Hidden Markov Model-Based Weighted Likelihood Discriminant for 2-D Shape Classification

Ninad Thakoor, *Student Member, IEEE*, Jean Gao, *Member, IEEE*, and Sungyong Jung, *Member, IEEE*

***Abstract*—The goal of this paper is to present a weighted likelihood discriminant for minimum error shape classification. Different from traditional maximum likelihood (ML) methods, in which classification is based on probabilities from independent individual class models as is the case for general hidden Markov model (HMM) methods, proposed method utilizes information from all classes to minimize classification error. The proposed approach uses a HMM for shape curvature as its 2-D shape descriptor. We introduce a weighted likelihood discriminant function and present a minimum classification error strategy based on generalized probabilistic descent method. We show comparative results obtained with our approach and classic ML classification with various HMM topologies alongside Fourier descriptor and Zernike moments-based support vector machine classification for a variety of shapes.**

***Index Terms*—Hidden Markov models (HMMs), image shape analysis, pattern classification.**

I. Introduction

OBJECT recognition is a classic problem in image processing and computer vision. Among others, object recognition based on shape is widely used. Given the volume of literature and the variety of the approaches developed for shape recognition, the problem can be classified into two categories:

1) cluster shapes that are similar to each other;
2) classify the given shape into one of the predefined classes.

Category 1 and Category 2 will be referred to as *shape clustering* and *shape classification* problems, respectively. The fundamental difference between the two is that the classification problem has a set of predefined classes while the clustering problem does not. In this paper, we will be concentrating on the shape classification where shapes are similar. This is an application of wide interest for surveillance, e.g., separating vehicle types (civilian surveillance application), classifying airplane or tank views (military surveillance applications), etc.

The first step towards the design of a shape classifier is feature extraction. Shape can be represented either by its contour or by its region [1], [2]. Curvature, chain codes, Fourier descriptors, etc., are contour-based descriptors while medial axis transform, Zernike moments, etc., are region-based features. Contour-based descriptors are widely used as they preserve the local information that is important in classification of complex shapes. Additionally, contours are the most natural way of representing and recognizing shapes by human beings. We choose curvature, which is the simplest and most natural of the contour-based shape descriptors. However, our approach can be modified to incorporate any other descriptor that preserves the local information of the contour. Examples of such descriptor include tangent space representation [3] and beam angle statistics (BAS) [4].

Feature extraction is followed by shape matching. In recent years, dynamic programming (DP)-based shape matching is being increasingly applied [5]–[8]. DP approaches are able to match the shapes part-by-part rather than point-by-point and are robust to deformation and occlusion. Hidden Markov models (HMMs) are also being explored as one of the possible shape modeling and classification frameworks [9]–[13]. HMM has been used successfully for analyzing complex signals such as speech [14], gesture [15], handwriting [16], etc. Apart from having all the properties of DP-based matching, HMM provides a probabilistic framework for training and classification.

The authors in [12] were the first to apply HMM to shape classification. They used autoregressive model for shape representation. Results were presented for stationary as well as nonstationary HMMs with 2 to 6 states. Arica and Vural [9] applied a circular HMM topology with eight states to model the shape. The shape was expressed in terms of 8-directional Freeman's code. This model topology is insensitive to the starting point and the sequence length. Their work presented results for content-based image retrieval (CBIR) rather than shape classification.

Cai and Liu [11] applied a Fourier descriptor-based HMM topology to classify the shapes. They modified HMM parameter re-estimation procedure to deal with the proposed HMM structure. Recently Bicego and Murino [10] proposed a curvature descriptor-based HMM. Curvatures are treated as mixtures of Gaussian and consequently an ergodic HMM is developed. The approach also applied Bayesian inference criterion (BIC) to select the optimum number of HMM states. Their work provides comprehensive results for classification with deformation, noise, and occlusion. These approaches presented classification results for very dissimilar shapes. However, in practical situations shapes to be classified are generally very similar. To handle such situation modification to existing approaches is mandatory.

Apart from the obvious advantages of HMM, such as robustness and time warping capability, it provides two levels of descriptions. Hidden state sequence can be considered as a simple

Manuscript received April 3, 2007; revised August 1, 2007. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Manuel Samuelides.

N. Thakoor and S. Jung are with the Electrical Engineering Department, University of Texas at Arlington, Arlington, TX 76010 USA (e-mail: ninad.thakoor@uta.edu; jung@uta.edu).

J. Gao is with the Computer Science and Engineering Department, University of Texas at Arlington, Arlington, TX 76019 USA (e-mail: gao@cse.uta.edu).

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.2007.908076

description and the likelihood description can be considered as a complex description. We use the state sequence description to achieve the invariant starting point detection and likelihood description for the classification. For a closed shape, choosing an invariant starting point is important so that point-by-point or segment-by-segment correspondence between two different shapes can be established. We propose an invariant starting point detection approach that minimizes difference between hidden state sequences to achieve this correspondence. HMM also provides inherent measure for classification in terms of maximum likelihood (ML) criterion, which is not the case for the other descriptions. The HMM approaches discussed above apply ML as their classification criterion. Due to good generalization property of HMM, applying ML criterion to classify similar shapes does not provide good classification. Additionally, ML criterion is evaluated using information from only one class and does not take advantage of information from the other classes. As ML criterion is designed to maximize likelihood of a single class, it does not guaranty good classification performance. Generally, shapes can be discriminated using only parts of the boundaries rather than comparing entire boundary. ML criterion does not provide such mechanism. Recently, Bicego *et al.* [17] have proposed a similarity-based classification approach for HMM that improves ML classification scheme. Observations are first transformed into a similarity space, which is done by calculating likelihood with respect to a few reference HMMs. The similarity space dimensionality is reduced by traditional techniques such as principle component analysis (PCA) or Fisher discriminant, followed by nearest neighbor classification. The method presents a generalized sequence classification approach; however, it does not take advantage of the nature of the 2-D shape classification problem. Combination of HMM and neural networks (NN) has been used in field of handwriting recognition [18], speech recognition [19]–[22], sonar signal classification [23], [24], etc., to improve upon the classification capability of HMM. However, researchers have preferred HMM combined with minimum classification error (MCE) formulation in speech recognition [25]–[27].

To overcome the shortcomings of ML classification, we propose a weighted likelihood discriminant for shape classification. The weighting scheme emulates comparison of parts of shape rather than the entire shape. The weights are estimated by applying generalized probabilistic descent (GPD) method to MCE formulation [28], [29]. Unlike ML criterion, the misclassification measure used with GPD combines information from all the classes to estimate the weights. As GPD method is designed to minimize the classification error, the proposed classifier gives good classification performance with similar shapes. To the best of our knowledge, this is the first effort to combine HMM and GPD to tackle the 2-D shape classification problem. Our preliminary work based on this idea appeared in [30]. Subsequently, a short version of current paper was published in [31].

This paper is organized as follows. Sections II and III give an overview of the HMM and GPD methods, respectively. The shape description phase of the proposed method is discussed in Section IV, while Section V formulates the weighted likelihood discriminant function. The GPD method-based training algorithm for the proposed discriminant function is described in Section VI. Experimental results are presented in Section VII, and the paper ends with the conclusions and suggestions for further research in Section VIII.

II. Hidden Markov Model

HMM is a stochastic signal model widely used in the field of speech processing. Recently, many researchers have applied ideas of HMM to shape recognition [9]–[13]. HMM can explain an observation sequence $O = O_1 O_2 \dots O_T$ in terms of an underlying state sequence $Q = q_1 q_2 \dots q_T$. In this section, we review HMM briefly. The details of HMM and its applications can be found in [14].

HMM is characterized by following parameters.

1) S, set of states. $S = \{S_1, S_2, \dots, S_N\}$, where N is number of states. State of HMM at instance t is denoted by q_t.
2) A, state transition probability distribution. $A = \{a_{ij}\}$, a_{ij} denotes the probability of changing the state from S_i to S_j

$$a_{ij} = P[q_{t+1} = S_j \,|\, q_t = S_i], \quad 1 \le i, j \le N. \tag{1}$$

3) B, observation symbol probability distribution. $B = \{b_j(o)\}$, $b_j(o)$ gives probability of observing symbol o in state S_j at instance t

$$b_j(o) = P[o \text{ at } t \,|\, q_t = S_j], \quad 1 \le j \le N. \tag{2}$$

4) π, initial state distribution. $\pi = \{\pi_i\}$, π_i gives probability of HMM being in state S_i at instance $t = 1$

$$\pi_i = P[q_1 = S_i], \quad 1 \le i \le N. \tag{3}$$

For convenience, HMM λ can be compactly denoted as

$$\lambda = (A, B, \pi). \tag{4}$$

The conventional parameter estimation procedure for HMM utilizes information from single class to train the HMM parameters. More classes can be added easily to the classifier without training the entire classifier again. Training is needed for the newly added class alone to incorporate the new class. On the other hand, HMM-based classifier cannot discriminate between similar shapes due to its ability to generalize. In practical situations, classification problems generally involve similar shape classes. A better training strategy is, thus, required to handle such situations. In the following section, we introduce the GPD method for training of a discriminant function that performs better in this condition.

III. Generalized Probabilistic Descent Method

The formulation of GPD [28], [29] is closely related to the concept of MCE learning. Consider a problem of classifying

an observation vector O into one of the M classes, $C_j, j = 1, 2, \ldots, M$, by discriminant function approach. According to probabilistic descent theorem, the classifier parameters Λ can be iteratively re-estimated to minimize a cost function. This cost function is a smooth representation of classification error, which means minimization of the cost function leads to the MCE.

MCE formulation is a three-step procedure. The first step is to select a discriminant function $g_j(O;\Lambda)$. This can be any conventional discriminant function such as a distance measure, ML or a function designed to suit the classification scheme under consideration. For the discussion in this section, we consider a discriminant that is to be maximized to identify the correct class. An observation O can be classified using this discriminant as

$$C(O) = C_i, \text{ where } i = \arg\max_j g_j(O;\Lambda). \tag{5}$$

Note that the above classification process is not differentiable, due to which the discriminant functions cannot be optimized with gradient search techniques.

A misclassification measure is introduced in the second step to embed the above decision process in a continuous differentiable functional form. One way of defining such a measure is

$$d_j(O;\Lambda) = -g_j(O;\Lambda) + \left\{ \frac{1}{M-1} \sum_{k=1,k\neq j}^{M} g_k(O;\Lambda)^\eta \right\}^{1/\eta} \tag{6}$$

where η is a positive smoothing factor. The misclassification measure has two components, one corresponding to the correct class C_j, i.e., $g_j(O;\Lambda)$ and the rest, which corresponds to incorrect classes. Thus, in (6), η controls the degree to which the maximum of the discriminant from incorrect classes dominates the expression in the bracket. Note that always

$$g_j(O;\Lambda) > \left\{ \frac{1}{M-1} \sum_{k=1,k\neq j}^{M} g_k(O;\Lambda)^\eta \right\}^{1/\eta}$$

and minimizing d_j for correct class corresponds to maximizing g_j (i.e., minimizing $-g_j$) that is required for the correct classification. Thus, for the correct classification, (6) has to be minimized. In the extreme case, when $\eta \to \infty$

$$d_j(O;\Lambda) = -g_j(O;\Lambda) + g_{k'}(O;\Lambda). \tag{7}$$

Here, k' is the maximum of the discriminant function of incorrect classes. Negative d_j indicates the correct classification and positive d_j indicates a classification error. This is equivalent to classification shown in (5).

In the third step, a cost function is defined that maps the misclassification measure between zero and one. Given in the following is one of the possible cost functions:

$$l_j(d_j) = \frac{1}{1 + e^{-\xi d_j}}, \quad \xi > 0. \tag{8}$$

To minimize the classification error, above cost function also has to be minimized. Similar to the misclassification measure, it is required that the discriminant function, and the cost function are continuous and differentiable functions of Λ. This ensures that numerical methods such as gradient search can be applied to optimize the parameters Λ. According to the probabilistic descent theorem, parameter re-estimation rule for the above formulation is given as

$$\Lambda_{n+1} = \Lambda_n - \varepsilon U \nabla l_j(O;\Lambda) \tag{9}$$

where U is a positive definite matrix and ε is a small real number. ε is called learning factor and it controls the speed and accuracy of convergence of the parameters.

With properly formulated misclassification measure, the GPD method can utilize information from all the classes for training and can be directly used to train the HMM discriminant function, i.e., the ML criterion. The other training approaches such as maximum mutual information (MMI)-based training [32], which use information from all the classes, also exist. However, the classification performance of properly designed and ML trained HMM cannot be improved significantly with MMI or GPD training of HMM [29]. Therefore, we stay with the optimally designed HMM as described in Section IV in our paper and make our contributions in designing robust discriminant functions with minimum error.

IV. Shape Description With HMM

A. HMM Topology

To achieve good classification results, ML approaches need carefully designed HMM topology with a large number of Gaussian mixtures. For the proposed approach in this paper, the description phase employs HMM topology proposed by Bicego and Murino [10]. The curvature [33] of the shape is used as the descriptor. Before formulating HMM, we briefly try to understand the underlying process of shape curvature. Any shape can be assumed to be formed by various segments, each of which has a constant curvature. Any deviation from the constant curvature can be due to the noise or due to the details of the shape. Each of these segments can be treated as a state of the HMM. Each state can be modeled as a Gaussian distribution with mean representing the constant curvature of the segment and standard deviation representing the deviation from constant curvature of the segment. For simple shapes, detecting the number of constant curvature segments might be possible. However, for the complex shapes detecting number of constant curvature segments is difficult. In addition, for a complex shape, number of these segments can be significantly high. In this case, computational complexity of HMM would increase. To determine the number of HMM states automatically in such cases, we apply same strategy as [10]. In the rest of the section, we discuss the formulation of the HMM for the shape.

The shape is first filtered with large variance Gaussian filter to reduce the effect of noise in curvature estimation. The filtered shape is normalized to a fixed perimeter to make the curvature invariant of the scale. Let the normalized shape be indicated by $D = \{D_n\}$ and $D_n = (x_n, y_n)$ for $1 \leq n \leq T$, where T is the normalized perimeter of the shape, and D_n indicates the coordinates of nth point of the shape. Finally, approximate

curvature at each point is calculated as the turn angle at that point. The turn angle θ_n at point D_n is defined as

$$\theta_n = \arctan \frac{y_n - y_{n-1}}{x_n - x_{n-1}} - \arctan \frac{y_n - y_{n+1}}{x_n - x_{n+1}}. \quad (10)$$

The turn angle θ_n is treated as the observation O_n for the HMM. Each shape class is modeled by a N-state ergodic HMM and observation symbol probability distribution, i.e., b_j of each state is modeled by a 1-D Gaussian distribution. the Gaussian mixture model (GMM) [34] for N clusters estimated from unrolled values of curvature of L samples of the shape, is used to initialize $B = \{b_j\} = \{(\mu_j, \sigma_j)\}$. Baum–Welch algorithm is then applied to estimate the parameters of the HMM $\lambda_j = (A_j, B_j, \pi_j)$.

In addition to these parameters, the number of HMM states N, is another important parameter. As N increases, computational complexity involved in the training and the classification increases. As the likelihood of the model increases with N, ML criterion cannot be used to select N. We choose the optimum N for the HMM by applying the BIC [35]. BIC penalizes the likelihood of the HMM according to its complexity. In [10], BIC is applied to GMM to select optimal N, but this gives optimal N for the GMM and not for the HMM. In our approach, BIC is applied to the HMM to ensure proper model selection. For the HMM topology discussed, BIC can be written as

$$\mathrm{BIC}_N(\lambda_j) = \log P(O|\lambda_j) - \frac{N^2 + 2N - 1}{2} \log(T). \quad (11)$$

The penalty term in (11) is derived from number of free parameter of the model and the observation sequence length. The number of states N is selected to maximize the $\mathrm{BIC}_N(\lambda_j)$.

B. Invariant Starting Point Detection

Before we propose the weighted discriminant function, it is important to detect the starting point invariantly as formulation depends on it. Note that alignment of the shapes is not required for training of HMM itself. Major axis-based rotation similar to that of He and Kundu [12] can be used to detect invariant starting point. However, in the case of some shapes, slight change in the shape results in a large change in the major axis. Shapes with a high degree of symmetry cannot be aligned properly with the method. This method cannot detect reflection of shape. To overcome this, we apply state sequence-based rotation to detect the starting point of the shape. Recall that the shape is modeled as a sequence of constant curvature segments. A comparison between these segments gives the criterion for alignment for our method. These segments are nothing but the optimum state sequence for the HMM.

Consider a HMM $\lambda = (A, B, \pi)$ trained with multiple training samples. Let $\bar{O} = \bar{O}_1 \bar{O}_2 \ldots \bar{O}_T$ be the most likely training sample

$$\bar{O} = \arg\max_O P(O|\lambda). \quad (12)$$

We call this sample the *reference shape*. Any closed shape $O = O_1 O_2 \ldots O_T$ belonging to the class is to be aligned with the reference shape $\bar{O}$. The best path sequence for $\bar{O}$ is given by $\bar{Q} = \bar{q}_1 \bar{q}_2 \ldots \bar{q}_T$

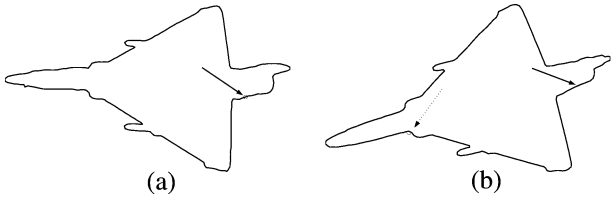

Fig. 1. Invariant starting point detection: (a) reference shape (solid arrow indicates reference starting point); (b) shape to be aligned (solid arrow indicates the detected starting point and dotted arrow indicates the original starting point).

$$\bar{Q} = \arg\max_{q_1 q_2 \ldots q_T} P(q_1 q_2 \ldots q_T|\bar{O}, \lambda). \quad (13)$$

Similarly, the best path $Q = \tilde{q}_1 \tilde{q}_2 \ldots \tilde{q}_T$ for O is

$$Q = \arg\max_{q_1 q_2 \ldots q_T} P(q_1 q_2 \ldots q_T|O, \lambda). \quad (14)$$

Mismatch between these descriptions is defined as

$$\Delta Q(k) = \sum_{t=1}^{T} \delta(t, h(t-k)) \quad (15)$$

where

$$\delta(n_1, n_2) = \begin{cases} 0, & \text{if } \bar{q}_{n_1} = \tilde{q}_{n_2}; \\ 1 & \text{if } \bar{q}_{n_1} \neq \tilde{q}_{n_2} \end{cases} \quad (16)$$

and

$$h(n) = \begin{cases} n, & \text{if } n > 0 \\ n + T, & \text{if } n \leq 0. \end{cases} \quad (17)$$

Let $O' = O'_1 O'_2 \ldots O'_T$ indicate the reflection of shape O and $\Delta Q'$ be the corresponding mismatch with respect to the reference shape. The aligned shape is given as

$$O^* = \begin{cases} O_{k^*+1} O_{k^*+2} \ldots O_T O_1 O_2 \ldots O_{k^*} \\ \qquad \text{if } \Delta Q(k^*) < \Delta Q'(k^*) \\ O'_{k^*+1} O'_{k^*+2} \ldots O'_T O'_1 O'_2 \ldots O'_{k^*} \\ \qquad \text{otherwise.} \end{cases} \quad (18)$$

where

$$k^* = \arg\min_{0 \leq k \leq T-1} \{\min(\Delta Q(k), \Delta Q'(k))\}. \quad (19)$$

In summary, the difference between $\bar{Q}$ and Q is minimized by circular shifting and reflecting Q to find an optimal rotation and reflection combination. Figs. 1 and 2 show results of the method when applied to one of the shape classes. Fig. 2(c) shows the match between two shapes that is the difference $(T - \Delta Q)$. Note that a similar plot can be obtained for $(T - \Delta Q')$. However, due to the symmetry of the shape it is not required here. The maximum occurs at $t = 95$, which is the detected as the starting point and is indicated by solid arrow in Fig. 1(b). In

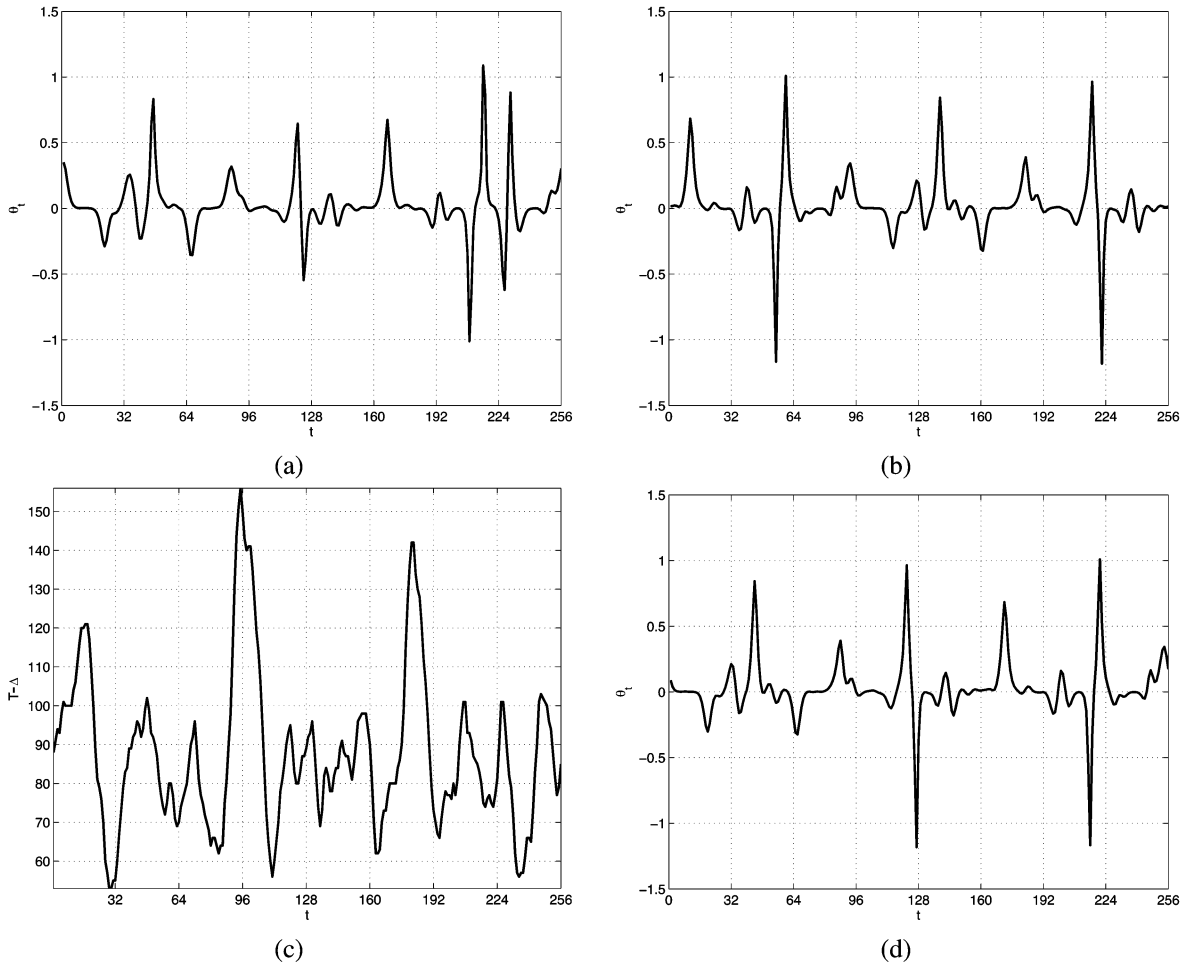

Fig. 2. Invariant starting point detection: (a) reference curvature vector; (b) observation to be aligned; (c) optimal path match; (d) aligned observation.

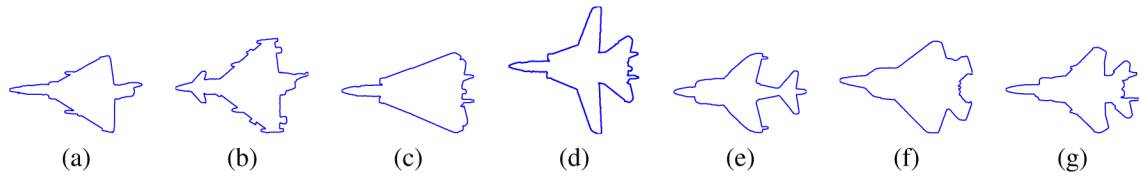

Fig. 3. Airplane shape classes: (a) Mirage; (b) Eurofighter; (c) F-14 wings closed; (d) F-14 wings opened; (e) Harrier; (f) F-22; (g) F-15.

the following sections, we assume that the starting point is already detected using this method and the obtained HMM description is used for the minimum error classification formulation. An advantage of the proposed starting point detection scheme lies in its robustness. Invariant starting point detection for shapes requires $O(MN^2T)$ calculations, where M is the number of classes, N is the number of HMM states, and T is the perimeter of the shape. Thus, the proposed method is computationally more expensive than simple rotation scheme based on major axis calculation that requires only $O(T)$ operations. In a scenario where shapes are already aligned, this step can be skipped completely. When shapes can be aligned correctly with simple rotation scheme, that rotation scheme should be preferred over HMM-based scheme.

V. Discriminant Function Formulation

In this section, we formulate a minimum error classifier with a weighted likelihood discriminant function. The discriminant function is derived from the intuitive idea that the similar shapes can be discriminated by comparing the parts of their boundaries. Meaning, some parts of the shape contour play important role in classification than the others. We signify the importance of part of shape in classification by assigning weights to it. The weights introduced in the discriminant function will be trained with GPD method.

Consider observation sequence to be classified, $O = O_1O_2 \ldots O_T$. We model the sequence with HMM $\lambda = (A, B, \pi)$. One of the possible state sequences for the given

observation sequence is given by $Q = q_1 q_2 \dots q_T$. The probability of observation sequence O is given by

$$P(O \mid \lambda) = \sum_{\forall Q} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \dots a_{q_{T-1} q_T} b_{q_T}(O_T). \quad (20)$$

Equation (20) is nothing but the ML criterion that is widely applied in classification problems. However, our goal is to weight the observations individually and the individual observation probabilities cannot be extracted from (20). To achieve this, we express the ML criterion in terms of a forward variable $\alpha_t(i)$. Forward variable is defined as

$$\alpha_t(i) = P(O_1 O_2 \dots O_t, q_t = S_i \mid \lambda). \quad (21)$$

Values of the forward variable can be calculated with the forward procedure [14]. Probability of partial observation sequence $O_1 O_2 \dots O_t$ given the model λ can be written as

$$P(O_1 O_2 \dots O_t \mid \lambda) = \sum_{i=1}^{N} \alpha_t(i). \quad (22)$$

For $t = 1$, (22) reduces to

$$P(O_1 \mid \lambda) = \sum_{i=1}^{N} \alpha_1(i). \quad (23)$$

Similarly for $t = 2$

$$P(O_1 O_2 \mid \lambda) = \sum_{i=1}^{N} \alpha_2(i). \quad (24)$$

As observations O_1 and O_2 are independent, from (23) and (24)

$$P(O_2 \mid \lambda) = \frac{\sum_{i=1}^{N} \alpha_2(i)}{\sum_{i=1}^{N} \alpha_1(i)}. \quad (25)$$

Repeating the above procedure, it can be shown that for $t > 1$

$$P(O_t \mid \lambda) = \frac{\sum_{i=1}^{N} \alpha_t(i)}{\sum_{i=1}^{N} \alpha_{t-1}(i)}. \quad (26)$$

Now, the logarithm of the ML criterion can be expressed in terms of observation probabilities as

$$\log P(O \mid \lambda) = \sum_{t=1}^{T} \log P(O_t \mid \lambda). \quad (27)$$

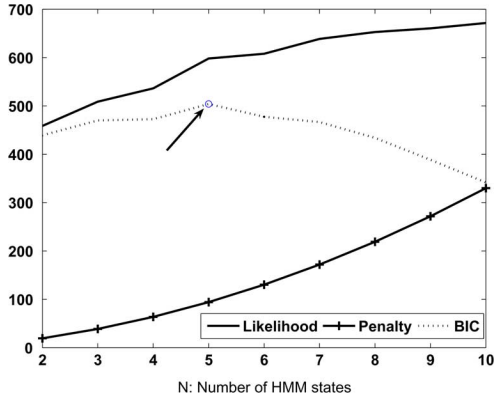

Fig. 4. Selection of HMM states for F14 (with its wings closed) based on BIC (Arrow indicates optimal value of BIC).

This function gives equal importance to every point of the shape in the classifications. Hence, we introduce a new discriminant function that weights the curvature likelihood of shape points according to their importance in classification.

The new discriminant function g_k is given by

$$g_k = \sum_{t=1}^{T} w_k(t) \cdot \log P(O_t \mid \lambda_k) \quad (28)$$

where w_k is weighting function for class C_k. w_k provides additional discrimination among the classes. These weights will be tuned by applying the GPD method to minimize the classification error. The weighting function at individual observation can be estimated by applying GPD to the current formulation. However, due to the large number of parameters (equal to T), the convergence of GPD will be slower and will need a large number of observation sequences for training. As mentioned previously in this section, to discriminate between similar shapes, comparison between parts of their contour is sufficient. As a result, shape can be weighted segment-by-segment instead of being weighted point-by-point. Following this idea, weighting functions are chosen to be windows that can adapt their position, spread and height. Although any smooth window function can be selected, our approach uses weighting function given in (29), which is a sum of S Gaussian shaped windows

$$w_k(t) = \sum_{i=1}^{S} h_{i,k} \cdot e^{-\frac{(t-\mu_{i,k})^2}{s_{i,k}^2}}. \quad (29)$$

Parameter $h_{i,k}$ governs the height, $\mu_{i,k}$ controls the position, while $s_{i,k}$ determines spread of ith window of kth class. In this case, we have only $3S$ parameters to estimate. The discriminant function can now be written as

$$g_k = \sum_{t=1}^{T} \sum_{i=1}^{S} h_{i,k} \cdot e^{-\frac{(t-\mu_{i,k})^2}{s_{i,k}^2}} \cdot \log P(O_t \mid \lambda_k). \quad (30)$$

In the next section, GPD method is applied to the above discriminant function to formulate the training algorithm.

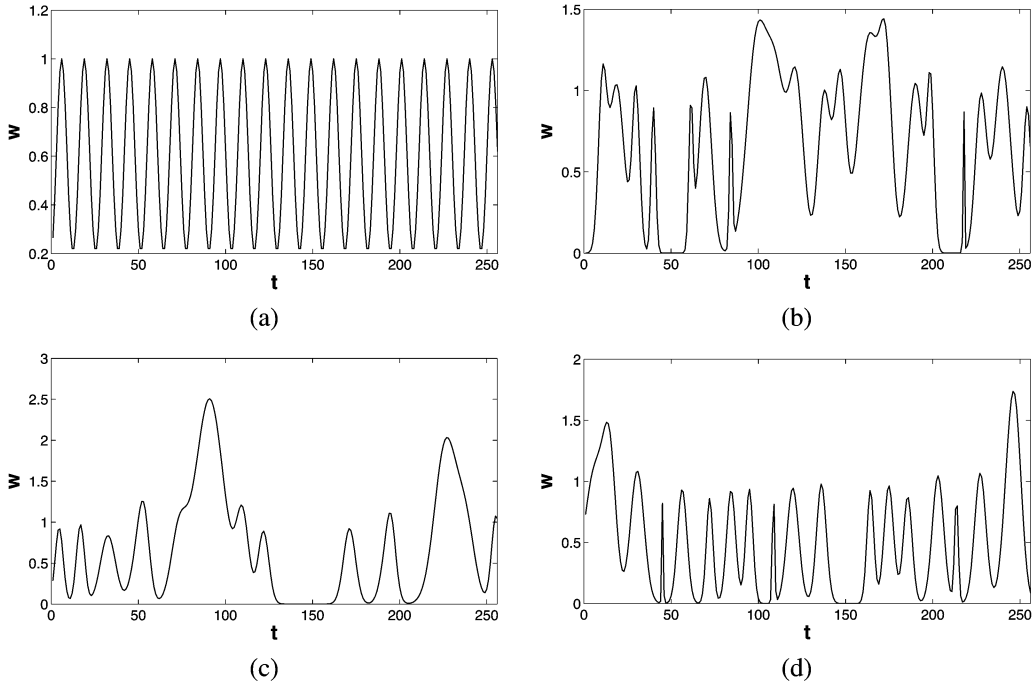

Fig. 5. Weights for various classes: (a) initial weights for all classes; (b) weights for Mirage after GPD; (c) weights for F-14 open after GPD; (d) weights for F15.

TABLE I
CONFUSION MATRIX FOR ML CLASSIFICATION

Classified as ⟶	Mirage	Eurofighter	F-14 Close	F-14 Open	Harrier	F-22	F-15
Mirage	7	0	16	1	0	6	0
Eurofighter	0	27	0	0	0	0	3
F-14 Close	0	0	29	1	0	0	0
F-14 Open	0	0	15	15	0	0	0
Harrier	0	0	1	0	28	0	1
F-22	0	0	0	0	1	29	0
F-15	0	0	0	0	0	0	30

VI. GPD ALGORITHM

To complete the formulation, we introduce a misclassification measure for observation sequence of jth class as

$$d_j = -g_j + \frac{1}{\eta}\log\left(\frac{1}{M-1}\sum_{k=1,k\neq j}^{M} e^{\eta\cdot g_k}\right) \tag{31}$$

and the corresponding cost function as

$$l_j = \frac{1}{1+e^{-\xi.d_j}}. \tag{32}$$

As discussed in Section III, the probabilistic descent re-estimation rule for parameters Λ is given as

$$\Lambda_{n+1} = \Lambda_n - \varepsilon U \nabla l_j. \tag{33}$$

For the proposed method, U is chosen to be identity matrix and the learning factor, ε is chosen to be a small number compared to the dynamic range of the parameter. The re-estimation rules in iteration n, for ith window parameters of kth class when C_j is the correct class are given by

$$h_{i,k}^{n+1} = h_{i,k}^{n} - \varepsilon_h \cdot \frac{\partial l_j}{\partial h_{i,k}} \tag{34}$$

$$\mu_{i,k}^{n+1} = \mu_{i,k}^{n} - \varepsilon_\mu \cdot \frac{\partial l_j}{\partial \mu_{i,k}} \tag{35}$$

$$s_{i,k}^{n+1} = s_{i,k}^{n} - \varepsilon_s \cdot \frac{\partial l_j}{\partial s_{i,k}} \tag{36}$$

for $1 \le i \le S, 1 \le k \le M$.

Partial derivatives appearing in (34)–(36) can be calculated by chain rule as

$$\frac{\partial l_j}{\partial h_{i,k}} = \frac{\partial l_j}{\partial d_j}\cdot\frac{\partial d_j}{\partial g_k}\cdot\frac{\partial g_k}{\partial h_{i,k}} \tag{37}$$

$$\frac{\partial l_j}{\partial \mu_{i,k}} = \frac{\partial l_j}{\partial d_j}\cdot\frac{\partial d_j}{\partial g_k}\cdot\frac{\partial g_k}{\partial \mu_{i,k}} \tag{38}$$

$$\frac{\partial l_j}{\partial s_{i,k}} = \frac{\partial l_j}{\partial d_j}\cdot\frac{\partial d_j}{\partial g_k}\cdot\frac{\partial g_k}{\partial s_{i,k}} \tag{39}$$

where

$$\frac{\partial l_j}{\partial d_j} = \frac{\xi e^{-\xi\cdot d_j}}{(1+e^{-\xi.d_j})^2}, \tag{40}$$

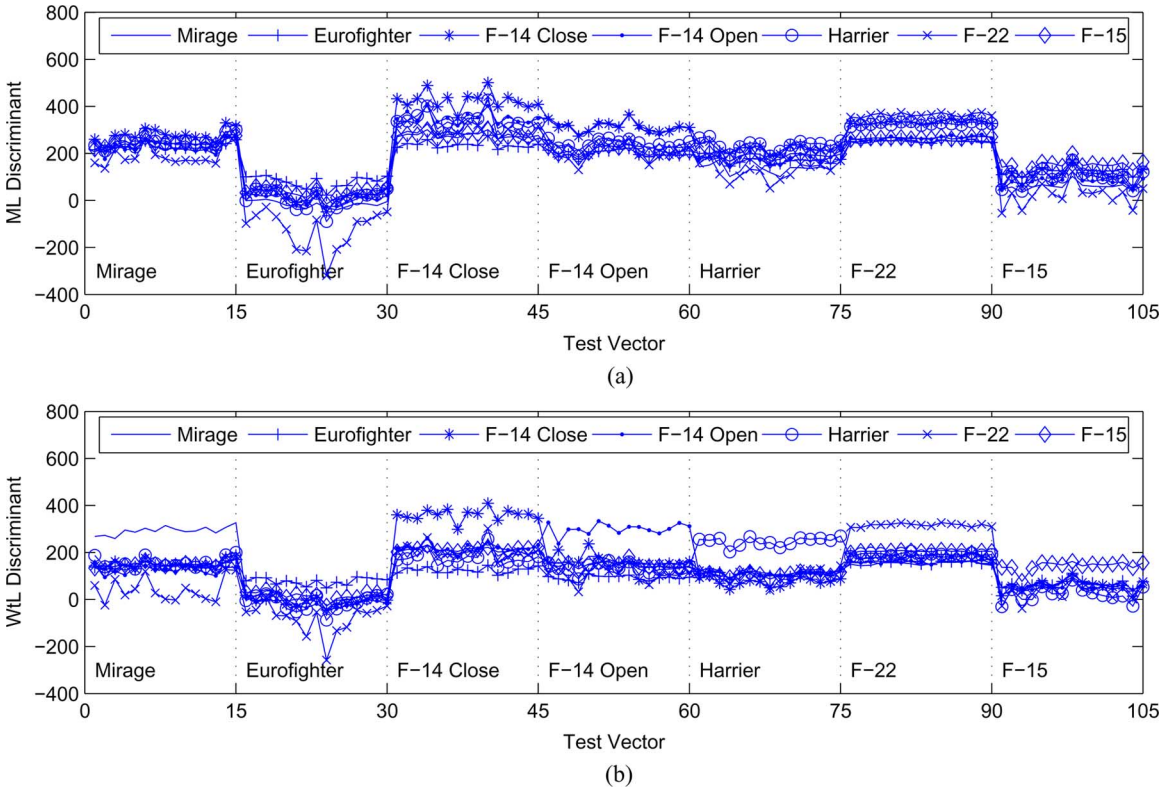

Fig. 6. (a) ML discriminant functions. (b) Weighted likelihood discriminant functions.

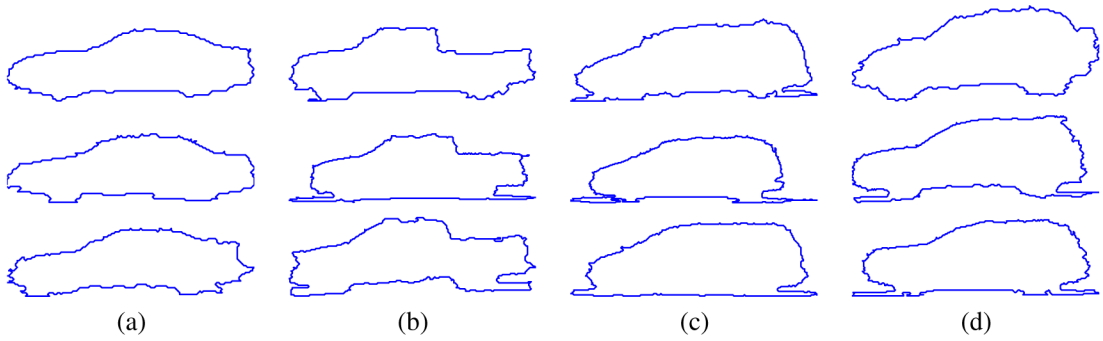

Fig. 7. Vehicle shape classes: (a) Sedan, (b) Pickup, (c) Minivan, and (d) SUV.

$$\frac{\partial d_j}{\partial g_k} = \begin{cases} -1, & j = k \text{ (for correct class)} \\ \frac{e^{\eta \cdot g_j}}{\sum_{k'=0, k' \neq k}^{M} e^{\eta \cdot g_{k'}}}, & j \neq k \text{ (for incorrect classes)} \end{cases} \tag{41}$$

$$\frac{\partial g_k}{\partial h_{i,k}} = \sum_{t=1}^{T} e^{-\frac{(t-\mu_{i,k})^2}{s_{i,k}^2}} \log P(O_t|\lambda_k) \tag{42}$$

$$\frac{\partial g_k}{\partial \mu_{i,k}} = \sum_{t=1}^{T} \frac{2h_{i,k}(t-\mu_{i,k}) e^{-\frac{(t-\mu_{i,k})^2}{s_{i,k}^2}}}{s_{i,k}^2} \log P(O_t \mid \lambda_k) \tag{43}$$

$$\frac{\partial g_k}{\partial s_{i,k}} = \sum_{t=1}^{T} \frac{2h_{i,k}(t-\mu_{i,k})^2 e^{-\frac{(t-\mu_{i,k})^2}{s_{i,k}^2}}}{s_{i,k}^3} \log P(O_t|\lambda_k). \tag{44}$$

Note that, in the above formulation, $\log P(O_t|\lambda_k)$ is treated as a constant, because the HMM parameters are not affected by the change in Λ. Appendix A gives the detailed algorithm for GPD-based training for the classifier.

VII. Experimental Results

As mentioned before, we focus on shape classification problem where shapes are similar. Most public benchmark databases are designed to test shape retrieval based on a similarity measure [36]–[38]. As we deal with the shape classification problem where shape classes are predefined and are very similar, these data sets are not best suited for our problem. The proposed method was tested with three different data sets, one of which is a subset of MPEG-7 CE Shape-1 Part-B [38] and two data sets are specially designed to test our method. The first data set is comprised of airplane shapes and the second data set included shapes of vehicles. The third data set we used was a subset of the MPEG-7 CE Shape-1 Part-B data set that was also used in [10] and [17] for experimentation.

TABLE II
CLASSIFIER RECALL WITH OVERALL ACCURACY IN PERCENTAGE FOR AIRPLANE SHAPES

Classifier→ Class↓	κ+ SVM	FD+ SVM	ZM+ SVM	HMM_E+ ML	HMM_C+ ML	HMM_{LR}+ ML	HMM_E+ WtL
Mirage	90.00	96.67	100.00	23.33	100.00	100.00	100.00
Eurofighter	73.33	100.00	50.00	90.00	100.00	100.00	100.00
F-14 Close	100.00	100.00	100.00	96.66	96.67	100.00	100.00
F-14 Open	86.67	100.00	86.67	50.00	100.00	80.00	93.33
Harrier	96.67	100.00	90.00	93.33	96.67	100.00	100.00
F-22	86.67	100.00	100.00	96.33	100.00	100.00	100.00
F-15	90.00	100.00	96.67	100.00	100.00	83.33	100.00
Overall	89.05	99.52	89.05	78.52	99.05	94.76	99.05

TABLE III
CLASSIFIER PRECISION IN PERCENTAGE FOR AIRPLANE SHAPES

Classifier→ Class↓	κ+ SVM	FD+ SVM	ZM+ SVM	HMM_E+ ML	HMM_C+ ML	HMM_{LR}+ ML	HMM_E+ WtL
Mirage	81.82	100.00	69.77	100.00	100.00	100.00	100.00
Eurofighter	84.62	100.00	100.00	100.00	100.00	100.00	100.00
F-14 Close	96.78	96.77	100.00	47.54	100.00	83.33	93.76
F-14 Open	92.86	100.00	100.00	88.25	93.75	82.76	100.00
Harrier	93.55	100.00	100.00	96.55	100.00	100.00	100.00
F-22	89.66	100.00	90.91	82.86	100.00	100.00	100.00
F-15	84.37	100.00	80.56	88.24	100.00	100.00	100.00

TABLE IV
CLASSIFIER RECALL WITH OVERALL ACCURACY IN PERCENTAGE FOR VEHICLE SHAPES

Classifier→ Class↓	κ+ SVM	FD+ SVM	ZM+ SVM	HMM_E+ ML	HMM_C+ ML	HMM_{LR}+ ML	HMM_E+ WtL
Sedan	96.67	96.67	100.00	90.00	93.33	86.67	86.67
Pickup	90.00	90.00	80.00	73.33	83.33	76.67	90.00
Minivan	46.67	73.33	66.67	60.00	83.34	83.33	83.33
SUV	16.67	70.00	70.00	26.67	33.33	36.67	76.67
Overall	62.50	82.50	79.17	62.50	73.33	70.83	84.17

TABLE V
CLASSIFIER PRECISION IN PERCENTAGE FOR VEHICLE SHAPES

Classifier→ Class↓	κ+ SVM	FD+ SVM	ZM+ SVM	HMM_E+ ML	HMM_C+ ML	HMM_{LR}+ ML	HMM_E+ WtL
Sedan	51.79	90.63	78.95	64.29	73.69	72.22	83.87
Pickup	77.14	96.43	66.67	84.62	80.64	95.83	96.43
Minivan	70.00	73.33	100.00	64.29	73.53	67.57	83.33
SUV	55.56	70.00	80.77	33.33	58.81	47.83	74.19

A. *Fighter Plane Shapes*

The fighter airplane shape database included Mirage, Eurofighter, F-14, Harrier, F-22, and F-15 [39]. Since F-14 has two possible shapes, one when its wings are closed and another when its wings are opened; the total number of shape classes are seven. Each class includes 30 shape samples. The shape database was created by taking digital pictures of die-cast replica models of these airplanes from top. Pictures were captured at 640×480 resolution, and were segmented using Spedge and Medge [40] color image segmentation algorithm. Contours of the segmented planes were used for training and testing of the classifier. Fig. 3 shows the extracted shapes for different classes.

Shapes were filtered with a Gaussian filter (standard deviation $= 10$) and the shape perimeter was normalized to 256 points. Tenfold cross validation was carried out for the normalized shapes [41]. For cross validation, the shape samples for each class were split into ten equal subsets. Out of these, nine subsets were used for training while one subset (that was not used for training) was used for testing. This process was repeated for different combinations of training and testing subsets. For the training samples of each class, HMM was built as explained in Section IV. Initially, a broad range for number of HMM states were considered and BIC was calculated. Fig. 4 shows plot of number of HMM states against likelihood and BIC for F14 with its wings closed. It can be seen that BIC declines steadily after five states as the penalty term in BIC grows rapidly compared to the gain in likelihood. Similar trends were visible in the other classes, as well. Thus, the optimum number of HMM states for each class was selected on the fly by applying BIC to models with 3 to 6 states. Sum of 20 Gaussian windows was used for formulation and training of the discriminant function. The window parameters were initialized to spread the windows uniformly over the shape [refer to Fig. 5(a)]. The training vectors were used to train the classifier with $\xi = 2$ and $\eta = 10$. Choice of smoothing parameters used, i.e., ξ and η, depends on amount of training data available. Similar to k-nearest neighbor

or Parzen window classifiers, smaller the training data available more the smoothing required [29]. Additionally, smoothing parameters also affect the optimization process. Smoother cost function surface has less number of local minima and is more tractable by gradient optimization. For all the experiments presented here, ξ and η were determined experimentally to maximize the classification accuracy.

For comparison, ML classification was carried out with optimal HMM after application of BIC. Table I gives classification results for ML classification (HMM-ML). From the confusion matrix of the ML classification, it can be seen that F-14 with its wings opened is misclassified as the F-14 with wings closed most of the times. This is because these shapes are very similar in most of their boundary. In addition, closed wing F-14 shape closely resembles the shape of a Mirage, leading to misclassification of Mirage shapes.

Fig. 5 shows the initial and estimated weights for various classes for one of the runs of the experiment. Weights for all the classes are initialized as shown in Fig. 5(a). After training, the weights modified to minimize the classification error can be seen in Fig. 5(b)–(d) for classes Mirage, F-14 with wings open and F-14, respectively.

Fig. 6 shows the ML discriminant functions and weighted likelihood discriminant functions for the test vectors. In Fig. 6, test vectors of the same class are grouped together. Labels just above the horizontal axis indicate the correct class for the test vector group and the dotted vertical lines separate the grouped correct classes. For correct classification, discriminant function of correct class should be maximum. The difference between the ML discriminant function of the correct class and the other classes is not clear in for all the classes. As a result, the classification accuracy for ML-based classification is not satisfactory. On the other hand, correct class can be easily identified for weighted likelihood discriminant, as difference between discriminant function of the correct class and the other classes is large. This large difference results into high classification accuracy for the proposed classifier.

Tables II and III compare overall classification accuracy, recall and precision with conventional descriptors such as curvature $(\kappa + \mathrm{SVM})$, Fourier descriptor $(\mathrm{FD} + \mathrm{SVM})$, Zernike moments $(\mathrm{ZM} + \mathrm{SVM})$ combined with support vector machine (SVM)-based classification, HMM with ML classification for ergodic $(\mathrm{HMM_E} + \mathrm{ML})$, circular $(\mathrm{HMM_C} + \mathrm{ML})$ and left right HMM $(\mathrm{HMM_{LR}} + \mathrm{ML})$ and proposed GPD-based weighted likelihood classification $(\mathrm{HMM_E} + \mathrm{WtL})$. Fourier descriptor were constructed for the normalized shape perimeter and Zernike moments were generated to order 30 (254 moments). Lagrangian SVM tools [42], [43] were used for SVM-based classification. HMM with ten states were used in experimentation with circular and left right topologies.

B. Vehicle Shapes

In the second experiment, we classified the vehicle shapes extracted from traffic videos using the motion information. Approach discussed in [44] was implemented and applied to outdoor videos to extract these shapes. The vehicles were classified into one of the four classes: sedan, pickup, minivan, or SUV.

Videos were captured at resolution of 320×240. As object extraction approach used does not deal with shadows, the extracted car shapes are distorted in the bottom half due to shadow. For each class, 30 samples were extracted from the video. Extracted shapes were filtered with a Gaussian filter (standard deviation $= 5$) to reduce the effect of the noise and the shape perimeter was normalized to 128 points. Similar to first experiment, tenfold cross validation was carried out for this experiment. After training HMM for all the classes, an optimum number of HMM states were selected by applying BIC to models with 3 to 6 states. Sum of 16 Gaussian windows was used for formulation and training of the discriminant function with $\xi = 10$ and $\eta = 10$.

The classification accuracy was expected to be lower than the first experiment due to following reasons.

- Shape samples show larger within-class variation, as shapes of vehicles of different makes and models vary.
- The contours extracted show higher degree of deformation due to the shadow problem in object extraction.

Comparative classification performance for individual class can be seen in Tables IV and V.

C. Subset of MPEG-7 Ce Shape-1 Part-B

MPEG-7 CE Shape-1 Part-B data set includes 1400 shape samples, 20 for each class [38]. Part of this data set was used in HMM-based shape classification experiments in [10], [17]. We use the same shape classes; however, we carried out two different subexperiments: one with all the shape samples $(\mathrm{HMM_E} + \mathrm{WtL_1})$ and the other similar to [10] and [17], where only 12 samples from each class are included $(\mathrm{HMM_E} + \mathrm{WtL_2})$. We choose following seven classes: Bone, Heart, Glass, Fountain, Key, Fork, and Hammer.

As seen in Fig. 8, the shape classes are very distinct, but the data set shows substantial within-class variations. The first subexperiment verifies that the proposed method performs satisfactorily in such a scenario. The shapes were filtered with a Gaussian filter (standard deviation $= 10$) to reduce the effect of the noise and the shape perimeter was normalized to 256 points. Similar to the preceding experiments, tenfold cross validation was carried out for this experiment. After training HMM for all the classes, an optimum number of HMM states were selected by applying BIC to models with 3 to 6 states. The sum of 20 Gaussian windows was used for formulation and training of the discriminant function with $\xi = 2$ and $\eta = 10$. Second subexperiment uses leave one out (LOO) scheme for accuracy calculation. Achieved accuracies are very close to the accuracies reported by Bicego *et al.* [17] with the same shapes being classified. All the other parameters are same as the first subexperiment. This facilitates direct comparison between the two methods that aim to improve standard HMM ML classification scheme. Overall classification accuracy with recall and precision are shown in Tables VI and VII.

VIII. Conclusion and Future Work

In this paper, we proposed a weighted likelihood discriminant function for very similar object shape classification based on the GPD theory and HMM. HMM was applied as a robust

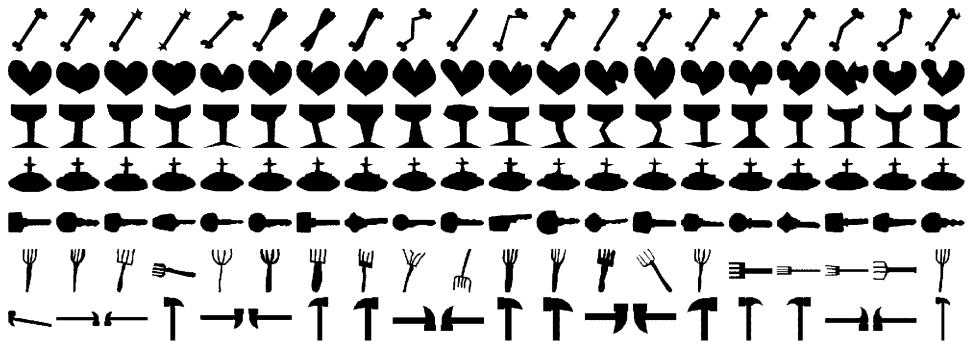

Fig. 8. Part of MPEG-7 CE Shape-1 Part-B shape classes.

TABLE VI
CLASSIFIER RECALL WITH OVERALL ACCURACY IN PERCENTAGE FOR SUBSET OF MPEG-7 CE SHAPE-1 PART-B SHAPES

Classifier→ Class↓	κ+ SVM	FD+ SVM	ZM+ SVM	HMM_E+ ML	HMM_C+ ML	HMM_{LR}+ ML	HMM_E+ WtL_1	HMM_E+ WtL_2
Bone	75.00	95.00	100.00	30.00	70.00	65.00	80.00	83.33
Heart	100.00	95.00	85.00	95.00	85.00	100.00	100.00	100.00
Glass	95.00	100.00	100.00	100.00	100.00	95.00	100.00	100.00
Fountain	100.00	95.00	100.00	100.00	100.00	100.00	100.00	100.00
Key	70.00	95.00	100.00	100.00	90.00	100.00	100.00	100.00
Fork	100.00	80.00	60.00	90.00	100.00	100.00	95.00	100.00
Hammer	45.00	100.00	100.00	45.00	100.00	60.00	100.00	100.00
Overall	83.57	94.29	92.14	80.00	92.14	88.57	96.43	97.62[a]

[a]Best classification accuracy reported by [17] for their experiment is 98.8% while worst is 92.9%.

TABLE VII
CLASSIFICATION PRECISION IN % FOR SUBSET OF MPEG-7 CE SHAPE-1 PART-B SHAPES

Classifier→ Class↓	κ+ SVM	FD+ SVM	ZM+ SVM	HMM_E+ ML	HMM_C+ ML	HMM_{LR}+ ML	HMM_E+ WtL_1	HMM_E+ WtL_2
Bone	93.75	95.00	86.96	100.00	87.50	76.47	100.00	100.00
Heart	86.96	95.00	100.00	100.00	100.00	100.00	100.00	100.00
Glass	90.48	100.00	90.91	64.52	95.24	79.17	90.91	100.00
Fountain	95.24	100.00	95.24	83.33	86.96	100.00	100.00	100.00
Key	82.35	79.17	86.96	62.50	81.82	83.33	86.96	85.71
Fork	83.33	100.00	100.00	94.74	100.00	100.00	100.00	100.00
Hammer	50.00	95.24	90.91	100.00	95.24	80.00	100.00	100.00

descriptor for the individual classes and the weighted likelihood discriminant function was used to discriminate amongst them. The weighting emulates feature selection scheme that selects salient features required to classify the shapes correctly. A training algorithm based on GPD method to estimate the optimal weights to minimize the classification error was formulated.

The performance of the proposed shape classification scheme was tested with three different shape data sets, fighter airplanes, vehicles, and subset of MPEG-7 CE Shape-1 Part-B data set. As these data sets were not generated synthetically, the results obtained for the classification are reliable in practical scenarios. In the first experiment, ML-based classification accuracy of 79% with the original ergodic model was improved to 99% by proposed method and in the second experiment it was improved from 63% to 84%. For MPEG-7 shape data set, the classification accuracy is increased to 96% from 80%. Interestingly, circular HMM has higher classification accuracies compared to the ergodic HMM that are comparable with weighted likelihood classifier in all but one case. A comparison was also carried out with Fourier descriptor and Zernike moments-based SVM classifier to show comparable or better classification performance and robustness of the scheme.

Resilience to occlusion was the prominent reason why an ergodic model was chosen instead of circular or left right model for the shape description. Although our experiments have demonstrated certain resilience to imperfectly segmented objects, occlusion remains unsolved problem in our method. However, we believe that occlusion can be handled on HMM description level and are currently working on this issue.

At present, the weighting windows are spread uniformly over the shape contour. However, these windows can be used as expert input to the classification system. One such example would be weighting only top parts of vehicle shapes as bottom parts of the vehicles are very similar, which, thus, are not important for classification. In our experiments, we avoid this to show generality of our approach. Additionally, number of weighting windows is currently adjusted heuristically and additional work is needed to select this automatically.

APPENDIX A
GPD-BASED TRAINING ALGORITHM

$M \leftarrow$ Number of classes, $S \leftarrow$ Number of windows,
$L \leftarrow$ Number of training samples per class.

for $k = 1$ to M **do**

for $i = 1$ to S **do**

Initialize window height $h_{i,k}$, window position $\mu_{i,k}$ and window spread $s_{i,k}$.

end for

Calculate weights w_k using (29).

end for

for $n = 1$ to $L \times M$ **do**

$j \leftarrow$ Correct class for current training sample.

Align the training sample with reference shape of jth class according to Section IV-B.

for $k = 1$ to M **do**

Calculate discriminant g_k using (28).

end for

Calculate misclassification measure d_j with (31).

Calculate cost function l_j with (32).

Calculate partial derivative $(\partial l_j)/(\partial d_j)$ with (40).

for $k = 1$ to M **do**

Calculate partial derivative $(\partial d_j)/(\partial g_k)$ with (41).

for $i = 1$ to S **do**

Calculate partial derivatives $(\partial g_k)/(\partial h_{i,k})$, $(\partial g_k)/(\partial \mu_{i,k})$ and $(\partial g_k)/(\partial s_{i,k})$ with (42)–(44).

Calculate partial derivatives $(\partial l_j)/(\partial h_{i,k})$, $(\partial l_j)/(\partial \mu_{i,k})$ and $(\partial l_j)/(\partial s_{i,k})$ with (37)–(39).

Calculate updated values of window parameters from (34)–(36).

end for

end for

Calculate updated weights w_k using (29) with updated window parameters.

end for

REFERENCES

[1] D. Zhang and G. Lu, "Review of shape representation and description technique," *Pattern Recognit.*, vol. 37, no. 1, pp. 1–19, Jan. 2004.

[2] L. Latecki, R. Lakamper, and T. Eckhardt, "Shape descriptors for non-rigid shapes with a single closed contour," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Jun. 2000, vol. 1, pp. 424–429.

[3] L. Latecki and R. Lakamper, "Shape similarity measure based on correspondence of visual parts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1185–1190, Oct. 2000.

[4] N. Arica and F. T. Y. Vural, "BAS: A perceptual shape descriptor based on the beam angle statistics," *Patt. Recognit. Lett.*, vol. 24, no. 9–10, pp. 1627–1639, Jun. 2003.

[5] E. Milios and E. G. M. Petrakis, "Shape retrieval based on dynamic programming," *IEEE Trans. Image Process.*, vol. 9, no. 1, pp. 141–147, Jan. 2000.

[6] E. G. M. Petrakis, A. Diplaros, and E. Milios, "Matching and retrieval of distorted and occluded shapes using dynamic programming," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 11, pp. 1501–1516, Nov. 2002.

[7] Y. Gdalyahu and D. Weinshall, "Flexible syntactic matching of curves and its application to automatic hierarchical classification of silhouettes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 12, pp. 1312–1328, Dec. 1999.

[8] T. Adamek and N. E. O'Connor, "A multiscale representation method for nonrigid shapes with a single closed contour," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 5, pp. 742–753, May 2004.

[9] N. Arica and F. Yarman-Vural, "A shape descriptor based on circular hidden Markov model," in *Proc. IEEE Int. Conf. Pattern Recognition*, 2000, vol. 1, pp. 924–927.

[10] M. Bicego and V. Murino, "Investigating hidden Markov models' capabilities in 2-D shape classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 281–286, Feb. 2004.

[11] J. Cai and Z.-Q. Liu, "Hidden Markov models with spectral features for 2-D shape recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 12, pp. 1454–1458, Dec. 2001.

[12] Y. He and A. Kundu, "2-D shape classification using hidden Markov model," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 11, pp. 1172–1184, Nov. 1991.

[13] A. Fred, J. Marques, and P. Jorge, "Hidden Markov models vs. syntactic modeling in object recognition," in *Proc. IEEE Int. Conf. Image Processing*, 1997, vol. 1, pp. 893–896.

[14] L. R. Rabiner, "A tutorial on hidden Markov models and selected application in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[15] S. Eickeler, A. Kosmala, and G. Rigoll, "Hidden Markov model based continuous online gesture recognition," in *Proc. IEEE Int. Conf. Pattern Recognition*, 1998, pp. 1206–1208.

[16] H. Jianying, M. K. Brown, and W. Turin, "HMM based online handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 10, pp. 1039–1045, Oct. 1996.

[17] M. Bicego, V. Murino, and M. Figueiredo, "Similarity-based classification of sequences using hidden Markov models," *Pattern Recognit.*, vol. 37, no. 12, pp. 2281–2291, 2004.

[18] P. Cosi, "Hybrid HMM-NN architectures for connected digit recognition," in *Proc. IEEE-INNS-ENNS Int. Joint Conf. Neural Networks*, 2000, vol. 5, pp. 85–90.

[19] R. Gemello, D. Albesano, and F. Mana, "CSELT hybrid HMM/neural networks technology for continuous speech recognition," in *Proc. IEEE-INNS-ENNS Int. Joint Conf. Neural Networks*, 2000, vol. 5, pp. 103–108.

[20] H. Bourlard and C. Wellekens, "Links between Markov models and multilayer perceptrons," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 12, pp. 1167–1178, Dec. 1990.

[21] N. Morgan and H. Bourlard, "Continuous speech recognition," *IEEE Signal Process. Mag.*, vol. 12, no. 3, pp. 25–42, May 1995.

[22] H. Misra, H. Bourlard, and V. Tyagi, "New entropy based combination rules in HMM/ANN multi-stream ASR," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Apr. 2003, vol. 2, pp. 741–744.

[23] M. Robinson, M. Azimi-Sadjadi, and J. Salazar, "Multi-aspect target discrimination using hidden Markov models and neural networks," *IEEE Trans. Neural Netw.*, vol. 16, no. 2, pp. 447–459, Mar. 2005.

[24] A. Kundu and G. C. Chen, "An integrated hybrid neural network and hidden Markov model classifier for sonar signals," *IEEE Trans. Signal Process.*, vol. 45, no. 10, pp. 2566–2570, Oct. 1997.

[25] Á. de la Torre, A. M. Peinado, A. J. Rubio, J. C. Segura, and C. Benítez, "Discriminative feature weighting for hmm-based continuous speech recognizers," *Speech Commun.*, vol. 38, no. 3–4, pp. 267–286, 2002.

[26] B.-W. Mak, Y.-C. Tam, and P. Li, "Discriminative auditory-based features for robust speech recognition," *IEEE Trans. Speech Audio Process.*, vol. 12, no. 1, pp. 27–36, Jan. 2004.

[27] R. Chengalvarayan and L. Deng, "Speech trajectory discrimination using the minimum classification error learning," *IEEE Trans. Speech Audio Process.*, vol. 6, no. 6, pp. 505–515, Nov. 1998.

[28] S. Katagiri, B.-H. Juang, and C.-H. Lee, "Pattern recognition using a family of design algorithms based upon the generalized probabilistic descent method," *Proc. IEEE*, vol. 86, no. 11, pp. 2345–2373, Nov. 1998.

[29] E. McDermott, *Handbook of Neural Networks for Speech Processing*. Norwood, MA: Artech House, 2000, ch. 5, pp. 159–216.
[30] N. Thakoor, S. Jung, and J. Gao, "Hidden Markov model based weighted likelihood discriminant for minimum error shape classification," presented at the IEEE Int. Conf. Multimedia and Expo, 2005.
[31] N. Thakoor and J. Gao, "Shape classifier based on generalized probabilistic descent method with hidden Markov descriptor," presented at the IEEE Int. Conf. Computer Vision, 2005.
[32] L. Bahl, P. Brown, P. de Souza, and R. Mercer, "Maximum mutual information estimation of hidden Markov model parameters for speech recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, 1986, vol. 11, pp. 49–52.
[33] L. F. Costa and R. M. Cesar, *Shape Analysis and Classifcation: Theory and Practice*. Boca Raton, FL: CRC, 2001.
[34] D. A. Reynolds and R. C. Rose, "Robust text-independant speaker identification using Gaussian mixture models," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 1, pp. 72–83, Jan. 1995.
[35] G. Schwarz, "Estimating the diamension of a model," *Ann. Statist.*, vol. 6, no. 6, pp. 461–464, 1978.
[36] Shape Indexing of Image Databases (SIID). [Online]. Available: http://www.lems.brown.edu/vision/researchAreas/SIID/
[37] Shape Queries Using Image Databases Database (SQUID). [Online]. Available: http://www.ee.surrey.ac.uk/Research/VSSP/imagedb/demo.html
[38] Shape Data for the MPEG-7 core experiment CE-Shape-1. [Online]. Available: http://www.cis.temple.edu/latecki/TestData/mpeg7shapeB.tar.gz
[39] D. Rendall, *Jane's Aircraft Recognition Guide*. New York: Harper Collins, 1999.
[40] J. Gao, A. Kosaka, and A. Kak, "Interactive color image segmentation editor driven by active contour model," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 1999, vol. 3, pp. 245–249.
[41] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2000.
[42] O. Mangasarian and D. R. Musicant, LSVM Software: Active Set Support Vector Machine Classification Software, 2000. [Online]. Available: www.cs.wisc.edu/musicant/lsvm/
[43] O. L. Mangasarian and D. R. Musicant, "Lagrangian support vector machine classification," Tech. Rep. 00-06, Data Mining Inst., Comput. Sci. Dept., Univ. Wisconsin, Madison, Jun. 2000. [Online]. Available: ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/00-06.ps
[44] N. Thakoor and J. Gao, "Automatic video object shape extraction and its classification with camera in motion," in *Proc. IEEE Int. Conf. Image Processing*, 2005, vol. 3, p. III-437-40.

Ninad Thakoor (S'04) received the B.E. degree in electronics and telecommunication engineering from University of Mumbai, Mumbai, India, in 2001, and the M.S. degree in electrical engineering from the University of Texas at Arlington in 2004, where he is currently pursuing the Ph.D. degree in electrical engineering.

His research interests include shape classification, stereo object segmentation, and motion segmentation.

Jean Gao (S'96–M'03) received the B.S. degree in biomedical engineering from the Shanghai Medical University, Shanghai, China, in 1990, the M.S. degree in biomedical engineering from the the Rose-Hulman Institute of Technology, Terre Haute, IN, in 1996, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, in 2002.

She is currently an Assistant Professor with the Computer Science and Engineering Department and the Director of Biocomputing and Vision Lab, University of Texas at Arlington. Her research interests include object tracking, motion estimation, stereo object segmentation, pattern recognition, medical imaging, and applications in computational biology, and clinical medical informatics.

Dr. Gao is the recipient of the prestigious CAREER Award from the National Science Foundation and the Outstanding Young Faculty Award from University of Texas at Arlington.

Sungyong Jung (M'02) received Ph.D. degree in electrical engineering from Georgia Institute of Technology, Atlanta.

Since Fall 2002, he has been a faculty member with the Electrical Engineering Department, University of Texas at Arlington. His research interests include UWB wireless IC design, analog and mixed-signal IC design, optoelectronic IC design, and modeling of high-speed circuit parasitic.