



PERGAMON

Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE @ DIRECT®

Pattern Recognition 37 (2004) 1913–1923

PATTERN
RECOGNITION

THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Object recognition based on image sequences by using inter-feature-line consistencies

Jiun-Hung Chen^a, Chu-Song Chen^{b,*}

^a*Department of Computer Science and Engineering, University of Washington, Seattle, USA*

^b*Institute of Information Science, Academia Sinica, Nankang, Taipei, Taiwan, R.O.C.*

Received 19 December 2002; accepted 5 December 2003

Abstract

An image sequence-based framework for appearance-based object recognition is proposed in this paper. Compared with the methods of using a single view for object recognition, inter-frame consistencies can be exploited in a sequence-based method, so that a better recognition performance can be achieved. We use the nearest feature line (NFL) method (IEEE Trans. Neural Networks 10 (1999) 439) to model each object. The NFL method is extended in this paper by further integrating motion-continuity information between features lines in a probabilistic framework. The associated recognition task is formulated as maximizing an a posteriori probability measure. The recognition problem is then further transformed to a shortest-path searching problem, and a dynamic-programming technique is used to solve it.

© 2004 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.

Keywords: Object recognition; Appearance-based object recognition; Nearest feature lines; Sequence-based object recognition

1. Introduction

Appearance-based methods [1–9] emphasize the use of view-based representations of objects, which are constructed from a set of views of an object in a pre-processing (or learning) stage, for object recognition or tracking. The collection of views is usually recorded in a compact way through principal component analysis (PCA) [4,5], support vector machine (SVM) [6,7] or neural networks [8,9]. In the past, Murase and Nayar [4] observed that all the training feature vectors (e.g., vectors in association with a PCA representation) of an object consist of a manifold in the feature space. They approximated the manifold by using a spline interpolation for the feature vectors of a set of sampled views. In addition, Roobaert and van Hulle [7] used SVM and Roth et al. [8] used sparse network of winnows (SNoW) for modeling the sampled views. Appearance-based techniques can also

be used for recognizing objects in a cluttered environment [5,6] and for tracking long image sequences or sequences across views [2].

Object recognition via linear combination [1,10–12] is an interesting and informative concept that receives considerable attentions in recent years. In Ref. [10], Ullman and Basri demonstrated that the variety of views depicting the same object under different transformations can often be expressed as the linear combination of a small number of views, and suggested how this linear combination property may be used in the recognition process. In Ref. [12], Vetter and Poggio proposed a method based on linear object classes for synthesizing new images of an object from a single 2D view of the object by using corresponding feature points. In addition to recognizing or synthesizing views of different poses, linear combination has also been shown very useful for visual recognition or view synthesis under different illumination conditions. For example, Belhumeur and Kriegman [11] have shown that a new image under all possible illumination conditions can be expressed as a linear combination of some basis images formed by a convex polyhedral cone in R^n if the illumination model is Lambertian.

* Corresponding author. Tel.: +886-2-27883799x1310; fax: +886-2-27824814.

E-mail address: song@iis.sinica.edu.tw (C.-S. Chen).

Recently, a linear method called the nearest feature line (NFL) was proposed for object recognition [1,13]. It uses the collection of lines passing through each pair of the feature vectors belonging to an object to model appearances of this object. Instead of using splines [4], the NFL method uses a linear structure to represent the appearance manifold, which has a close relationship with the linear-combination approaches mentioned above. In essence, infinite feature vectors of the object class can be generated from finite sample vectors with the NFL method. Note that NFL can be treated as an extension of the nearest-neighbor (NN) method, and it has been theoretically proven that the NFL method can achieve a lower probabilistic error than NN when the number of available feature points for each object class is finite and the dimension of a feature space is high [14]. An experimental evaluation of the NFL method in image classification and retrieval was given in Ref. [13], which shows that it can make efficient use of knowledge about multiple prototypes of a class to represent that class.

In this paper, a framework for sequence-based object recognition is proposed by employing the concept of feature lines (FLs). In particular, by further considering inter-feature-line consistencies, our method can use several images of an object as the training input for building an image-sequence-based recognition system. More specifically, the database built in our work contains information about possible moves between views in the database. Therefore, it contains much more information than an unordered set of views, and our method would only be applicable to problem domains where this information is available. The main idea of our method is that it tries and finds objects that not only match the individual images, but also makes sure that the sequence of views in the query could match a similar sequence of views in the database. In other words, the input database does not consist of isolated example images, but rather that these images are related to each other via motion consistency. Hence, our method would work much better as more images are added, which means that the performance improvement over adding more images would be relatively better for this method than for other methods. In our framework, a recognition task is formulated as a problem of maximizing an a posteriori probability measure. This problem is further reduced to a most-probable-path searching problem in a specially designed graph, which can be effectively solved with dynamic programming.

This paper presents a general framework for sequence-based object recognition, which can be used for real-world applications such as face recognition. For example, by incorporating our recognition method with existing face detection and tracking algorithms [15,16], this framework can be used to achieve sequence-based face recognition for person identity verification. The remainder of this paper is organized as follows. In Section 2, we present a probabilistic formulation for sequence-based object recognition by employing inter-feature-line consistencies. Section 3 gives the main algorithm of this paper. Some experimental results are

shown and discussed in Section 4. Finally, we make conclusions in Section 5.

2. Object recognition by using inter-feature-line consistency

In the following, we will first review the NFL method in Section 2.1. Then, we will characterize inter-feature-line consistencies formulated in our work in Section 2.2.

2.1. Approximate manifold with feature Lines—an introduction

Assume that we have M objects, and let $X^c = \{\mathbf{x}_i^c | i = 1, \dots, N_c\}$ be a set of N_c training feature vectors belonging to object c , $c = 1, 2, \dots, M$. An FL $\overline{\mathbf{x}_i^c \mathbf{x}_j^c}$ ($i \neq j$) of object c is defined as a straight line passing through $\mathbf{x}_i^c$ and $\mathbf{x}_j^c$. An FL space of object c is denoted by $S^c = \{\overline{\mathbf{x}_i^c \mathbf{x}_j^c} | 1 \leq i, j \leq N_c, i \neq j\}$, where $\overline{\mathbf{x}_i^c \mathbf{x}_j^c} = \overline{\mathbf{x}_j^c \mathbf{x}_i^c}$, $1 \leq i, j \leq N_c$, and the number of FLs in S^c , denoted by K_c , is $N_c(N_c - 1)/2$. When there are M classes in the database, M such FL spaces can therefore be constructed, composed of a total number of $N_{total} = \sum_{c=1}^M K_c$ FLs. Let $\Gamma = S^1 \cup S^2 \cup \dots \cup S^M$ be the collection of all N_{total} FLs. The FL distance from a query $\mathbf{q}$ to some FL $\overline{\mathbf{x}_i \mathbf{x}_j}$ ($i \neq j$) is defined as

$$d(\mathbf{q}, \overline{\mathbf{x}_i \mathbf{x}_j}) = \|\mathbf{q} - \mathbf{p}\|, \quad (1)$$

where $\|\cdot\|$ is the 2-norm and $\mathbf{p}$ is the projection point of the query $\mathbf{q}$ onto $\overline{\mathbf{x}_i \mathbf{x}_j}$. The projection point $\mathbf{p}$ can be computed as $\mathbf{p} = \mathbf{x}_i + \mu(\mathbf{x}_j - \mathbf{x}_i)$ and the position parameter $\mu \in R$ is

$$\mu = \frac{(\mathbf{q} - \mathbf{x}_i)^T(\mathbf{x}_j - \mathbf{x}_i)}{(\mathbf{x}_j - \mathbf{x}_i)^T(\mathbf{x}_j - \mathbf{x}_i)}. \quad (2)$$

Fig. 1(a) illustrates FLs and FL distances. Note that when $0 \leq \mu \leq 1$, $\mathbf{p}$ is an interpolating point between $\mathbf{x}_i$ and $\mathbf{x}_j$.

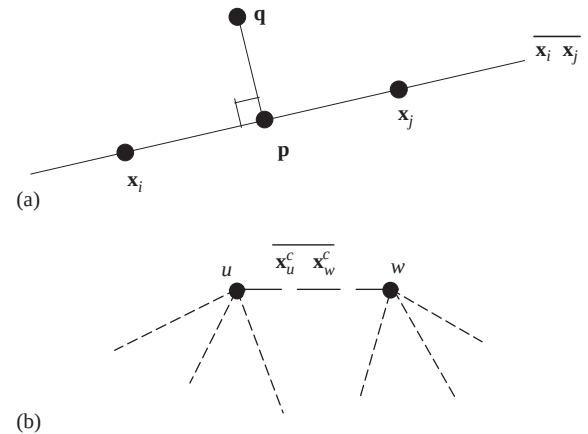


Fig. 1. (a) The concepts of an FL $\overline{\mathbf{x}_i \mathbf{x}_j}$ and the FL distance from a query $\mathbf{q}$ to $\overline{\mathbf{x}_i \mathbf{x}_j}$. (b) The concept of neighboring FLs. The feature lines drawn with dash lines are neighboring FLs of $\overline{\mathbf{x}_u^c \mathbf{x}_w^c}$.

Otherwise, $\mathbf{p}$ is a extrapolating point either on the $\mathbf{x}_i$ side (when $\mu > 1$) or the $\mathbf{x}_j$ side (when $\mu < 0$). NFL recognizes $\mathbf{q}$ as object c^* by computing the minimal FL distance between all features lines contained in Γ as shown below.

$$\begin{aligned} d(\mathbf{q}, \overline{\mathbf{x}_i^{c^*} \mathbf{x}_j^{c^*}}) &= \min_{1 \leq c \leq M} \min_{\mathbf{x}_i^c \mathbf{x}_j^c \in \mathcal{S}^c} d(\mathbf{q}, \overline{\mathbf{x}_i^c \mathbf{x}_j^c}) \\ &= \min_{\mathbf{x}_i^c \mathbf{x}_j^c \in \Gamma} d(\mathbf{q}, \overline{\mathbf{x}_i^c \mathbf{x}_j^c}). \end{aligned} \quad (3)$$

2.2. Inter-feature-line consistencies

In essence, the NFL method [1,13] only uses the distance-from-manifold information for object recognition. In our framework, we further extend the NFL method to a new one where inter-feature-line consistencies are incorporated by using the concept of neighboring FLs. Given an FL $\overline{\mathbf{x}_u^c \mathbf{x}_w^c}$, we denote $\Psi_{c,u,w} = \{\overline{\mathbf{x}_u^c \mathbf{x}_w^c}\} \cup \{\overline{\mathbf{x}_u^c \mathbf{x}_{w_1}^c} | 1 \leq w_1 \leq N_c, w_1 \neq u, w_1 \neq w\} \cup \{\overline{\mathbf{x}_{u_1}^c \mathbf{x}_w^c} | 1 \leq u_1 \leq N_c, u_1 \neq w, u_1 \neq u\}$ to be the set of its neighboring FLs. An illustration of neighboring FLs is shown in Fig. 1(b). The number of neighboring FLs of $\overline{\mathbf{x}_u^c \mathbf{x}_w^c}$ is therefore equal to $2N_c - 3$.

Let $\mathbf{A}_q = \{\mathbf{q}_0, \mathbf{q}_1, \dots, \mathbf{q}_L\}$ be a sequence of $L + 1$ views. In addition, let $\Psi = \{\mathbf{l}_i \in \Gamma | i = 0, 1, \dots, L\}$ be a set of $L + 1$ FLs where each $\mathbf{q}_i$ matches its projection point $\mathbf{p}_i$ on $\mathbf{l}_i$. We recognize $\mathbf{A}_q$ by finding Ψ^* that maximizes the following a posteriori probability:

$$\begin{aligned} \Psi^* &= \arg \max_{\Psi} P(\Psi | \mathbf{A}_q) \\ &= \arg \max_{\Psi} P(\mathbf{l}_0, \dots, \mathbf{l}_L | \mathbf{q}_0, \dots, \mathbf{q}_L) \\ &= \arg \max_{\Psi} P(\mathbf{q}_0, \dots, \mathbf{q}_L | \mathbf{l}_0, \dots, \mathbf{l}_L) P(\mathbf{l}_0, \dots, \mathbf{l}_L) \\ &= \arg \max_{\Psi} P(\mathbf{l}_0) P(\mathbf{q}_0 | \mathbf{l}_0) \\ &\quad \times \prod_{i=0, \dots, L-1} P(\mathbf{l}_{i+1} | \mathbf{l}_i) P(\mathbf{q}_{i+1} | \mathbf{l}_{i+1}), \end{aligned} \quad (4)$$

where the last equality holds by assuming that

- (i) $P(\mathbf{q}_i | \mathbf{l}_i), i = 0, \dots, L$ are independent of each other, and
- (ii) $P(\mathbf{l}_0, \dots, \mathbf{l}_L)$ can be modeled by a first-order Markov chain. That is, $P(\mathbf{l}_i | \mathbf{l}_{i-1}, \mathbf{l}_{i-2}, \dots, \mathbf{l}_0) = P(\mathbf{l}_i | \mathbf{l}_{i-1})$ for all $i = 1, \dots, L$.

the likelihoods $P(\mathbf{q}_i | \mathbf{l}_i), i = 0, \dots, L$, have to be specified. First, we can see that $P(\mathbf{q}_i | \mathbf{l}_i)$ is a probability measure in association with the similarity between $\mathbf{y}_i$ and $\mathbf{q}_i$, where $\mathbf{y}_i$ is the projection point of $\mathbf{q}_i$ onto $\mathbf{l}_i$. Hence, $P(\mathbf{q}_i | \mathbf{l}_i)$ can be set as being decreased with $\|\mathbf{q}_i - \mathbf{y}_i\|$, the distance between the observation $\mathbf{q}_i$ to its projection point $\mathbf{y}_i$. We thus refer this probability to as the *probability caused by the distance from appearance manifolds* (PDAM). Second, because the image sequence to be recognized consists of consecutive views of an object, $P(\mathbf{l}_{i+1} | \mathbf{l}_i)$ is larger when $\mathbf{l}_{i+1}$ is a more reasonable consequent of $\mathbf{l}_i$ by considering motion continuity. We refer $P(\mathbf{l}_{i+1} | \mathbf{l}_i)$ to as the *probability caused by motion continuity* (PMC). Nevertheless, note that here the concept of “motion continuity” has nothing to do with velocity or rotation, but rather only whether the relationship between 2 images in the database matches the relationship between images in the query.

3. Probabilistic framework and main algorithm

3.1. Probability distribution setting for PDAM and PMC

To evaluate Eq. (4), we make the following assumptions:

First, the PDAM is modeled as a Gaussian distribution, $P(\mathbf{q}_k | \overline{\mathbf{x}_u^c \mathbf{x}_w^c}) = \exp(-d_{c,u,w;k}^2 / 2\sigma^2) / Z$, where σ is a chosen constant, Z is a normalization constant which normalizes $P(\mathbf{q}_k | \overline{\mathbf{x}_u^c \mathbf{x}_w^c})$ to be a probability density function, and

$$d_{c,u,w;k} = d(\mathbf{q}_k, \overline{\mathbf{x}_u^c \mathbf{x}_w^c}) \quad (5)$$

for $k = 0, \dots, L, c = 1, \dots, M, 1 \leq u, w \leq N_c$ and $u \neq w$.

Second, let the PMC be defined as follows:

$$P(\mathbf{l}_{i+1} | \mathbf{l}_i) = \begin{cases} 0, & \text{if } \mathbf{l}_{i+1} \text{ is not a neighbor of } \mathbf{l}_i, \\ \frac{1}{N(\mathbf{l}_i)}, & \text{otherwise.} \end{cases} \quad (6)$$

where $N(\mathbf{l}_i) = 2N_c - 3$ is the number of neighboring FLs for $\mathbf{l}_i$.

Third, we assume equal priori probabilities for all the FLs by setting $P(\mathbf{l}) = 1/N_{total}$ for $\mathbf{l} \in \Gamma$.

Taking a natural log of Eq. (4), the following formulation can be derived:

$$\begin{aligned} \Psi^* &= \arg \max_{\Psi} \ln(P(\mathbf{l}_0) P(\mathbf{q}_0 | \mathbf{l}_0) \\ &\quad \times \prod_{i=0, \dots, L-1} P(\mathbf{l}_{i+1} | \mathbf{l}_i) P(\mathbf{q}_{i+1} | \mathbf{l}_{i+1})) \\ &= \arg \min_{\Psi} J(\Psi; \mathbf{A}_q), \quad \text{where} \end{aligned} \quad (7)$$

$$J(\Psi; \mathbf{A}_q)^1 = \begin{cases} \infty & \text{if } P(\mathbf{l}_{i+1} | \mathbf{l}_i) = 0; \\ \sum_{i=0}^L d(\mathbf{q}_i, \mathbf{l}_i)^2 / 2\sigma^2 + \sum_{i=0}^{L-1} \ln(2N_c - 3) + \ln N_{total}; & \text{otherwise.} \end{cases} \quad \text{for some } i \in \{0, \dots, L-1\}, \quad (8)$$

To evaluate Eq. (4), the transition probabilities between the feature lines $\mathbf{l}_i$ and $\mathbf{l}_{i+1}$, $P(\mathbf{l}_{i+1} | \mathbf{l}_i), i = 0, \dots, L-1$, and

¹ We assume that $\ln 0 = -\infty$.

From Eq. (7), to find Ψ^* that maximizes the a posteriori probability is equivalent to find Ψ^* that minimizes the objective function J defined in Eq. (8). Note that it is computationally intractable to use brute force for computing Ψ^* . In this work, the PDAM and PMC are encoded in a matching graph, and dynamic programming is adopted for solving this minimization problem.

3.2. Construction of matching graph

We construct a graph G containing $L+1$ levels. There are N_{total} nodes in each level, where N_{total} is the total number of FLs defined before. For the k th level, ($k = 0, \dots, L$), a number of N_{total} nodes, denoted by $n_{\overline{x_u^c x_w^c};k}$, $c \in \{1, 2, \dots, M\}$ and $u, w \in \{1, 2, \dots, N_c\}$, are constructed in association with it. In addition to these nodes, we also construct a source node n_{-1} and a sink node n_{L+2} . Then, some edges are constructed by connecting nodes in G as follows. The source node n_{-1} and sink node n_{L+2} are fully connected to nodes at level 1 and level $L+1$, respectively. For each adjacent levels k and $k+1$, $k = 0, \dots, L-1$, there is an edge $e(\overline{x_{u_1}^c x_{w_1}^c}; k; \overline{x_{u_2}^c x_{w_2}^c}; k+1)$ linking $n_{\overline{x_{u_1}^c x_{w_1}^c};k}$ to $n_{\overline{x_{u_2}^c x_{w_2}^c};k+1}$ if $\overline{x_{u_1}^c x_{w_1}^c} \in \Psi_{c,u_2,w_2}$, the set of neighbor FLs of $\overline{x_{u_2}^c x_{w_2}^c}$. Fig. 2 shows an illustration of the graph G in association with the case in which there are 2 objects and each object has 4 views.

For each node $n_{\overline{x_u^c x_w^c};k}$ (except the source and sink nodes), a node score $s_{c;u,w;k}$ is assigned to it by setting

$$s_{c;u,w;k} = d_{c;u,w;k}^2 / \sigma^2. \quad (9)$$

This node score is used to encode the log likelihood of the PDAM in Eq. (8). In addition, the scores of the source and sink nodes, s_{-1} and s_{L+2} , are both set to 0.

Then, the cost of a node is defined from the principle of dynamic programming as shown in the following. First, the cost of the source node, $cost(n_{-1})$, is set to zero. Then, the cost of each of the other nodes is defined recursively as:

$$cost(n_{\overline{x_u^c x_w^c};k}) = s_{c;u,w;k} + \min \{ cost(n_{\overline{x_{u'}^c x_{w'}^c};k'}) \mid n_{\overline{x_{u'}^c x_{w'}^c};k'} \in \Theta_{c;u,w;k}, k' < k \} \quad (10)$$

for $k = 1, \dots, L+2$, $c = 0, \dots, M$ and $0 \leq u, w \leq N_c$. Note that $\Theta_{c;u,w;k}$ is the set of nodes having edges linking to $n_{\overline{x_u^c x_w^c};k}$.

Each path starting from the source node and ending at the sink node represents a sequence of matches between a view belonging to A_q and an FL belonging to Γ . The cost of the sink node, $cost(n_{L+2})$, is then referred to as the minimal cost, and it is easy to verify that $cost(n_{L+2}) = \min_{\Psi} J(\Psi; A_q)$ defined in Eq. (8). The path associated with the minimal cost or, equivalently, the shortest path from the source to the sink nodes, is then referred to as the matching (or optimal) path in this work. The FL associated with nodes in G (except the source and sink nodes) are then treated as a sequence of matched FLs of the sequence of testing consecutive views, A_q , and the object represented by these FLs then serves as the recognition result.

In our work, to avoid recursive programming, the Dijkstra algorithm [17] is used to find the optimal path. For each node, an incoming edge with the lowest accumulated cost

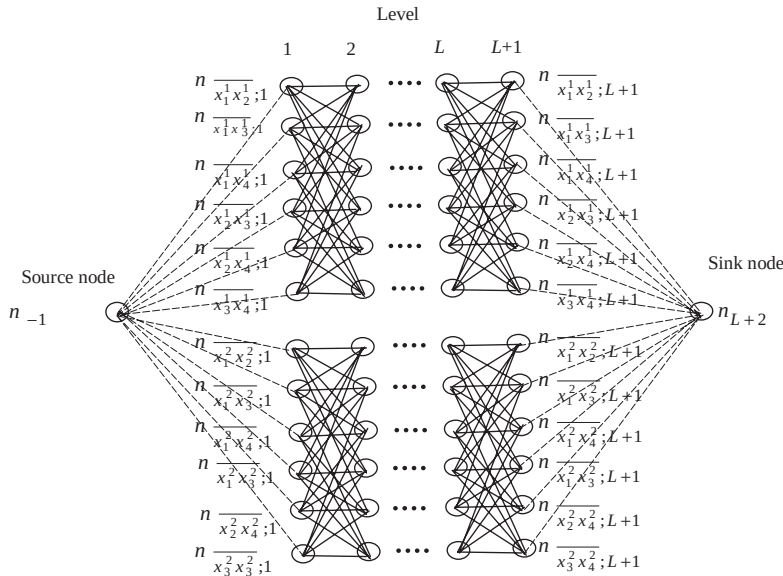


Fig. 2. An example of a graph G that is constructed for the case of two objects, where each object contains 4 views. There are therefore $6 = C_2^4$ FLs for each object.

is retained in our approach. After finding the best incoming choice for all nodes, our process backtracks, from the sink to the source nodes, to obtain the optimal path. Except for the source and sink nodes, each node passed by the optimal path then represents a match between a view belonging to A_q and an FL belonging to Γ .

4. Experimental results and discussions

4.1. Experimental results

4.1.1. Coil-100 object recognition

The Coil-100 data set [18] was widely used as an object-recognition benchmark [4,7,8]. In this data set, there are 100 objects and each object has 72 different views (images) that are taken every 5° around an axis passing through the object. Each image is a 128×128 color one with R, G, B channels. Fig. 3(a) and (b) shows these 100 objects and some sampled views of 10 of these objects, respectively.

We follow the experimental settings in [8], which used only a limited number of views per objects for training. In our experiment, 4 different views per object (0° , 90° , 180° and 270°) were used for training, as shown in Fig. 4(a), and the other 6800 (i.e., $(72 - 4) \times 100 = 6800$) images were used for testing. In other words, for each object, 6 features lines can be constructed from its 4 training views.

In our experiment, the number of consecutive views (i.e., $L + 1$) used for testing was set to be 1, 3, 5, ..., 17. A sequence of consecutive testing views of an object are shown in Fig. 4(b). Note that when the number of consecutive testing views is 1, our method degenerates to the NFL method with only a single image as its input. For each number of consecutive testing views, the average recognition error rate over 6800 tests was computed and recorded. Note that the sets of training and testing views are disjoint in our experiment, and thus the training views were dropped out when an image sequence was sampled for testing.

In the first test, the input image format was set to be colored 16×16 , which is the same as that in Ref. [7], and the 16×16 image directly serves as the feature vector. Fig. 5 shows the experimental results by using our method. From this figure, it can be seen that our method can considerably improve the recognition performance of the NFL method as the number of consecutive testing views is increased. To further show that our probabilistic framework can integrate the consecutive visual clues better, two additional methods, NN-voting and NFL-voting, which are simply extended from the NN and NFL methods to those employing image sequences by majority voting (i.e., the identity of recognition is determined by which receiving the maximal number of votes when every view in the sequence is independently recognized), respectively. As shown in

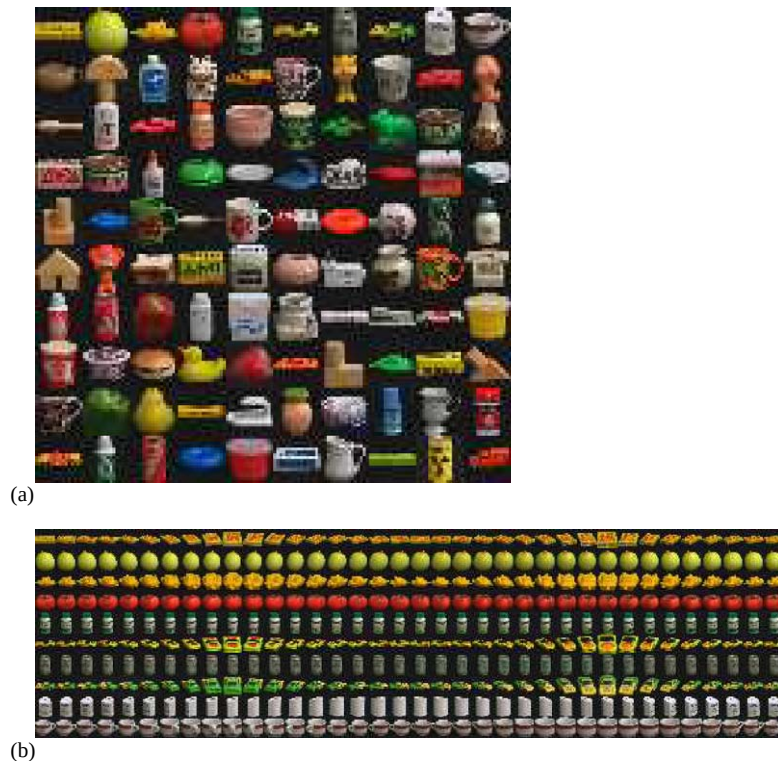


Fig. 3. Coil-100 data set. (a) The 100 objects in the Coil-100 data set. (b) Some views of 10 of the objects shown in (a) (sampled by every 10°).



Fig. 4. (a) All the training views used in the experiments. (b) The sequences of views of the first object used in our experiment (with the lengths being 1–17).

Fig. 5, our method outperforms either NN-voting or NFL-voting because appearance-similarity and motion-continuity information is appropriately exploited.

In the second test, we investigate how the recognition error rates of our method can improve as more training images were added. In this test, 32×32 gray images from the same database were used, as those adopted in Ref. [8], and the 32×32 image directly serves as the feature vector. We vary the number of training views from 4 to 8 and Fig. 6 shows these results. For example, when the numbers of training examples (FLs) increase from 4 (6) to 6 (15) and 8 (28), the lowest recognition error rates decrease from 10.38% to 3.5% and 0.67%, respectively. From these experimental results, it is shown that the performance of the method would dramatically improve as more images are added. This figure also shows that our method can greatly improve the single-view-based NFL method and are better than the NN-voting and NFL-voting methods.

4.1.2. SOIL-47 object recognition

The SOIL-47 data set [19] was recently used as a color object-recognition task and we use it in the following experiment. In the data set, there are 47 objects and each object has 21 different views (images) that are taken every 9° around an axis passing through the object. Each image is downsampled to a 24×30 color one with R, G, B channels and serves as the features being used. Fig. 7(a) and (b) shows these 47 objects and some sampled views of 2 of these objects, respectively.

In our experiment, 4 different views (that is, 6 FLs) per object were used for training as shown in Fig. 8, and the other 799 (i.e., $(21 - 4) \times 47 = 799$) images were used for testing. A sequence of consecutive testing views of an object are

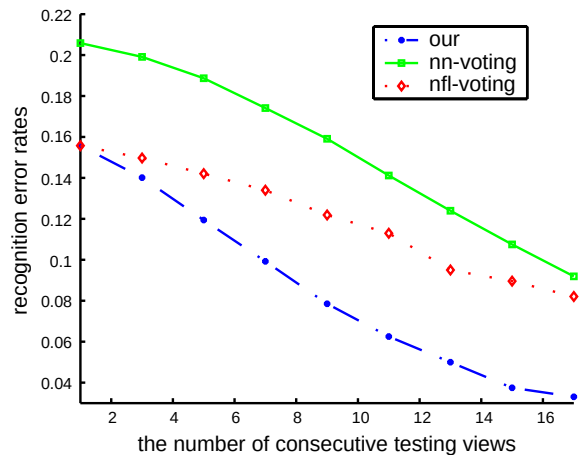


Fig. 5. Object recognition results using the same image format (i.e., color 16×16 images) as that in Ref. [7]. We compare our method (our) with nearest-neighbor voting (NN-voting) and nearest-feature-line voting (NFL-voting) methods.

constructed as described in experiments for Coil-100 data set. Fig. 9 shows the experimental results using our method. Similar to above Coil-100 experimental results, our method also outperforms either NN-voting or NFL-voting.

4.1.3. Face recognition

We perform face recognition on a face-only database.² There are 1280 images of 128 persons, where each person has 10 images with distinct poses or expressions per person.

² This database can be downloaded from <http://smart.iis.sinica.edu.tw/html/face.html>.

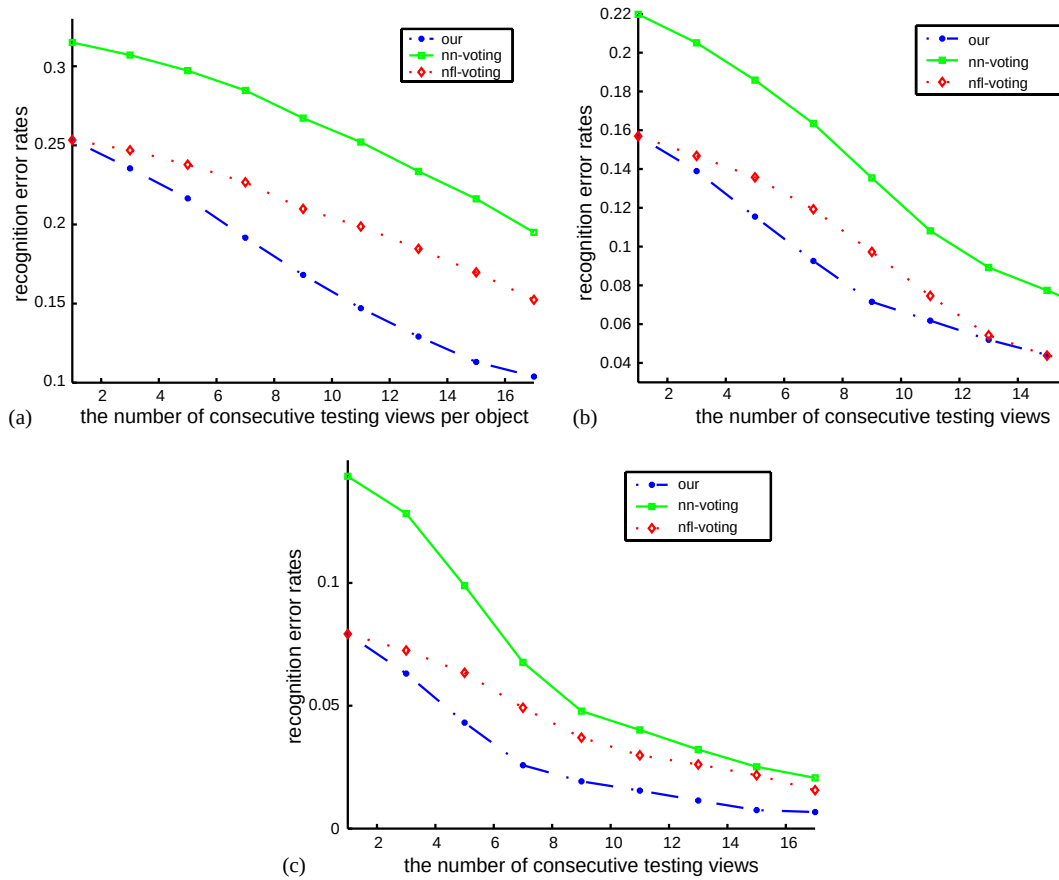


Fig. 6. Object recognition results using the same image format (i.e., gray level 32×32 images) as that in Ref. [8]: (a) 4 training views, (b) 6 training views, (c) 8 training views.

Each image size is normalized to be 32×32 , which directly serves as the features being used. Fig. 10 (a) and (b) show all 128 persons and all 10 views of the first 5 persons in this face database. In the following experiments, we use 5 different views (i.e., 10 FLs) per person in training as shown in Fig. 10(c) (left part), and other (i.e., $(10 - 5) \times 128 = 640$) images are used for testing. Hence, for each object, 10 features lines can be constructed from its 5 training views. We present results for 1, 3, and 5 consecutive testing views. Some examples of sequences of 3 views are shown in Fig. 10(c) (right part).

Table 1 shows the experimental results using our method and the comparisons between our method and NN-voting and NFL-voting. In addition, it also shows that our method give the best performance.

4.2. Discussions

To our best knowledge, no image-sequence-based methods have been used for testing these three databases in the past, and some existing results are introduced in the following for comparison.

For Coil-100 database, the following results have been reported. When the image format is colored 16×16 , and the number of training views is 4, the PCA-and-Spline-Manifold [4] and the linear SVM method [7] achieved 12.4% and 13.1% recognition error rates, respectively (the NFL method achieved 15.6% in our testing). When the image format is 32×32 gray, the linear SVM method [7] and the SNoW-with image method [8] achieve 18.4% and 21.5% error rates, respectively (the NFL method achieved 25.3% in our testing). From Figs. 5 and 6(a), it can be seen that our method can outperform the existing results shown above when only 5 and 9 consecutive images of objects are used for colored 16×16 and 32×32 gray image formats (4 training views), respectively.³ For the face-only-face database, 13% recognition error rate can be achieved in Ref. [20] when their

³ The above results were all tested based purely on image intensity information. Roth et al. [8] have further exploited edge information to achieve a better error rate, 11.7%, in their SNoW-with-image-and-edge method. Our result is better when the number of views in use is 15 without using edge information as shown in Fig. 5 (4 training views).



Fig. 7. SOIL-47 data set. (a) 47 different objects in the SOIL-47 data set. (b) Each views of 2 different objects (images in the first two rows are for a object, images in the last two rows are for the other object).

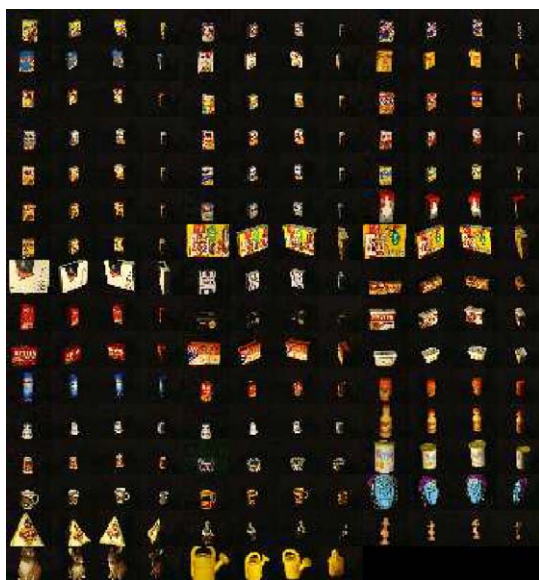


Fig. 8. All training views used in our experiments.

method is tested on a part of this database (100 persons) and 6 images per person are used as training images. From Table 1, it can be seen that our method has lower recognition error rates even when only 4 training views are used. The results reveal that the recognition performance can be considerably upgraded by appropriately exploiting useful visual clues contained in an image sequence.

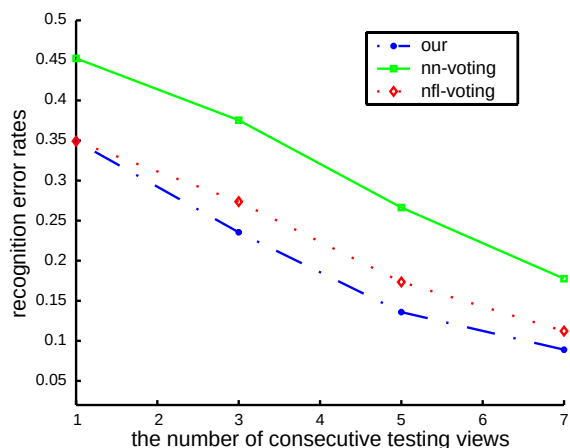


Fig. 9. Object recognition results on the SOIL-47 data set. We compare our method (our) with nearest-neighbor voting (NN-voting) and nearest-feature-line voting (NFL-voting) for image sequences.

Some further remarks are addressed below:

Remark 1 (neighborhood relationships). Although each of our experiments uses a database with a single linear sequence of views as the object rotates about a single axis, other kinds of neighborhood relationships in the database could be exploited. For example, our approach can also be used for the case that a database contains views as the object is rotated about two axes [21]. In this case, the neighborhood relationship is two-dimensional but not one-dimensional, and our approach is still applicable for this case.

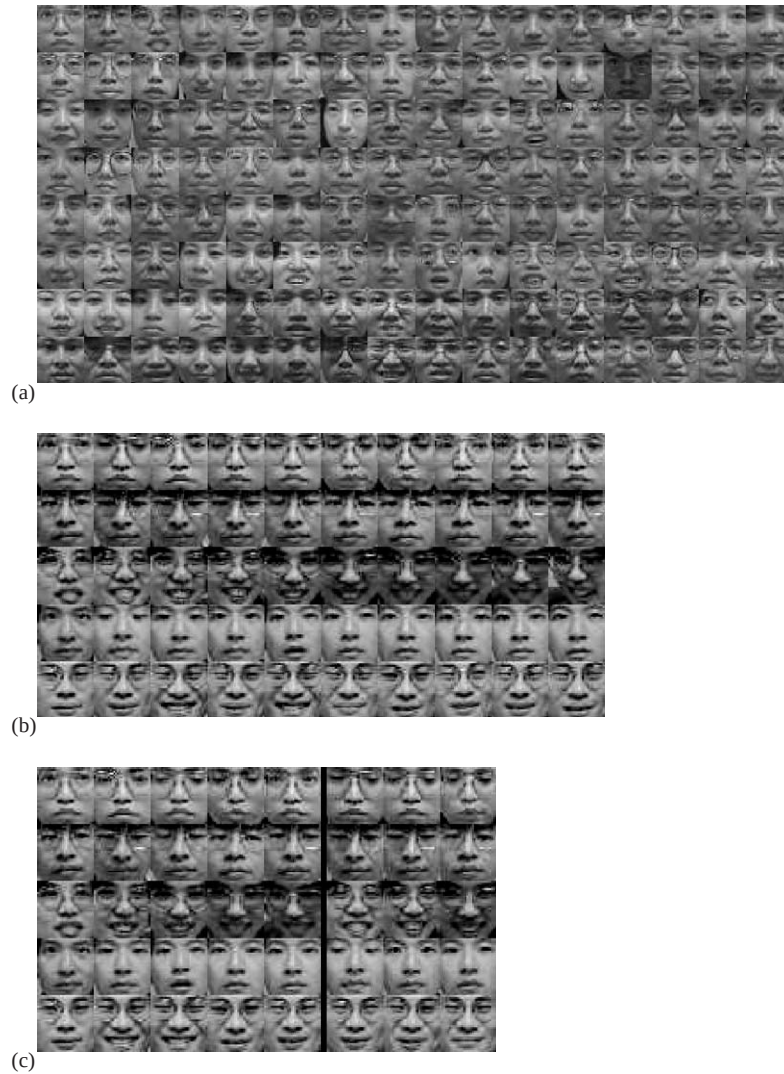


Fig. 10. A face-only database. (a) All 128 persons in this database. (b) All 10 views of the first 5 persons shown in the first row of (a). (c) Training views (left part) and some examples of sequences of 3 consecutive testing views of these 5 persons (right part), respectively. Note that the training views are formed by collecting faces in columns 1, 3, 5, 7 and 9 shown in (b).

Remark 2 (feature selection). Although raw data was directly used as feature in our experiment, this is not the only choice. Our method can also use features produced by feature-extraction or feature-generation processes such as PCA or linear discriminant analysis, and features generated in such ways would have chance to be helpful for either the computational efficiencies or the recognition accuracies.

5. Conclusions

There are several characteristics of our framework for appearance-based object recognition using a sequence of

views. First, we model appearance manifolds by NFL and emphasize inter-feature-line consistencies. Second, we take both the probability caused by the distance from manifold, PDAM, and the probability caused by motion continuity, PMC, into considerations. Third, to handle the associated recognition problem, we construct a matching graph in which PDAM and PMC are incorporated, and transform this problem into a shortest path problem that can be effectively solved by using dynamic programming. The experimental results on the Coil-100 data set, SOIL-47 data set and the face-only database show that our method achieves high recognition rates for object recognition and face recognition. Our method thus provides an effective way

Table 1

Performance comparison between our method (ours) with nearest-neighbor voting (NN-voting) and nearest-feature-line voting (NFL-voting) methods on the face-only database.

	Ours	NN-voting	NFL-voting
Number of testing views (1)	5.31	6.71	5.31
Number of testing views (3)	0.26	2.86	2.6
Number of testing views (5)	0	1.55	1.56

Recognition error rates in % on different numbers of consecutive testing views are shown. Our method gives the best performance

for appearance-based object recognition using a sequence of views.

Acknowledgements

We express our great appreciation to the anonymous reviewer for his or her highly constructive comments. The comments help very much in refining and polishing this paper.

References

- [1] S.Z. Li, J. Lu, Face recognition using the nearest feature line method, *IEEE Trans. Neural Networks* 10 (1999) 439–443.
- [2] M.J. Black, A. Jepson, Eigentracking: robust matching and tracking of articulated objects using a view-based representation, *Int. J. Comput. Vision* 26 (1998) 63–84.
- [3] B. Moghaddam, A. Pentland, Probabilistic visual learning for object representation, *IEEE Trans. on Pattern Anal. Mach. Intell.* 19 (1997) 696–710.
- [4] H. Murase, S.K. Nayar, Visual learning and recognition of 3D objects from appearance, *Int. J. Comput. Vision* 14 (1995) 5–24.
- [5] K. Ohba, K. Ikeuchi, Detectability, uniqueness, and reliability of eigen widows for stable verification of partially occluded objects, *IEEE Trans. Pattern Anal. Mach. Intell.* 19 (1997) 1043–1048.
- [6] C. Papageorgiou, T. Poggio, A pattern classification approach to dynamic object detection, in: *Proceedings of International Conference on Computer Vision*, Corfu, Greece, 1999, pp. 1223–1228.
- [7] D. Roobaert, M.M. van Hulle, View-based 3D object recognition with support vector machines, in: *Proceedings of 1999 IEEE International Workshop on Neural Networks for Signal Processing*, Madison, WI, USA, 1999, pp. 344–347.
- [8] D. Roth, M.-H. Yang, N. Ahuja, Learning to recognize 3D objects, *Neural Comput.* 14 (2002) 1071–1103.
- [9] C. Yuan, H. Niemann, An appearance based neural image processing for 3-D object recognition, in: *Proceedings of International Conference on Image Processing*, Vancouver, Canada, 2000, pp. 344–347.
- [10] S. Ullman, R. Basri, Recognition by linear combinations of models, *IEEE Trans. Pattern Anal. Mach. Intell.* 13 (1991) 992–1006.
- [11] P.N. Belhumeur, D.J. Kriegman, What is the set of images of an object under all possible illumination conditions? *Int. J. Comput. Vision* 28 (1998) 245–260.
- [12] T. Vetter, T. Poggio, Linear object classes and image synthesis from a single example image, *IEEE Trans. Pattern Anal. Mach. Intell.* 19 (1997) 696–710.
- [13] S.Z. Li, K.L. Chan, C.L. Wang, Performance evaluation of the nearest feature line method in image classification and retrieval, *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (2000) 1335–1339.
- [14] Z. Zhou, S.Z. Li, K.L. Chan, A theoretical justification of nearest feature line method, in: *Proceedings of International Conference on Pattern Recognition*, Barcelona, Spain, 2000, 759–762.
- [15] Y.-S. Chen, et al., Video-based eye tracking for autostereoscopic displays, *Opt. Eng.* 40 (2001) 2726–2734.
- [16] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: *IEEE Conference on Computer Vision and Pattern Recognition*, Kauai, HI, USA, 2001, pp. 511–518.
- [17] T.H. Cormen, C.E. Leiserson, R.L. Rivest, *Introduction to Algorithms*, MIT Press, Cambridge, MA, 1990.
- [18] S.A. Nene, S.K. Nayar, H. Murase, Columbia object image library (COIL-100), Technical Report CUCS-006-96, Columbia University, 1996.
- [19] D. Koubaroulis, J. Matas, J. Kittler, Evaluating colour-based object recognition algorithms using the SOIL-47 database, in: *Proceeding of the Fifth Asian Conference on Computer Vision*, Melbourne, Australia, 2002, pp. 840–845.
- [20] L.-F. Chen, H.-Y.M. Liao, M.-T. Ko, J.-C. Lin, G.-J. Yu, A new LDA-based face recognition system which can solve the small sample size problem, *Pattern Recogn.* 33 (2000) 1713–1726.
- [21] Y.-P. Hung, C.-S. Chen, Y.-P. Tsai, S.-W. Lin, Augmenting panoramas with object movies by generating novel views with disparity-based view morphing, *J. Visual. Comput. Animat.* 13 (2002) 237–247.

About the Author—JIUN-HUNG CHEN Received the B.S. and the M.S. degrees in computer science and information engineering from National Taiwan University, Taipei, Taiwan, in 1997 and 1999, respectively. He was a research assistant in the Institute of Information Science, Academia Sinica, Taipei, Taiwan until October 2003. Currently, he is a graduate student at the Department of Computer Science and Engineering, University of Washington at Seattle. His research interests include pattern recognition, image processing and computer vision.

About the Author—CHU-SONG CHEN Received a B.S. in Control Engineering from National Chiao-Tung University, Hsing-Chu, Taiwan, in 1989. He received an M.S. in 1991 and a Ph.D. in 1996, respectively, both from the Department of Computer Science and Information Engineering, National Taiwan University. He is currently an associate research fellow of the Institute of Information Science, Academia

Sinica, where he has been a postdoctoral fellow from 1997 to 1999, and has been promoted to the tenure track since 1999. Dr. Chen has received both the outstanding paper award of the Image Processing and Pattern Recognition (IPPR) Society and the best paper award of the Image Processing and Application Association (IPAA) of Taiwan, ROC in 1997. He has received the outstanding paper award in the field of computer applications in ICS'2000 held in Chiayi, Taiwan. His research interests include pattern recognition, computer vision, signal/image processing, and computer graphics.