versity, Xi’an, China, and the Ph.D. degree in
applied mathematics from the University of Strath-
clyde, Strathclyde, U.K., in 1983, 1985, and 1992,
respectively.
From 1985 to 1988, he was a Lecturer with the
Department of Mathematics, Xi’an Jiaotong Univer-
sity. He was a Postdoctoral Research Fellow with
the Department of Mathematics, Keele University,
Keele, U.K., from January 1992 to December 1994, and with the Department
of Mathematical Sciences, Brunel University, Uxbridge, U.K., from January 1995
to October 1997. In November 1997, he joined the School of Mathematical and
Informational Sciences, Coventry University, Coventry, U.K., as a Senior Lec-
turer, where he was promoted to Reader in Applied Mathematics in September
2000 and to Professor of Applied Mathematics in September 2003. Currently,
he is a Professor with the Institute of Applied Mathematics, Academy of
Mathematics and Systems Science, Chinese Academy of Sciences, Beijing,
China. His current research interests include direct and inverse scattering
problems, computational electromagnetics, partial differential equations, and
machine learning.

Peng Zhang received the B.Sc. degree from Shan-
dong University, Jinan, China, in 2006. He is cur-
rently working toward the Ph.D. degree in applied
mathematics in the Institute of Applied Mathematics,
Academy of Mathematics and Systems Science,
Chinese Academy of Sciences, Beijing, China.
His current research interests are the theory
and application of manifold learning and nonlinear
dimensionality reduction.

Suiwu Zheng received the B.Sc. degree from Harbin
Engineering University, Harbin, China, in 2005. He is
currently working toward the Ph.D. degree in the
Institute of Automation, Chinese Academy of
Sciences, Beijing, China.
His current research interest covers manifold-
learning methods for visual tracking, dynamic image
processing, and robotic vision.