Available online at www.sciencedirect.com
SCIENCE DIRECT®
Medical Image Analysis 9 (2005) 491–502
MEDICAL IMAGE ANALYSIS
www.elsevier.com/locate/media

An EM algorithm for shape classification based on level sets ☆

Andy Tsai [a,b,*], William M. Wells [a,c], Simon K. Warfield [a,c,d], Alan S. Willsky [b]

[a] Department of Radiology at Brigham and Women's Hospital, Harvard Medical School, 75 Francis Street, Boston, MA 02115, USA
[b] Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Building 32-D, Cambridge, MA 02139, USA
[c] Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Building 32-D, Cambridge, MA 02139, USA
[d] Department of Radiology at Boston Children's Hospital, Harvard Medical School, 300 Longwood Avenue, Boston, MA 02115, USA

Available online 19 July 2005

Abstract

In this paper, we propose an expectation-maximization (EM) approach to separate a shape database into different shape classes, while simultaneously estimating the shape contours that best exemplify each of the different shape classes. We begin our formulation by employing the level set function as the shape descriptor. Next, for each shape class we assume that there exists an unknown underlying level set function whose zero level set describes the contour that best represents the shapes within that shape class. The level set function for each example shape in the database is modeled as a noisy measurement of the appropriate shape class's unknown underlying level set function. Based on this measurement model and the judicious introduction of the class labels as the hidden data, our EM formulation calculates the labels for shape classification and estimates the shape contours that best typify the different shape classes. This resulting iterative algorithm is computationally efficient, simple, and accurate. We demonstrate the utility and performance of this algorithm by applying it to two medical applications.
© 2005 Elsevier B.V. All rights reserved.

Keywords: Shape classification; Shape estimation; Level set methods; EM algorithm; Computer-aided diagnosis

1. Introduction

Shape classification can be defined as the systematic arrangement of shapes within a database, based on some similarity criteria. It has received considerable attention in recent years with important applications to problems such as computer aided diagnosis, handwriting recognition, and industrial inspection. All classification techniques require a shape descriptor that can effectively capture the important information regarding a shape, and a similarity measure that can accurately and efficiently compare different shapes. Given that, it is evident that several key issues need to be resolved while performing shape classification. In particular, a shape descriptor needs to be employed, alignment and/or correspondence issues need to be addressed, and a similarity metric needs to be designed. This paper addresses the similarity metric used in classifying shapes.

1.1. Relationship to prior work

The metric used by various classification schemes in the literature can be broadly categorized into those based on feature matching and those based on dense matching. Dense matching algorithms are computationally expensive as they try to transform or warp one shape into another based on some energy optimization scheme. For example, Del Bimbo and Pala (1997) derived a similarity

☆ This work was supported by MGH Internal Medicine Residency Program, BWH Diagnostic Radiology Residency Program, AFOSR Grant F49620-00-1-0362, Whitaker Foundation, NIH Grants R21 MH67054, R01 LM007861, P41 RR13218, NSF ERC9731748, NIH 5 P41 RR13218, and JHU EEC9731748.
* Corresponding author.
E-mail address: atsai@mit.edu (A. Tsai).

1361-8415/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.media.2005.05.001

measure between two shapes based on the amount of elastic deformation energy involved in matching the shapes. Cohen et al. (1992) developed an explicit mapping between two shape contours based on finite element analysis. Basri et al. (1995) used the sum of local deformations needed to change one shape into another as the similarity metric in comparing two shapes.

Feature matching algorithms are more popular and utilize low-dimensional feature vectors extracted from the shapes for classification. For example, Dionisio and Kim (2002) classified objects based on features computed from polygonal approximations of the object. Kawata et al. (1998) extracted surface curvatures and ridge lines of pulmonary nodules from 3D lung CT images to discriminate between malignant and benign nodules. In (Golland et al., 1999), skeletons are used to extract features which are then used within different linear classification methods (Fisher linear discriminant and linear support vectors method). Gdalyahu and Weinshall (1999) constructed syntactic representation of shapes (with primitives consisting of line segments and attributes consisting of length and orientation), and used a variant of the edit matching procedure to classify silhouettes.

We consider our algorithm as a feature matching algorithm. The individual pixels associated with each shape's level set representation are the features associated with that particular shape. One might argue that the dimensionality of this feature space is too high, and is not really a feature space as it does not capture only the salient information pertinent to a shape. However, we believe that it is the over representation or redundancy within this feature space that lends simplicity to our formulation and affords us the ability to capture very subtle differences among shapes for classification. We then incorporated this high-dimensional feature vector within an EM framework to provide us with a simple and principled approach of comparing shapes for classification.

It is worth noting that our shape classification algorithm is in part motivated by two EM-based algorithms already in the literature. In (Wells et al., 1996), the authors formulated an adaptive brain segmenter, based on the EM algorithm, that utilizes knowledge of tissue intensity properties and MRI intensity inhomogeneities to correct and segment MR images of the brain. The E-step in that formulation calculates the tissue labels and the M-step calculates the smooth MRI intensity inhomogeneities. In (Tsai et al., 2001), the authors formulated a ladar target segmentation and range profiling algorithm based on the EM algorithm. The E-step in that formulation calculates the segmentation labels for the ladar image and the M-step calculates the smooth range profiles. In both these works, the E-step is employed as a labeling process while the M-step is used to estimate a smooth field. This setup bears much resemblance to the algorithm that we will be presenting in this paper. Specifically, the E-step in our formulation calculates the shape class labels and the M-step calculates the smooth level set functions associated with the various shape classes.

Of note, our classification technique bears some resemblence to the k-means clustering algorithm (MacQueen, 1965) which groups data points, and not contours or shapes, according to a parametric model. Specifically, the M-step of our algorithm is analogous to calculating the mean value of each k-means cluster. In addition, the E-step of our algorithm is analogous to assigning each example data point to its closest cluster mean value. However, as mentioned already, k-means algorithm is not equipped to cluster shapes as it can only handle data points. Obviously, an ad hoc two-step shape classifier can be formed by employing a feature extractor first to transform the shapes to a set of representative feature points prior to the application of the k-means algorithm for shape classification.

1.2. Contributions of our work

Given a database of example shapes, the goal of our algorithm is twofold: (1) to separate the example shapes into different groups of approximately the same shapes (based on some similarity measure), and (2) to estimate the shape contour for each group that best represents or typifies the shapes contained within that group. Accomplishing these two tasks of shape classification and estimation is difficult, and is the problem which we focus on in this paper.

Importantly, in our shape classification and estimation problem, if the underlying shape contour of each shape class is known a priori, then various pattern recognition techniques in the literature can be employed to separate the shapes within the database into different groups. Similarly, if the class labels of every example in the database is known a priori, then the underlying contour for each shape class can be estimated by calculating the "average" shape contour within each shape class. Needless to say, it is difficult to calculate or determine either the representative shape contour of each class or the class labels of the example shapes in the database without knowledge of the other. However, we will show in this paper that it is possible to estimate both using the EM algorithm.[1]

The main contribution of this paper is the formulation of an EM framework that incorporates the level set method (which we employ as the implicit representation of shape) in a mathematically sound and principled fashion to yield a powerful and accurate methodology

[1] A preliminary conference paper based on this work can be found in (Tsai et al., 2004).

for shape classification. In particular, the E-step of this algorithm calculates the classification label for each example shape, and the M-step of this algorithm calculates the pixelwise implicit level set representation of the shape that best describes each shape class.

The rest of this paper is organized as follows. Section 2 describes how shapes are aligned to one another and how they are represented in our algorithm. Section 3 illustrates how we incorporated the level set methods into the EM framework for shape classification and estimation. In Section 4, we present generalizations of our algorithm and demonstrate them through various simulations. In Section 5, we present two evaluations of our algorithm by applying our technique to two medical problems and then comparing our results to those of medical experts. We conclude in Section 6 with a summary of the paper and a discussion on future research directions.

2. Shape alignment and representation

In the setting of trying to compare different shapes for classification, the alignment of the example shapes within this database $\mathscr{T}$ is paramount. In particular, the example shapes within this database need to be normalized to one another with respect to rotation, shift, and scale so that a proper comparison can be performed without interference from pose variations of the different example shapes. We chose to employ the alignment strategy described in (Tsai et al., 2003) which jointly align all the shapes within the database from a variational perspective.

Once the shapes are aligned, a shape representation scheme needs to be chosen. A number of shape representation schemes have been proposed in the literature ranging from skeletons (Bloomenthal and Lim, 1999; Golland et al., 1999) to primitives (Cesar and Costa, 1996; Gdalyahu and Weinshall, 1999) to Fourier decompositions (Staib and Duncan, 1992) to meshes (Li et al., 2001). Obviously, each type of shape descriptor has its advantages and disadvantages. In addition, a particular descriptor may be more natural or more appropriate to use in certain problems/settings or algorithmic approaches than others. Thus, it goes without saying that the choice of shape descriptor is very much dependent on the problem at hand and the algorithmic approach.

Next, we describe the level set function as our implicit representation of shape. This approach had been utilized by others, most notably by Osher and Sethian (1988) and Leventon et al. (2000). Let the shape database $\mathscr{T}$ consist of a set of L *aligned* contours $\{\mathscr{C}_1, \mathscr{C}_2, ..., \mathscr{C}_L\}$. We employ the signed distance function as the shape descriptor in representing each of these contours (Osher and Sethian, 1988). In particular, each contour is embedded as the zero level set of a signed distance function with negative distances assigned to the inside and positive distances assigned to the outside. This technique yields L level sets functions $\{Y_1, Y_2, \ldots, Y_L\}$, with each level set function consisting of $\mathscr{N}$ samples (using identical sample locations for each function). Here, if M and N denote the x- and y-dimensions of the level set, then $\mathscr{N} = M \times N$. In the case of 3D, O is used to denote the z-dimension, and $\mathscr{N} = M \times N \times O$.

Many advantages over other shape descriptors are associated with the use of level set functions for shape representation. Specifically, this approach is well known for its simplicity and numerical stability, its ability to accurately capture subtle shape characteristics, its seamless handling of topological changes in shape, and its capability to accomodate 2D and 3D data.

3. Shape classification and estimation

In this section, we describe how we employ the EM algorithm in accomplishing our two goals: separating the shape database into different groups and estimating the shape contours that best represent each group. In Section 3.1, we introduce the hidden variable, and describe the measurement and probabilistic models necessary in formulating the EM-based algorithm. We then show how they are incorporated into the EM procedure in Section 3.2. We conclude this section by describing the initialization scheme that we utilize in kick starting our EM iterative scheme.

3.1. Measurement and probabilistic models

For simplicity of derivation and clarity of presentation, we assume that there are only two shape classes within the database which we would like to group. It is important to realize, however, that it is straightforward to generalize our algorithm to classify more than two classes. This will be demonstrated in Section 4.2. By limiting ourselves to the classification of only two shape classes, we can employ the binary class label $C = \{C_1, C_2, \ldots, C_L\}$ to indicate which of the two shape classes each of the example shapes belongs to. Specifically, each $C_l \ \forall l = 1, \ldots, L$ takes on the values of 0 or 1.

In our problem formulation, because we have limited ourselves to the classification of two shape classes, we postulate that there are only two unknown level set functions $X = \{X_1, X_2\}$, one associated with each of the two shape classes. Additionally, these level set functions have the property that the zero level sets of X_1 and X_2 represent the underlying shape contours of the two shape classes A and B. Importantly, there are no restrictions placed on X being a signed distance function. Next, we view each example shape's level set function

Y_l as a noisy measurement of either X_1 or X_2. Based on this formulation, the explicit dependence of Y_l on X and C_l is given by the following measurement model:

$$Y_{l_i} = \begin{bmatrix} C_l & (1 - C_l) \end{bmatrix} \begin{bmatrix} X_{1_i} \\ X_{2_i} \end{bmatrix} + v_i \quad \forall i = 1, \ldots, \mathcal{N} \tag{1}$$

where $v \sim \mathcal{G}(0, \sigma^2 I)$ represents the measurement noise with σ as the standard deviation of the noise process.[2] This measurement model gives us the following conditional probability of Y given C and X:

$$p(Y|X,C) = \prod_{l=1}^{L} p(Y_l|X,C_l) = \prod_{l=1}^{L} \begin{bmatrix} C_l & (1 - C_l) \end{bmatrix} \begin{bmatrix} \prod_{i=1}^{\mathcal{N}} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(Y_{l_i} - X_{1_i})^2}{2\sigma^2}} \\ \prod_{i=1}^{\mathcal{N}} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(Y_{l_i} - X_{2_i})^2}{2\sigma^2}} \end{bmatrix}. \tag{2}$$

Of note, this probability model bears resemblance to the stochastic framework introduced in (Paragios et al., 2002) for the construction of prior shape models.

We assume that the class labels C and the level set representations of the shape contours X are statistically independent, and hence

$$p(C|X) = p(C). \tag{3}$$

Without any prior knowledge regarding the classifications of the various example shapes in the database, we set

$$p(C_l) = \begin{cases} 0.5 & \text{if } C_l = 0 \\ 0.5 & \text{if } C_l = 1 \end{cases} \quad \forall l = 1, \ldots, L. \tag{4}$$

3.2. The EM framework

The EM procedure, first formalized by Dempster et al. (1977), is a powerful iterative technique suited for calculating the maximum-likelihood (ML) estimates in problems where parts of the data are missing. The missing data in our EM formulation is the class labels C. That is, if the class labels for the different shapes within the database are known, then estimating the underlying shape contour which best represents each shape class would be straightforward. The observed data in our EM formulation is Y, the collection of level set representations of the example shapes. Finally, X is the quantity to be estimated in our formulation.

3.2.1. The E-step

The E-step computes the following auxilliary function Q:

$$Q(X|X^{[k]}) = \langle \log p(Y, C|X) | Y, X^{[k]} \rangle, \tag{5}$$

where $X^{[k]}$ is the estimate of X from the kth iteration, and $\langle \cdot \rangle$ represents the conditional expectation over C given Y and the current estimate $X^{[k]}$. Using Bayes' rule and our earlier simplified assumption that C and X are statistically independent, Q can be rewritten as

$$Q(X|X^{[k]}) = \langle \log p(Y|X,C) | Y, X^{[k]} \rangle + \langle \log p(C) | Y, X^{[k]} \rangle. \tag{6}$$

Since the M-step will be seen below to be a maximization of $Q(X|X^{[k]})$ over X, we can discard the second term in Eq. (6) since it does not depend on X. Expanding the remaining term in Eq. (6), we have that[3]

$$Q(X|X^{[k]}) = -\sum_{l=1}^{L} \begin{bmatrix} \langle C_l | Y_l, X^{[k]} \rangle & (1 - \langle C_l | Y_l, X^{[k]} \rangle) \end{bmatrix} \times \begin{bmatrix} \sum_{i=1}^{\mathcal{N}} (Y_{l_i} - X_{1_i})^2 \\ \sum_{i=1}^{\mathcal{N}} (Y_{l_i} - X_{2_i})^2 \end{bmatrix}. \tag{7}$$

As evident from above, the core of the E-step is the computation of $\langle C_l | Y_l, X^{[k]} \rangle$. Using the formula for expectations, Bayes's rule, and Eqs. (2)–(4), we find that

$$\langle C_l | Y_l, X^{[k]} \rangle = \frac{\prod_{i=1}^{\mathcal{N}} \exp(-(Y_{l_i} - X_{1_i}^{[k]})^2 / 2\sigma^2)}{\prod_{i=1}^{\mathcal{N}} \exp(-(Y_{l_i} - X_{1_i}^{[k]})^2 / 2\sigma^2) + \prod_{i=1}^{\mathcal{N}} \exp(-(Y_{l_i} - X_{2_i}^{[k]})^2 / 2\sigma^2)}. \tag{8}$$

This equation is equivalent to calculating the posterior shape class probabilities assuming that the underlying level set functions X_1 and X_2 are known.

3.2.2. The M-step

Estimates of X_1 and X_2 are obtained in the M-step of our formulation by maximizing the auxiliary function Q. In other words, the M-step calculates the $X^{[k+1]}$ such that

$$X^{[k+1]} = \arg\max_X Q(X|X^{[k]}). \tag{9}$$

To solve for $X^{[k+1]}$, we imposed the zero gradient condition to Eq. (7). In particular, by differentiating $Q(X|X^{[k]})$ with respect to X_{1_i} and X_{2_i} for each pixel i, and setting each resulting equation to 0, we obtain the following two expressions:

[2] The notation I represents the identity matrix and the notation $\mathcal{G}(\mu, \Lambda)$ represents a Gaussian random vector with mean μ and variance Λ.

[3] Here, in order to take the log of $p(Y|X,C)$ in Eq. (2), we have used the fact that each C_l is a binary indicator which selects the appropriate probability distribution function.

$$X_{1_i}^{[k+1]} = \frac{\sum_{l=1}^{L} \langle C_l | Y_{l_i}, X_i^{[k]} \rangle Y_{l_i}}{\sum_{l=1}^{L} \langle C_l | Y_{l_i}, X_i^{[k]} \rangle} \quad \forall i = 1, \ldots, \mathcal{N},$$
$$X_{2_i}^{[k+1]} = \frac{\sum_{l=1}^{L} \left(1 - \langle C_l | Y_{l_i}, X_i^{[k]} \rangle\right) Y_{l_i}}{\sum_{l=1}^{L} \left(1 - \langle C_l | Y_{l_i}, X_i^{[k]} \rangle\right)} \quad \forall i = 1, \ldots, \mathcal{N}. \tag{10}$$

Eq. (10) is equivalent to an ML estimator of the level set functions X_1 and X_2 when the shape labels are known. Of interest, note that both X_1 and X_2 are weighted averages of distance maps from different shape examples.[4] As a result, neither X_1 nor X_2 are signed distance functions because distance functions are not closed under linear operations. Though this gives rise to an inconsistent framework, it does not affect the quality of the shape representation based on our experimental results.

3.2.3. Initialization

Since the EM algorithm is an iterative scheme, initialization is required. The initialization of this algorithm can either be an initial guess of the shape labels (the output of the E-step) or of the level set for each shape class (the output of the M-step). Either one of these initialization schemes can be employed in a straightforward manner. Specifically, to obtain an initial estimate of the shape labels, one can randomly label the various shapes in the database into n different labels with n denoting the number of classes that we wish to classify the database into. So far, we have limited ourselves to classification of two shape classes so $n = 2$. The random labels serve as the initial shape labels in our EM algorithm. Alternatively, to obtain an initial estimate of the level set for each shape class, one can randomly divide the database into n groups. Based on this random grouping scheme, the arithematic average level set for each assigned group is calculated. The resulting average level sets serve as the initial level sets in our EM algorithm. Through various experiments, we have found that both of these approaches are robust to the initial random grouping scheme.

Besides providing the initialization to the EM algorithm, the parameter σ also needs to be chosen prior to starting our algorithm. In all our experiments, we empirically chose $\sigma = \frac{1}{2}\sqrt{M^2 + N^2}$ for 2D cases. In the case of 3D, we use $\sigma = \frac{1}{2}\sqrt{M^2 + N^2 + O^2}$. Intuitively, this value of σ was chosen to ensure that it is large enough so that various level set differences will lie well within the Gaussian distribution.

[4] As weighted averages of different distance maps, even with large number of EM iterations, the estimates of X_1 and X_2 will continue to be stable level set functions.

4. Generalization

In this section, we offer a few generalizations to the methodology developed in Section 3. Specifically, in Section 4.1, we illustrate how our algorithm can be applied to classify shapes having differing topologies. In Section 4.2, we illustrate how our algorithm can be used to classify more than just two shape classes. And finally, in Section 4.3, we illustrate how our algorithm can be used to classify 3D shapes. Complex synthetic 2D and 3D shapes are employed in each subsection to complement the text, and to serve as a vehicle to explain and highlight each generalization of our methodology.

4.1. Classification of shapes with different topologies

In Fig. 1, we show the outlines of 12 aligned synthetically generated hand written fours. Notice that some outlines require two separate curves while other only require one curve in describing the shapes. Level set methods afford us the luxury of describing these shapes without any additional overhead. In fact, topological different shapes are transparent to the users in this classification scheme. Each of these shapes are embedded in a 200×200 pixel level set image. It took 7 iterations for our algorithm to converge to the results shown in Table 1 and Fig. 2. Based on Fig. 2, it appears that Class A includes those fours with an "open" top and Class B includes those fours with a "closed" top. Of note, as shown in this example, our algorithm does not classify shapes based on topology. Specifically, the topology of Shape #10 is closer to Class B (requiring two curves) but is classified as Class A because it has an "open" top. On a 933 MHz Pentium 3 personal computer, this particular simulation took approximately 8.80 s to converge.

4.2. Classification of arbitrary number of shape classes

Up to this point, we have only shown derivations and simulation results for the classification of two shape classes. We demonstrate, in this section, how our algorithm can be generalized, in a straightforward manner, to handle classification of more than two shape classes. Let n denote the number of shape classes that one would like to separate the database into. Redefine the binary shape labels as C_{lh} where $h = 1,\ldots,n$ denotes the shape class number and $l = 1,\ldots,L$ again denotes the particular example shape within the database so that

$$\sum_{h=1}^{n} C_{lh} = 1 \quad \forall l = 1, \ldots, L, \tag{11}$$

and

$$p(C_{lh}) = \frac{1}{n} \quad \forall h = 1, \ldots, n \quad \text{and} \quad \forall l = 1, \ldots, L. \tag{12}$$

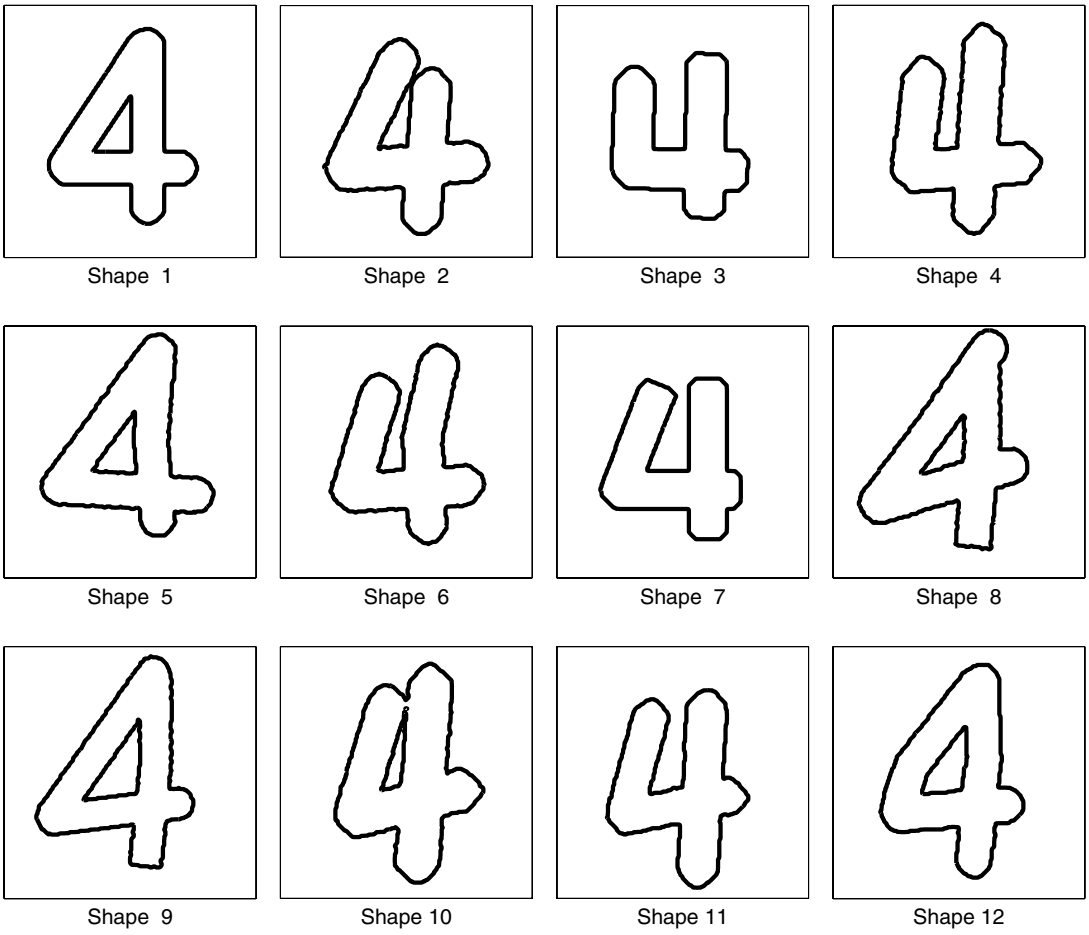

Fig. 1. Database of 12 contours of the number four.

Table 1
Classification of the number four by our EM algorithm

Shape	1	2	3	4	5	6	7	8	9	10	11	12
Class	B	A	A	A	B	A	A	B	B	A	A	B

The EM algorithm classified the fours into class A or class B.

As an example, if example shape #5 is classified as belonging to shape class #3, then

$$C_{5,3} = 1 \tag{13}$$

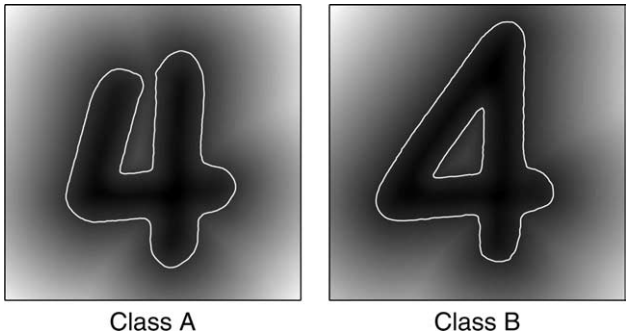

Fig. 2. Level set estimates of the two shape classes with the zero level set marked in white.

and

$$C_{5,h} = 0 \quad \forall h = 1,\ldots,n \text{ except } h = 3. \tag{14}$$

Besides redefining the class label C, we also need to redefine X so that

$$X = \{X_1, X_2, X_3, \ldots, X_n\}. \tag{15}$$

Based on these new definitions, the E-step calculates

$$\langle C_{lh} | Y_l, X^{[k]} \rangle = \frac{\prod_{i=1}^{\mathcal{N}} \exp(-(Y_{l_i} - X_{h_i}^{[k]})^2/2\sigma^2)}{\sum_{\tilde{h}=1}^{n} \left\{ \prod_{i=1}^{\mathcal{N}} \exp(-(Y_{l_i} - X_{\tilde{h}_i}^{[k]})^2/2\sigma^2) \right\}} \quad \forall h = 1,\ldots,n, \tag{16}$$

and the M-step calculates

$$X_{h_i}^{[k+1]} = \frac{\sum_{l=1}^{L} \langle C_{lh} | Y_{l_i}, X_i^{[k]} \rangle Y_{l_i}}{\sum_{l=1}^{L} \langle C_{lh} | Y_{l_i}, X_i^{[k]} \rangle} \quad \forall i = 1,\ldots,\mathcal{N} \quad \text{and} \quad \forall h = 1,\ldots,n. \tag{17}$$

To illustrate the classification of more than two shape classes, we show in Fig. 3 a database consisting of 12 aligned fighter planes. The goal is to separate this database into three separate groups. Grossly examining this database, one criteria for separation is likely based on

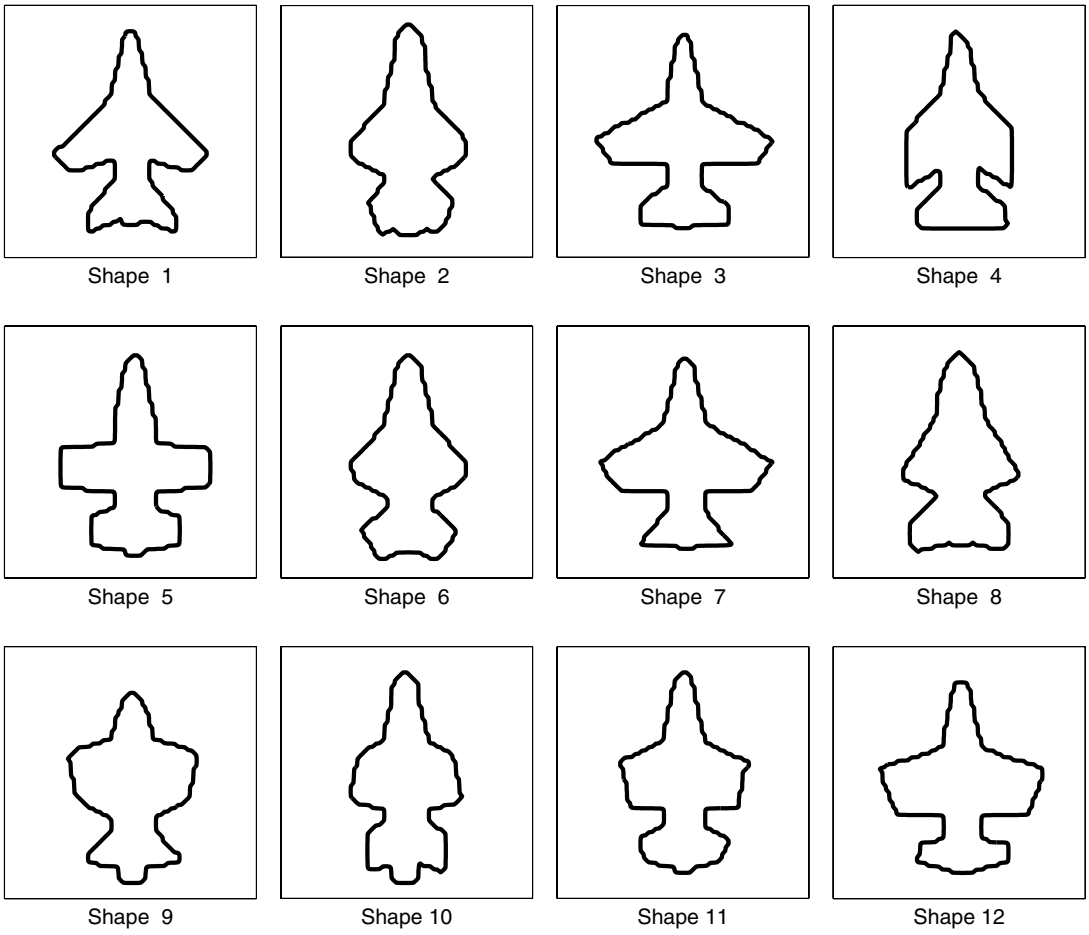

Fig. 3. Database of 12 fighter planes.

Table 2
Classification of the fighters by our EM algorithm

Shape	1	2	3	4	5	6	7	8	9	10	11	12
Class	B	A	B	A	B	A	B	A	C	A	C	B

the length of the wing span. We embedded each of these fighter planes in a 75×75 pixel level set image. It took 7 iterations for the algorithm to converge to the results shown in Table 2 and Fig. 4. Based on these results, it appears that Class A contains those planes with a short wing span, Class B contains those planes with a long wing span, and Class C contains those planes with medium wing span. On a 933 MHz Pentium 3 personal computer, this simulation took approximately 2.97 s to converge.

Fig. 5 shows a database consisting of 15 aligned real fish silhouettes. The task is to classify the silhouettes into three groups. Visually examining the database, it is relatively easy to pick out the silhouettes that belong to the "shark" shape class. However, separating the rest of the silhouettes into the remaining two groups is not trivial. Each of these silhouettes are embedded in a 500×500 pixel level set image. It took 7 iterations for the algorithm to converge to the results shown in Table 3 and Fig. 6. Indeed as expected, Class B is the "shark" group. Class A and Class C differ from one another based on the width or girth of the fish silhouettes. On a 933 MHz Pentium 3 personal computer, this simulation took approximately 56.82 s to converge. The longer elapse time for this simulation is mainly due to the larger size of the level set images.

4.3. Extension to 3D

The generalization of our EM-based shape classification algorithm to 3D is also straightforward. The only difference here is that the bounding surfaces of each 3D shape is embedded as the zero level set of a signed distance function with negative distances assigned to the inside and positive distances assigned to the outside of the 3D object.

Fig. 7 shows a database of seven aligned 3D cartoon vehicles. Each of these shapes is embedded in a $50 \times 50 \times 50$ pixel level set volume. It took 10

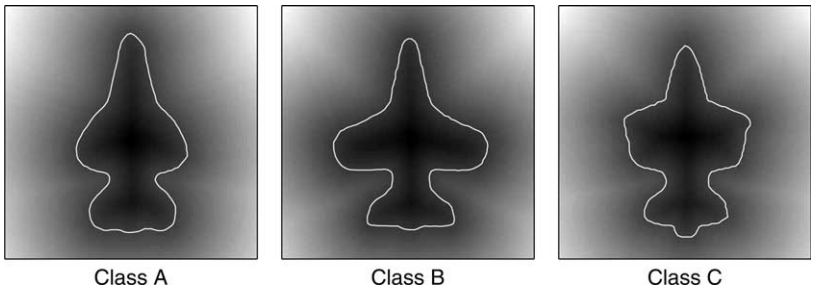

Fig. 4. Level set estimates of the three shape classes with the zero level set marked in white.

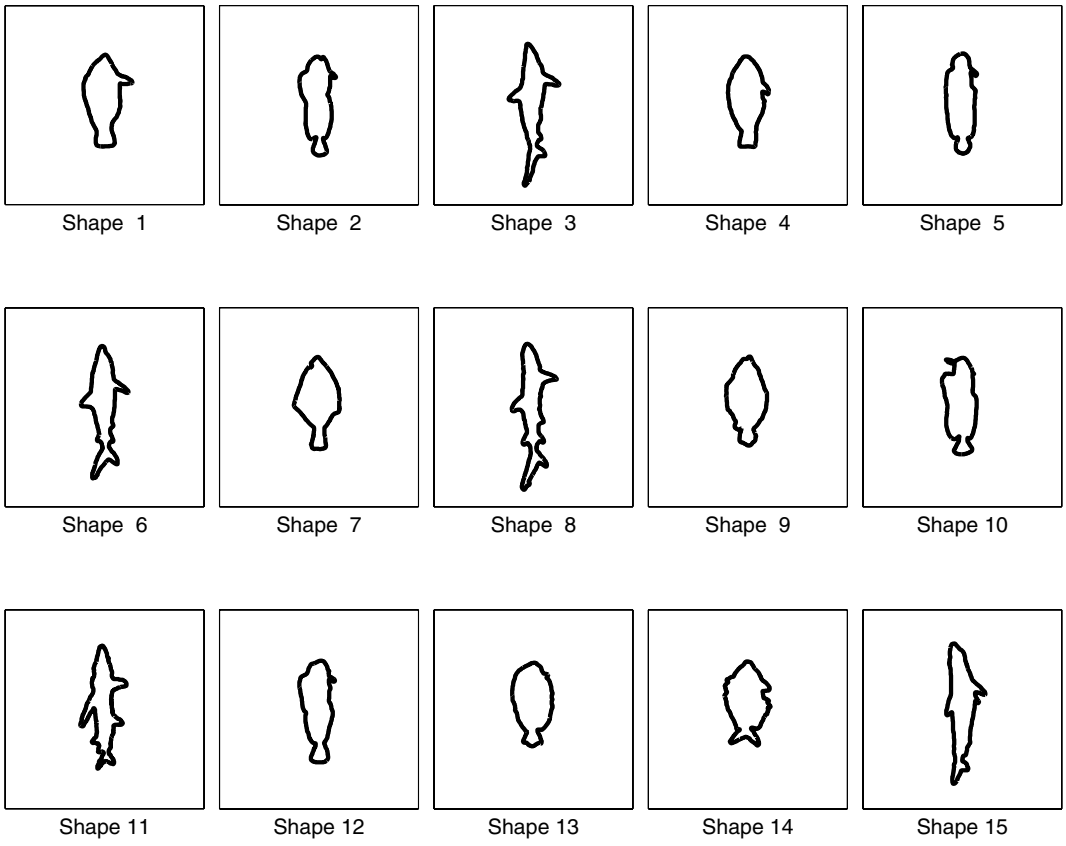

Fig. 5. Database of 18 fish silhouettes.

Table 3
Classification of the fish silhouettes by our EM algorithm

Shape	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Class	A	C	B	A	C	B	A	B	A	C	B	C	A	A	B

The EM algorithm classified the fish silhouettes into class A, B, or C.

iterations for the algorithm to converge to the results shown in Table 4 and Fig. 8. Based on Fig. 8, it appears that the classification is done based on the size of each vehicle's trunk. Specifically, Class A includes those vehicles with a low or flat trunk while Class B includes those vehicles with a elevated trunk. And as expected, Table 4 shows that vehicles #2, #3 and #4 are labeled as Class A, and vehicles #1, #5, #6 and #7 are labeled as Class B. On a 933 MHz Pentium 3 personal computer, this particular simulation took approximately 8.30 s to converge.

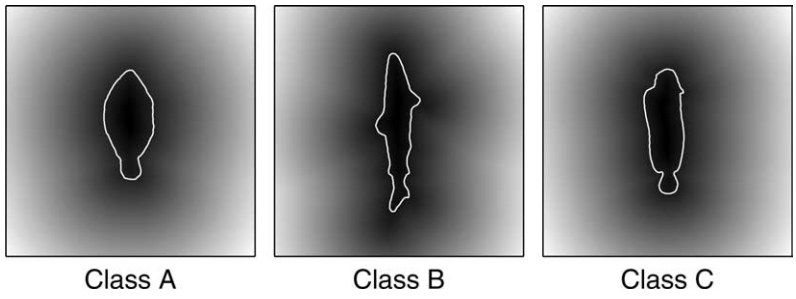

Fig. 6. Level set estimates of the three shape classes with the zero level set marked in white.

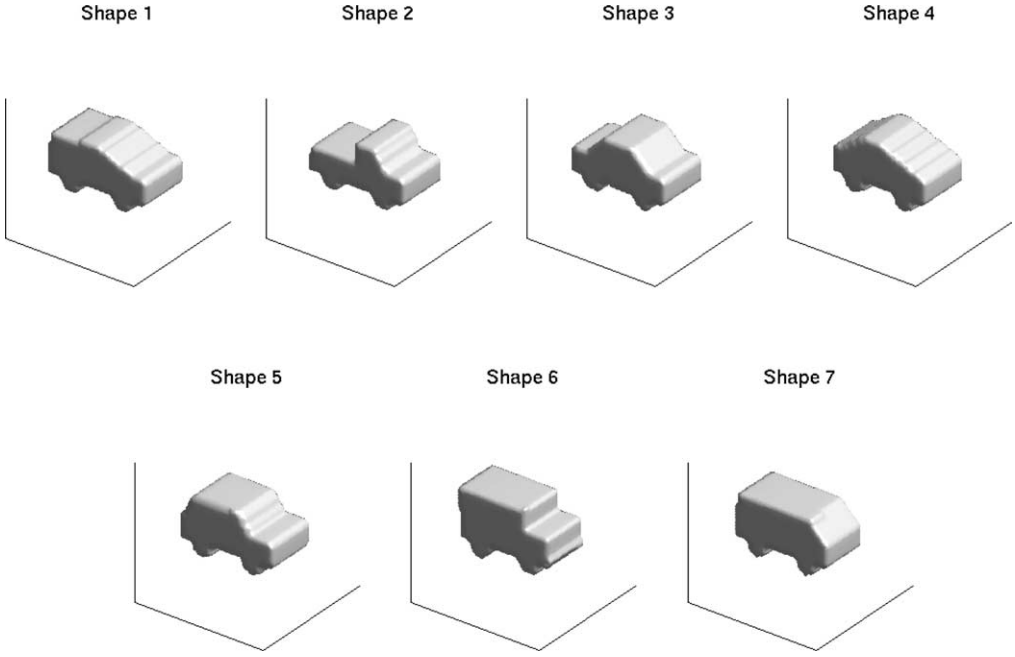

Fig. 7. Database of seven vehicles.

Table 4
Classification of the vehicles by our EM algorithm

Shape	1	2	3	4	5	6	7
Class	B	A	A	A	B	B	B

The EM algorithm classified the vehicles into class A or class B.

5. Applications

In this section, we present two medical applications to illustrate the performance of our algorithm. Section 5.1 illustrates a 2D example, while Section 5.2 illustrates a 3D example.

5.1. Chest radiographs of normal and emphysematous patients

Emphysema is a lung disease which involves the destruction of alveoli and its surrounding tissue. Typical findings on chest X-rays of emphysema patients include hyperinflation of the lung fields and flattened diaphram. Fig. 9 shows a database consisting of eight sets of aligned contours with each set representing the outlines of the right and left lung fields from a different patient's chest radiograph. The eight patients' chest radiographs have been classified a priori by a radiologist as having either normal or emphysematous lung.

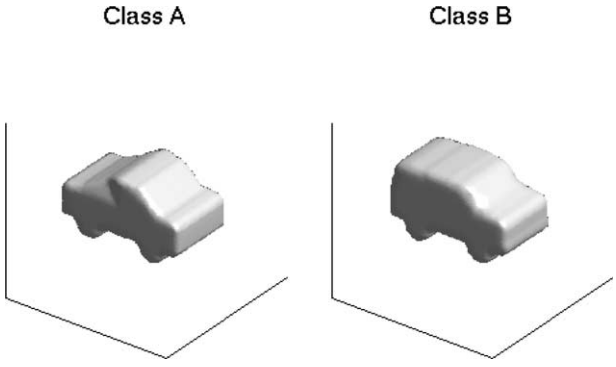

Fig. 8. Shape estimates of the two shape classes.

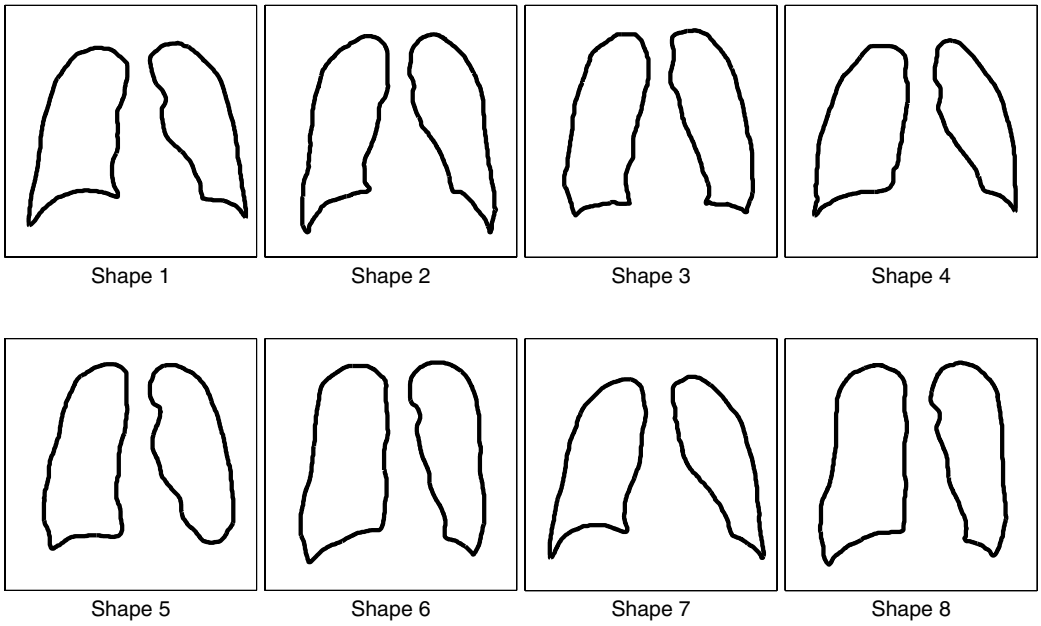

Fig. 9. Database of eight contours outlining the right and left lung fields from a collection of chest radiographs.

The experimental results based on our algorithm are shown in Table 5 and Fig. 10. The two images shown in Fig. 10 represent the level set functions of the two shape classes with the zero level set of each level set function outlined in white. The white contours can be thought of as the representative shape of each shape class. Table 5 shows that the grouping scheme generated by our EM algorithm exactly matched the one generated by the radiologist. In particular, notice that Class A corresponds to normal and Class B corresponds to diseased patients. Not surprisingly, Class B's representative shape shows the hyperinflated lung as well as the flatten diaphram typical of emphysematous patients. For this particular experiment with each shape having 300×300 pixels, it took 5 iterations to converge requiring approximately 1.67 s on an Intel Xeon 4.4 GHz dual processor computer.

Table 5
Comparion of chest radiograph labelings between a radiologist and our EM algorithm

Shape	1	2	3	4	5	6	7	8
Radiologist	Nl	Nl	Dz	Nl	Dz	Dz	Nl	Dz
EM	A	A	B	A	B	B	A	B

The radiologist classified the chest radiographs into normal (Nl) or one with emphysema disease (Dz). The EM algorithm classified them into class A or class B.

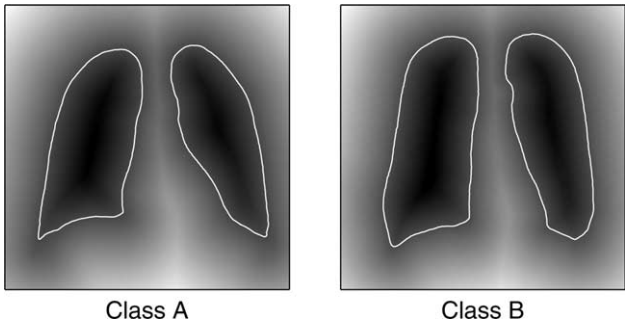

Fig. 10. Level set estimates of the two shape classes with the zero level set marked in white.

5.2. Cerebellum of neonates with Dandy–Walker syndrome

Dandy–Walker Syndrome is a congenital brain malformation associated with agenesis of the cerebellum. Our task is to separate the aligned cerebellum database shown in Fig. 11 into normal cerebellums and those afflicted with Dandy–Walker Syndrome. The eight cerebellums in the database are known a priori to either have the disease or not. These eight cerebellums are hand-segmented from eight MRI data set.

The experimental results based on our algorithm are shown in Table 6 and Fig. 12. The two shapes shown in Fig. 12 are the representative shapes of the two shape classes. Table 6 shows that the grouping scheme generated by our EM algorithm matched the correct answer. In particular, notice that Class A corresponds to normal and Class B corresponds to diseased patients. Class B's representative shape shows partial agenesis of the superior aspect of the cerebellum. For this 3D experiment, each example shape's level set function is represented by $256 \times 256 \times 50$ pixels. In terms of processing time, 10 iterations were required for convergence taking

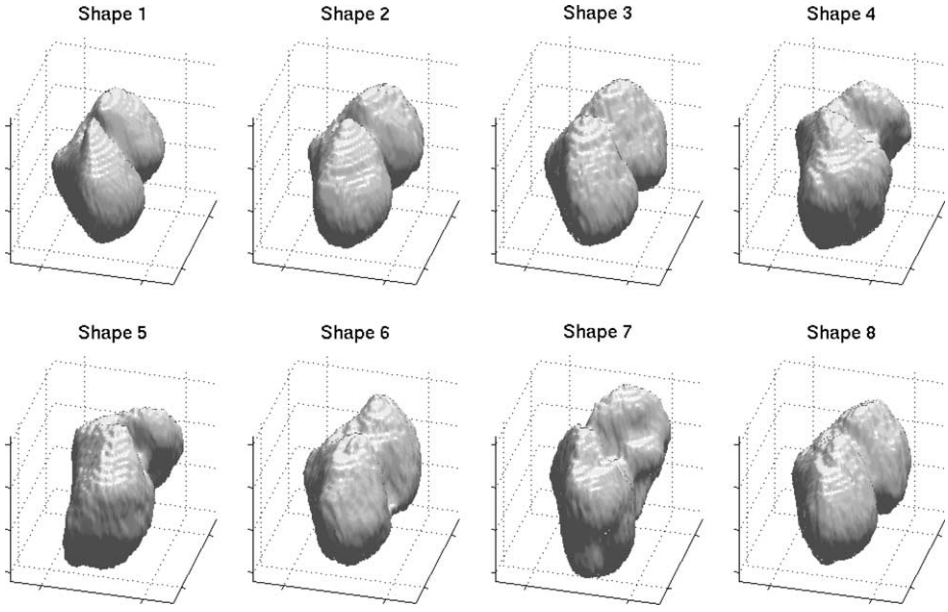

Fig. 11. Database of normal and diseased cerebellums.

Table 6
Comparion of labelings between the truth and our EM classifier

Shape	1	2	3	4	5	6	7	8
Diagnosis	Nl	Nl	Nl	DWS	DWS	Nl	DWS	Nl
EM Classifier	A	A	A	B	B	A	B	A

The diagnosis of the cerebellum is either normal (Nl) or one with Dandy–Walker syndrome (DWS). The EM algorithm classified the cerebellums into class A or class B.

approximately 39 s on an Intel Xeon 4.4 GHz dual processor computer.

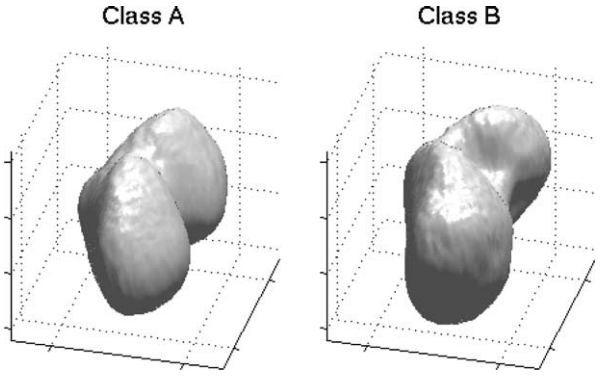

Fig. 12. Shape estimates of the two classes.

6. Conclusion and future research directions

We have outlined a novel approach for statistical shape classification and estimation based on the EM algorithm and the level set representation of shapes. The approach we have outlined is fast as demonstrated by the short processing time, and flexible as it can handle the classification of complex shapes (including those that have dimensionality greater than two and those with complex topologies), and the classification of multiple shape classes. Finally, the results of the two medical applications we showed are encouraging as they illustrated the accuracy of our methodology suggesting potential applications of this method for computer aided diagnosis.

We are currently exploring the use of other implicit shape representations (other than distance transforms) that will not cause any inconsistencies in the shape representation during the calculation of the M-step (i.e. the level set function estimated in the M-step is no longer a distance function). We are also interested in extending this formulation to enable it to provide users with information regarding the specific differences among the different shape classes as this will be immensely helpful to clinicians. Finally, we are actively exploring the use of Eq. (8) as a similarity measure in matching a query shape to a large database for shape retrieval with applications to hand writing recognition.

Acknowledgements

The authors thank Drs. Andrea Mewes and Catherine Limperopoulos for their help in the segmentation of the cerebellum data set.

References

Basri, R., Costa, L., Geiger, D., Jacobs, D., 1995. Determining the similarity of deformable shapes. IEEE Workshop: Phys Based Modeling in Comput Vis. pp. 135–143.

Bloomenthal, J., Lim, C., 1999. Skeletal methods of shape manipulation. In: IEEE Proceedings of the Shape Modeling and Applications. pp. 44–47.

Cesar, R., Costa, L., 1996. Towards effective planar shape representation with multiscale digital curvature analysis based on signal processing techniques. Pattern Recogn 29 (9), 1559–1569.

Cohen, I., Ayache, N., Sulger, P., 1992. Tracking points on deformable objects using curvature information. ECCV, 458–466.

Del Bimbo, A., Pala, P., 1997. Visual image retrieval by elastic matching of user sketches. IEEE Trans. PAMI 19, 121–132.

Dempster, A., Laird, N., Rubin, D., 1977. Maximum-likelihood from incomplete data via the EM algorithm. J. Royal Stat. Soc. Ser. B 39, 1–38.

Dionisio, C., Kim, H., 2002. A supervised shape classification technique invariant under rotation and scaling. International Telecommunications Symposium.

Gdalyahu, Y., Weinshall, D., 1999. Flexible syntactic matching of curves and its application to automatic hierarchical classification of silhouettes. IEEE Trans. PAMI 21, 1312–1328.

Golland, P., Grimson, E., Kikinis, R., 1999. Statistical shape analysis using fixed topology skeletons: corpus callosum study. IPMI, 382–387.

Kawata, Y., Niki, N., Ohmatsu, H., Kakinuma, R., Eguchi, K., Kaneko, M., Moriyama, N., 1998. Classification of pulmonary nodules in thin-section CT images based on shape characteristics. IEEE Trans. Nucl. Sci. 45, 3075–3082.

MacQueen, J., 1965. Some methods for classification and analysis of multivariate observations. In: Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, vol. 1.

Leventon, M., Grimson, E., Faugeras, O., 2000. Statistical shape influence in geodesic active contours. IEEE Conf. Comp. Vision Patt. Recog. 1, 316–323.

Li, X., Woom, T., Tan, T., Huang, Z. Decomposing polygon meshes for interactive applications. ACM Symposium on Interactive 3D Graphics 3/2001.

Osher, S., Sethian, J., 1988. Fronts propagation with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations. J. Comput. Phys. 79, 12–49.

Paragios, N., Rousson, M., 2002. Shape priors for level set representations, ECCV June 02, Copenhagen, Denmark.

Staib, L., Duncan, J., 1992. Boundary finding with parametrically deformable contour models. IEEE Trans. Patt. Anal. Mach. Intell. 14, 1061–1075.

Tsai, A., Zhang, J., Willsky, A., 2001. EM algorithms for image processing using multiscale models and mean field theory, with applications to laser radar range profiling and segmentation. Opt. Eng. 40, 1287–1301.

Tsai, A., Yezzi, A., Wells, W., Tempany, C., Tucker, D., Fan, A., Grimson, E., Willsky, A., 2003. A shaped-based approach to segmentation of medical imagery using level sets. IEEE Trans. Med. Imaging 22, 137–154.

Tsai, A., Wells, W., Warfield, S., Willsky, A., 2004. Level set methods in an EM framework for shape classification and estimation. MICCAI, 1–9.

Wells, W., Grimson, W., Kikinis, R., Jolesz, F., 1996. Adaptive segmentation of MRI data. IEEE Trans. Med. Imaging 15, 429–442.