# Learning Sparse Kernels from 3D Surfaces for Heart Wall Motion Abnormality Detection

**Glenn Fung, Sriram Krishnan, R. Bharat Rao**
IKM CKS, Siemens Medical Solutions USA

**Hui Chen**
Biometrics Solutions, Motorola

## Abstract

Coronary heart disease (CHD) is a global epidemic that is the leading cause of death worldwide. CHD can be detected by measuring and scoring the regional and global motion of the left ventricle (LV) of the heart. This project describes a novel automatic technique which can detect the regional wall motion abnormalities of the LV from echocardiograms. Given a sequence of endocardial contours extracted from LV ultrasound images, the sequence of contours moving through time can be interpreted as a three-dimensional (3D) surface. From the 3D surfaces, we compute several geometry-based features (shape-index values, curvedness, surface normals, etc.) to obtain histograms-based similarity functions that are optimally combined using a mathematical programming approach to learn a kernel function designed to classify normal vs. abnormal heart wall motion. In contrast with other state-of-the-art methods, our formulation also generates sparse kernels. Kernel sparsity is directly related to the computational cost of the kernel evaluation, which is an important factor when designing classifiers that are part of a real-time system. Experimental results on a set of echocardiograms collected in routine clinical practice at one hospital demonstrate the potential of the proposed approach.

## Introduction

Cardiovascular Disease (CVD) is a global epidemic that is the leading cause of death worldwide (17mil. deaths per year) (World Health Organization 2004). Coronary Heart Disease (CHD) accounts for more than half the CVD deaths and it is the single largest killer in the world. It is well-known that early detection (along with prevention) is an excellent way of controlling CHD. CHD can be detected by measuring and scoring the regional and global motion of the left ventricle (LV) of the heart. It typically results in wall-motion abnormalities, i.e., regional segments of the LV wall move abnormally (move weakly, not at all, or out of sync with the rest of the heart), or indeed the entire heart, is compromised. The most used method to image the LV is the echocardiogram - an ultrasound video of different 2-D cross-sections of the LV. Unfortunately, echocardiograms are notoriously difficult to interpret, even for the experts. Recent studies have shown that even world-class experts agree on their diagnosis only 80% of the time (Hoffmann *et al.* 2002), and intra-observer studies have shown a similar variation when the expert reads the same case twice at widely different points in time. Therefore, there is a need for an automated reader-assistance tool system that can provide objective diagnostic assistance, particularly for the less-experienced cardiologist.



In this paper, we proposed a novel approach to the automatic wall-motion abnormality detection task from echocardiograms. Prior approaches to this problem relied heavily on prior knowledge from the doctors to either calculate some diagnosis-relevant features or to estimate the relation among the different segments of the heart (Qazi *et al.* 2007). In contrast, in this paper, we assume that few or none extra information is provided. Thus the classification has to rely purely on information about the physical movement of the walls that can be estimated and approximately extracted from the echocardiograms.

Furthermore, we proposed an algorithm that is similar in nature to the algorithms proposed in (Lanckriet *et al.* 2003; Bach, Lanckriet, & Jordan 2004; Fung *et al.* 2004) and more recently (Kim, Magnani, & Boyd 2006), to automatically learn the best kernel for a given classification task. However, in contrast with these methods our proposed algorithm has two main differences:

- It learns kernels that depend on a small subset of the given predefined initial kernel function family.
- The resulting learned kernels will depend on a minimal set of the original basis kernel functions (training data).

This is very important when creating a classifier for use in a real-time system (which is the case here), as it can reduce the testing computational time considerable.

We work with the sequence of endocardial contours extracted from LV ultrasound images. The sequence of contours evolving through time can be represented as a 3D surface, see Figure  for an example.

Given a 3D surfaces representing a sequence of contours evolving through time, we perform the following steps:

1. Compute shape index values, curvedness and surface normals, and construct histograms which approximate distributions of these features over the surfaces.

2. Calculate several different initial measurements of distributions difference/similarity for the distribution obtained in the previous step: $\lambda^2$-divergence (Schiele & Crowley 2000), Kullback-Leibler (KL) divergence (Cover & Thomas 1991), Bhattacharyya distance (Therrien 1989), $L_1$ norm, $L_2$ norm (Golub & Van Loan 1996) and the sample correlation coefficient (Lee 2004). In this way, we obtain initial similarity matrices (kernels) to estimate the similarity between surfaces.

3. We solve a mathematical-programming-based formulation to combine the kernels in an optimal way to learn a composed kernel that is designed specifically to classify the sequence of contours into abnormal or normal. The final learned kernel is sparse in the sense that it only depends on a small subset of the initial similarity function provided and that it only depends on an small set of basis functions.

The following section provides some introductory medical background on cardiac ultrasound and the standard methodology used by cardiologists to score wall-motion abnormalities. We also describe our real-life dataset, which consists of echocardiograms used by cardiologists at several leading hospitals to diagnose wall-motion abnormalities. In Section , we present an overview of the image processing algorithm that precedes our surface-based classification stage which includes surface interpolation, curvature estimation and initial similarity functions calculation (Section ). In section , we present a state-of-the-art classification technique to obtain an optimal linear combination of initial kernels (Fung *et al.* 2004) and propose a new convex formulation that incorporates the sparsity requirement into the optimization problem. We present our numerical results in Section , and finally we provide some conclusions and ideas for future work.

## Medical Background Knowledge

### Divisions of the Heart

There are many imaging modalities that have been used to measure and assess left ventricular function for clinical management and research; for this project we chose to use echocardiography.

The Cardiac Imaging Committee of the Council on Clinical Cardiology of the American Heart Association has created a standardized recommendation for the orientation of the heart, angle selection and names for cardiac planes and number of myocardial segments (Cerqueira *et al.* 2002). This is the standardization used in this project. Echo images are collected from four standard views: apical 4 chamber (A4C), apical 2 chamber (A2C), parasternal long axis (PLAX) or apical 3 chamber (A3C), and parasternal short axis (PSAX). The planes used to cut the heart to display these standard views are displayed in Figure from reference (Catherine M. Otto 2000). The left ventricle (LV) is divided into 17 myocardial segments that can be seen in one or two of the standard views described above.

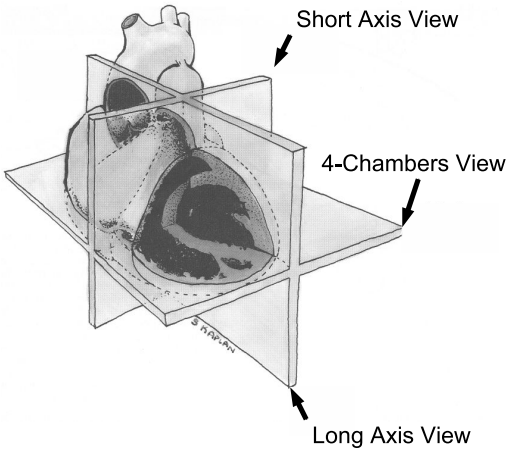


Figure 1: The planes used to cut the heart to display the standard views: the long-axis view extends from the LV apex through the aortic valve plane. The short-axis view is perpendicular to the long-axis view resulting in a circular view of the LV. The four-chamber view is perpendicular to both the long and short-axis views and includes the left and right ventricle, and left and right atrium (taken from reference ).

### Description of the Data

The data is based on standard adult transthoracic B-mode ultrasound images collected from the four standard views described previously. Currently we only utilize two of the four possible views - A4C and A2C (an example of an A4C image can be found in Figure ) - which show 12 of the 16 total segments (we ignore the apex, segment 17, since it is nearly impossible to measure). But these 12 are enough to achieve our goal of assessing wall motion abnormality. Even though we have images at different levels of stress (resting, low-dose stress, peak-dose stress, recovery) this work is based on images taken when the patient was resting. The goal is to automatically provide an initial score, or classification, to determine whether a heart wall is normal or abnormal given the ultrasound (A4C view or A2C view).

The data consists of 320 cases for which we have associated images as well as ground truth; all of which were generated using pharmacological stress, which allows the physician to control the amount of stress a patient experiences (in this case induced by dobutamine). All the cases have been labeled at the segment level by a group of trained cardiologists; this is what we refer to as ground truth. Each of the 16 segments were labeled 1 - 5 (1 = normal, 2 = hypokinetic, 3 = akinetic, 4 = dyskinetic, 5 = aneurysm). For simplification purposes we converted this 5-class problem to a binary class problem (1 = normal, 2 - 5 = abnormal). About a ramdomly chosen 70% of the 320 available cases were used for training (224 cases) and the remaining 30% of the cases (96 cases) were ear-marked as the final test set; The wall level classification labels can be obtained from the segment level labels by applying the following definition: a heart wall is considered abnormal if one or more segments are abnormal.

### Preparation of the Data

Our application consists of two main parts: image processing and classification. The echos are run through an algorithm which automatically detects and tracks both the interior (endocardial) and exterior (epicardial) borders of the LV (Comaniciu 2003; Comaniciu, Zhou, & Krishnan 2004). Motion interferences (e.g. probe motion, patient movement, respiration, etc.) are compensated for by using global motion estimation based on robust statistics outside the LV. This is done so that only the heart's motion is analyzed. Then, for each view, a 3D (2D + time) shape is obtained obtained which is based on the interior contour tracked through time, form the basis for the regional wall motion classification.

## Image processing

The first step toward classification of the heart involves automatic contour generation of the LV (Georgescu *et al.* ). Ultrasound is known to be noisier than other common medical imaging modalities such as MRI or CT, and echos are even worse due to the fast motion of the heart muscle and respiratory interferences. The framework used by our algorithm is ideal for tracking echo sequences since it exploits heteroscedastic (i.e. locationdependent and anisotropic) measurement uncertainties. The process can be divided into 2 steps: border detection and border tracking. Border detection involves localizing the LV on multiple frames of the image clip (shown in Figure as a box drawn around the LV), and then detecting the LV's shape within that box. Seventeen control points are placed along the interior border of the LV to show where the border was detected. These points are then extended outwards to find the external (epicardial) border of the LV. Border tracking involves tracking both these contours together from one frame to the next through the entire movie clip. Motion interferences (e.g. probe motion, patient movement, respiration, etc.) are compensated for by using global motion estimation based on robust statistics outside the LV. This global motion estimation can be seen in Figure as a vertical (red) line near the center of the image.

Note that even though our system tracks both the inner and outer contour, for this paper we only make use of the inner contour. Future extensions of this work would address how to combine information from both contours to improve classification.

After detection and tracking a three dimensional shape is generated from the inner contour tracked through time (For example of such 3D shapes see Figure .(a).

## Standard similarity functions between surfaces

In this section, we describe the detail procedure to calculate features from the given 3D surface representation of the temporal sequence of the LV and how to evaluate the difference between surfaces in terms of different measurements. The process consists of several steps:

1. First, given a sequence of endocardial contours extracted from LV ultrasound images, we perform a temporal and spatial interpolation to form a dense 3D surface.

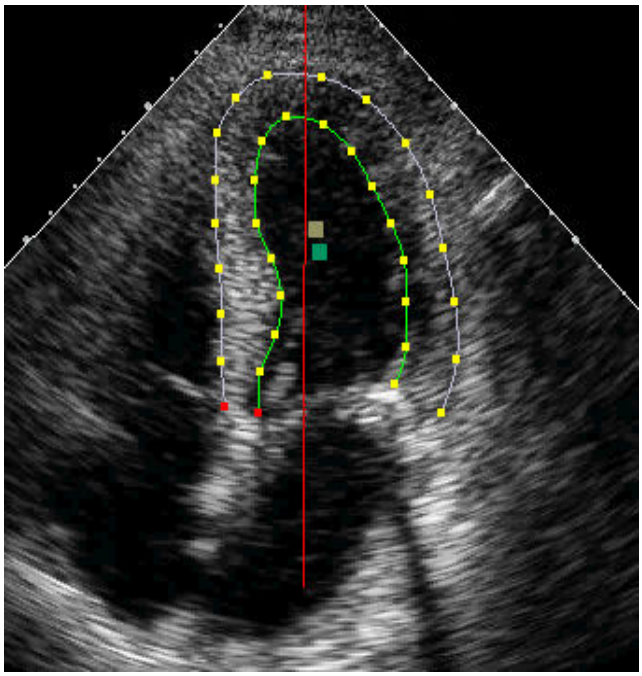

Figure 2: One frame from an A4C image clip with the outer and inner contour control points shown. The (red) vertical line shows use of global motion compensation, and the two squares denote the centers of the individual contours.

2. We use differential geometry to compute features from the surfaces, for instance, the shape index values, curvedness, normal directions of the 3D surface. They are used to construct a histogram-based representation which captures the characteristics of the surface. Histograms have proven to be simple and powerful representations for pattern classification and object recognition.
3. Next we compare the histograms using the following different distance/similarity measurements: $\lambda^2$-divergence (Schiele & Crowley 2000), Kullback-Leibler (KL) divergence (Cover & Thomas 1991), Bhattacharyya distance (Therrien 1989), $L_1$ norm, $L_2$ norm (Golub & Van Loan 1996) and the sample correlation coefficient (Lee 2004).

### Surface Interpolation

The original contour sequence extracted from the ultrasound images is very sparse, it has only 17 points on the contour and it has 9 frames on average. In order to estimate curvatures accurately, it is necessary to interpolate the original contour sequence to form a denser surface in the spatial and the temporal domain.

Given the contour sequence, we performed a spline interpolation in the temporal domain and then performed an spatial interpolation along each the contour. Figure .(a) shows the original contour surface for a particular segment and an entire contour and Figure .(b) shows the spline interpolated surface for the particular segment and the entire contour. Note that as desired, a dense and smooth surface is obtained after the spatial and temporal interpolation.

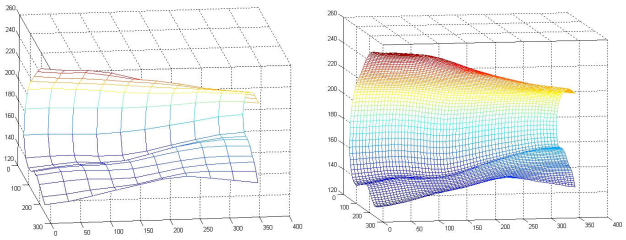

Figure 3: **(a)**(left) The original contour surface for a A4C view . **(b)**(right) shows the corresponding spline interpolated surface.

## Computation of Surface Properties

**Curvature Estimation** In order to estimate curvatures, we fit a biquadratic surface of the form:

$$f(x,y) = ax^2 + by^2 + cxy + dx + ey + f \quad (1)$$

to a local area and use a least squares method to estimate the parameters of the quadratic surface. Then we use differential geometry to calculate the surface normal, Gaussian and mean curvatures and the principal curvatures (Flynn & Jain 1989). Given the first order derivatives $f_x$, $f_y$ and the second order derivatives $f_{xx}$,$f_{xy}$ and $f_{yy}$, the surface normal $\vec{n}$, Gaussian curvature $K$, mean curvature $H$, principal curvatures $k_{1,2}$ are respectively given by equations (2), (3), (4) and (5) presented below.

$$\vec{n} = \frac{(-f_x, -f_y, 1)}{\sqrt{1 + f_x^2 + f_y^2}} \quad (2)$$

$$K = \frac{f_{xx}f_{yy} - f_{xy}^2}{(1 + f_x^2 + f_y^2)^2} \quad (3)$$

$$H = \frac{f_{xx} + f_{yy} + f_{xx}f_y^2 + f_{yy}f_x^2 - 2f_x f_y f_{xy}}{2(1 + f_x^2 + f_y^2)^{1.5}} \quad (4)$$

$$k_{1,2} = H \pm \sqrt{H^2 - K} \quad (5)$$

**Shape index and curvedness** Once we have the maximum principal curvature $k_1$ and minimum principal curvature $k_2$ we can compute shape index $S_i$ values and curvedness $C$ using the equations presented below.

The shape index $S_i$, a quantitative measure of the shape of a surface at a point $p$, is defined by:

$$S_i(p) = \frac{1}{2} - \frac{1}{\pi} tan^{-1} \frac{k_1(p) + k_2(p)}{k_1(p) - k_2(p)} \quad (6)$$

where $k_1$ and $k_2$ are maximum and minimum principal curvatures respectively (Dorai & Jain 1997). According to this definition, all shapes can be mapped into the interval $S_i \in [0, 1]$, depending on the shape index value (SIV), surfaces can be classified into different types. For example the SIV for a Dome will be in the range $[13/16, 15/16$ and the SIV for a spherical cup will be in $[0, 1/16)$ (Koenderink & Doorn 1992; Dorai & Jain 1997)). .

While the shape index is independent of scale, the curvedness is inversely proportional to the size of a surface patch, and it can be interpreted as a local measure of the curvedness of a given surface. The curvedness of a surface at point $p$, is defined by:

$$C = \sqrt{\frac{k_1^2(p) + k_2^2(p)}{2}} \quad (7)$$

In the areas of the flat surface a low curvedness exists and in the areas of sharp curvature a high curvedness exists. For instance, the curvedness of a planar surface is 0; the curvedness of a unit sphere is 1. The shape index $S_i$ and the curvedness $C$ complement each other in defining the local surface shape and size.

**Using Surface Normals** So far we have explained how to compute shape index values and curvedness for the points on the surface formed by endocardial or inner contours. Another useful attribute we chose to use is the mean surface normal value. Since we would like to have the histogram rotational invariant, we choose the mean surface normal as the reference normal. We then compute the dot product between the reference normal and other normals. The mean surface normal can be computed by the equation (8) where $M$ and $N$ is are the dimensions of the image (in pixels) normalized to be of unit length.

$$\bar{n} = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} \vec{n}_{ij} \quad (8)$$

## Comparison of surfaces

Having computed the three surface property values: shape index values, curvedness and the mean surface normal for every point on the surface formed by the endocardial contours, we then construct three (one per value) 1-D histograms by accumulating in particular bins. Once we have histogram representations, we compare them using different distance and similarity measurements. The choice of distance measurements between histograms may affect the classification performance. Therefore, we evaluate the difference using a set of measurements, including distribution-based distances ($\chi^2$- divergence, Bhattacharyya Distance, *KL* distance), Euclidean distances and sample correlation coefficients. These distances can be thought of as kernels which will be optimally linearly combined by the classification technique presented in Section .

Next, we present a quick summary of each of the distances used to compare the empirical distributions (histograms) obtained. For the rest of the section, $Q$ and $V$ will denote two given empirical distribution or 1-D histograms and $q_i$ and $v_i$ are the numbers corresponding to the $i$th bin.

**$\chi^2$- divergence** Since the histogram is an approximation of probability density function, it is natural to use $\chi^2$-divergence (Schiele & Crowley 2000), a widely used func-

tion to assess the dissimilarity between two probability density functions. The $\chi^2$-divergence function is given by:

$$\chi^2(Q,V) = \sum_i \frac{(q_i - v_i)^2}{q_i + v_i} \tag{9}$$

Note that the dissimilarity value $\chi^2(Q,V)$ lies between 0 and 2. If the two histograms are exactly the same, the dissimilarity will be zero. If the two histograms do not overlap with each other, it will achieve the maximum value of 2.

**Bhattacharyya Distance** The Bhattacharyya distance (Therrien 1989), is another distance measurement between two probability density functions that it is usually used to measure the separability of two given classes and its defined by:

$$L_p(Q,V) = (1 - \sum_i \sqrt{q_i * v_i})^{1/2} \tag{10}$$

***Kullback Leibler (KL)* Divergence** The KL divergence is another popular distance measurement between two probability density functions (Cover & Thomas 1991). We use the symmetric version of $KL$ distance with respect to both distributions, which is defined by:

$$\begin{aligned} D_{KL}(Q,V) &= 0.5 * (D(Q,V) + D(V,Q)) \\ &= \frac{(\int Q \log \frac{Q}{V} + \int V \log \frac{V}{Q})}{2} \end{aligned} \tag{11}$$

**$L_1$ and $L_2$ norm** Since 1-D histogram can be thought of as a vector, we use $L_1$ norm and $L_2$ (Golub & Van Loan 1996) norm to compute the distance between the two vectors $Q$ and $V$:

$$L_p(Q,V) = (\sum_i |q_i - v_i|^p)^{1/p}, \quad p \in \{1,2\} \tag{12}$$

**Sample Correlation coefficient** The sample correlation coefficient is a quantity to measure the strength of linear association between two variables. We use the following normalized correlation coefficient (Lee 2004).

$$r(Q,V) = \frac{\sum_i (q_i - \bar{Q})(v_i - \bar{V})}{\sqrt{(\sum_i (q_i - \bar{Q}))^2 (\sum_i (v_i - \bar{V}))^2}} \tag{13}$$

Once we have three 1-D histograms $h_1$, $h_2$ and $h_3$ for each feature (1:shape index, 2:curvedness and 3:surface normal), we use the mean value of the distances

$$D(S_i, S_j) = \frac{1}{3} \sum_{k=1}^{3} d(h_k^i, h_k^j) \tag{14}$$

to compute the difference between any two surfaces $S_i$ and $S_j$. In equation (14), the superscript $i$ and $j$ denote surface $i$ and surface $j$ respectively; where the distance $d(h^i, h^j)$ could be one of the five measurements mentioned above.

Note that every distance $D_k(S_i, S_j), \quad k \in \{1, \ldots, 5\}$ can be trivially converted to a similarity measure or a kernel function by normalizing the distance function to have values between 0 nd 1 an subtracting 1 form the normalized distance value. Another way to do it is to apply a Gaussian kernel transformation

$$K(S_i, S_j) = \exp(-\alpha D(S_i, S_j)^2)$$

In this paper we utilized the first option in order to avoid the need to tune the scaling parameter $\alpha$.

In the next section, we describe the mathematical-programming-based formulation utilized to optimally design a linear combination of the initial kernels for the classification problem at hand.

## Learning an optimal similarity function for heart wall motion abnormality detection

The use of a linear combination of kernels formed by a family of different kernel functions and parameters have been recently proposed (Hamers *et al.* 2003; Lanckriet *et al.* 2003; Fung *et al.* 2004). This transforms the problem of choosing a kernel model into one of finding an "optimal" linear combination of a set of pre-existing kernels. When the right regularization is applied, a final kernel is constructed according to the specific classification problem to be solved without sacrificing capacity control. By combining kernels we have the potential to improve prediction accuracy by "tailoring" a kernel specially design for the classification task at hand.

Next, we briefly present a formulation very similar to the one presented in (Fung *et al.* 2004) and that will be used in our numerical results.

### Automatic Kernel Proximal Support Vector Machine (AK-PSVM)

In (Fung *et al.* 2004) an algorithm that consists in a simple iterative procedure for generating an optimal Kernel Fisher Discriminant classifier was presented. The kernel model is assume to be a linear combination of $k$ kernels of the form:

$$K(A,B) = \sum_{i=1}^{k} a_i K_i(A,B) \tag{15}$$

where $A \in R^{m \times n}$ and $B \in R^{n \times l}$. The $K_i$ are members of a potentially larger pre-defined family of heterogeneous kernels. each $K_i$ is a kernel function that maps $R^{m \times n} \times R^{n \times l}$ into $R^{m \times l}$.

Using this approach, the task of finding an appropriate kernel can be incorporated into the optimization framework. Here we present a nearly identical formulation that is based on a proximal support vector machine (PSVM) (Fung & Mangasarian 2001) as the core classification model instead of the kernel Fisher discriminant.

$$\begin{aligned} \min_{(v,\gamma,a \geq 0) \in R^{m+1}} \quad & \nu \tfrac{1}{2} \| d - ((\textstyle\sum_{i=1}^{k} a_i K_i) v \; - e\gamma) \|^2 \\ & + \tfrac{1}{2} (v'v + \gamma^2 + a'a) \end{aligned} \tag{16}$$

The $i$ component of the vector $d$ represents the label of the corresponding data point (1:abnormal , -1: abnormal) and the $i$ row of $k_i$ (and hence of K) corresponds to a point in the training dataset. The corresponding nonlinear classifier to this nonlinear separating surface is given by:

$$\left(\sum_{j=1}^{k} a_j K_j(x', A'_{tr})\right) \upsilon - \gamma = \begin{cases} > 0, \text{ then } x \in A_+ \\ \leq 0, \text{ then } x \in A^- \end{cases} \tag{17}$$

Where $A^+$ and $A-$ denote the points in the positive and negative classes respectively and $A_{tr}$ is the training set. In our case the $K_i$ are defined by the similarity matrices obtained by using the six different measurements.

An advantage of this approach is that it can be solved very efficiently by alternate optimization since it can be seen as a biconvex optimization problem (Bezdek & Hathaway 2002). One drawback of this approach is that it does not produce sparse solutions with respect to the variables $a$ and $\upsilon$ which leads to solutions that depend on all the kernels $K_i$, furthermore since $\upsilon$ is very likely not to have zero components, in order to evaluate equation (17) for a new testing point $x$ we need to calculate the distance $D_i(x', y)$ between x and every point $y$ in the training set. This is potentially a problem for our application, since the classification has to be performed in real time. Calculating all the 6 distances to all the members of the training is very time consuming and hence not feasible for our system classifier. To address this problem we proposed an algorithm that is inspired by the AK-PSVM algorithm but that uses the 1-norm to regularize the variables $a$ and $\upsilon$ to obtain sparser solutions that depend on only a few $K_i$ and a minimal set of kernel basis.

## Sparse Automatic Kernel Support vector Machine

It is an accepted fact (Bradley & Mangasarian 1998; Guyon & Elisseeff 2003) that 1-norm-based classification problems are amenable to feature suppression. By suppressing components of the weight vector $\upsilon$, a nonlinear rectangular kernel is generated that utilizes only a few of the available training data to characterize a nonlinear separating surface. This results in substantial reduction in kernel data-dependence with test set correctness comparable to that obtained by using a conventional square kernel that depends on all the training data. This reduction in data dependence also results in faster classifiers that requires less storage.

We propose the following modification to formulation (16) that utilizes 1-norm regularization over the variables $a$ and $\upsilon$:

$$\begin{aligned} \min_{(\upsilon,\gamma,y,a\geq 0)\in R^{n+1+m}} \quad & \|y\|_2^2 + \nu_1\|\upsilon\|_1 + \nu_2\|a\|_1 \\ \text{s.t.} \quad & D\left((\textstyle\sum_{i=1}^{k} a_i K_i(A, A'))\upsilon - e\gamma\right) + y \geq e \\ & y \geq 0 \end{aligned} \tag{18}$$

In this formulation $D$ is a square diagonal matrix with $D_{ii} = d_i$. Similarly to formulation (16), problem (18) can be seen as a biconvex program of the form,

$$\begin{aligned} \min_{(S,T,U)\in(R^{m+1},R^k,R)} \quad & F(S,T,U) \\ \text{s.t.} \quad & G(S,T,U) \geq 0 \end{aligned} \tag{19}$$

where $S = \upsilon$ , $T = a$ and $U = (y, \gamma)$.

When $T = \hat{a}$ is fixed, problem (19) becomes the following constrained quadratic program (QP):

$$\begin{aligned} \min_{(\upsilon,\gamma,y)\in R^{n+1+m}} \quad & \|y\|_2^2 + \nu_1\|\upsilon\|_1 \\ \text{s.t.} \quad & D\left(\hat{K}\upsilon - e\gamma\right) + y \geq e \\ & y \geq 0 \end{aligned} \tag{20}$$

where

$$\hat{K} = \sum_{i=1}^{k} a_i K_i(A, A')$$

On the other hand when $S = \hat{\upsilon}$ is fixed, problem (19) becomes the following QP:

$$\begin{aligned} \min_{(\gamma,y,a\geq 0)\in R^{n+1+m}} \quad & \|y\|_2^2 + \nu_2\|a\|_1 \\ \text{s.t.} \quad & D\left((\textstyle\sum_{i=1}^{k} \Lambda_i a_i) - e\gamma\right) + y \geq e \\ & y \geq 0 \end{aligned} \tag{21}$$

where $\Lambda_i = K_i\upsilon$.

Similar to formulation (16), formulation (21) is biconvex but not strongly convex, this means that a global unique solution is not guaranteed and that formulation (21) converges to a local minimum, however as based in our experiments, the algorithm often converges in 6 or 7 iterations to a good quality solution with the desired degree of sparsity on the variables $a$ and $\upsilon$.

Note that when solving formulation (18), there are three different competing tasks for which the formulation has to find an acceptable balance: 1) to reduce the classification error, 2) to obtain an sparse vector $a$ to reduce the initial kernel dependency, and 3) to reduce the kernel basis dependency by obtaining an sparse $\upsilon$.

Furthermore, it is interesting to note that even when formulation (18) produces good sparse classifiers, we noted that performance can even be improved by solving an standard SVM formulation with the resulting optimal sparse kernel:

$$\hat{K}(x, \tilde{A_{tr}}) = \sum_{i=1}^{k} \tilde{a}_i K_i(x, \tilde{A_{tr}}') \tag{22}$$

that results from solving formulation (18). where $\tilde{A_{tr}} = \{y_l \in A_{tr} | u_l \neq 0\}$.

We are ready now to describe our proposed algorithm.

**Algorithm 1 Sparse Automatic kernel selection SVM (SAK-SVM)**

*Given $m$ data points in $R^n$ represented by the $m \times n$ matrix $A$ and vector $L$ of $\pm 1$ labels denoting the class of each row of $A$ , the parameters $\nu_1, \nu_2$ and an initial $a^0 \in \Re^k$, we generate a nonlinear sparse classifier similar to (17) as follows:*

*(0) Calculate $K_1, \ldots, K_k$ , the $k$ kernels on the kernel family, where for each $i$, $K_i = K_i(A, A')$.*

*For each iteration $i$ do:*

*(i) given an $a^{(i-1)}$ calculate the linear combination*

$$K = \sum_{j=1}^{k} a_j^{(i-1)} K_j$$

*(i) Solve subproblem (20) to obtain $(\upsilon^{(i)}, \gamma^{(i)})$.*

*(ii) Calculate $\Lambda_l = K_l \upsilon^{(i)}$ for $l = 1, \ldots, k$.*

*(iii) Solve subproblem (21) to obtain $a^i$.*

*Stop when a predefined maximum number of iterations is reached or when there is sufficiently little change of the objective function of problem (18) evaluated in successive iterations. Solve an standard SVM formulation with the learned optimal kernel function (22).*

Next we present numerical results that confirm the usefulness of our proposed approach.

## Experimental Results

As pointed out in subsection , the data consists of 320 cases for which we have two views (A4C and A2C). These two views show 12 of the 16 total segments. The training set (TR) consists on 224 cases and the final testing set (TE) consists on 96 cases. We performed two sets of experiments: one for each view (A4C and A2C). For each one of the views we compared three methods: A standard PSVM formulation (Fung & Mangasarian 2001), as a baseline; the AK-PSVM formulation (16) that is very similar to the one proposed in (Fung *et al.* 2004); and our proposed approach SAK-SVM. We report the area under the ROC (AUC) curve on the testing set as a measure of performance.

For each method we tuned the parameters involved (one for PSVM and AK-PSVM, two for SAK-SVM) by 5-fold cross validation. The number of folds was kept smaller that the usual 10 because the class distribution is skewed toward the normal cases. Only about 15% of the cases are abnormal. By performing 5-fold cross validation we ensure that each class is fairly represented in each fold. The parameters required for both methods were chosen to be in the set $\{2^{-7}, 2^{-2}, \ldots, 2^{0}, \ldots, 12^{7}\}$. To solve the resulting quadratic programming (QP) problems we used CPLEX 9.0 (ILO 2004).

Results including the AUC and the number of nonzero elements for both vectors $a$ and $\upsilon$ are reported in Table 1.

For the A4C view experiments, all the three methods performed similarly (not statistically significant). However, the kernel obtained by SAK-SVM only utilizes two of the 6 initial kernel functions (The L1 norm distance and the correlation distance) and it only needs to calculate those two similarities with respect to 16 of the 224 available training cases. Hence, the kernel learned by using SAK-SVM only depends on about $7\%$ of the training datapoints instead on depending on all the training datapoints (which is the case when using PSVM and AK-SVM).

Similarly, For the A2C view experiments, PSVM obtained a relative poor performance compared to the other two methods. AK-PSVM performed slightly better than SAK-SVM, but again SAK-SVM only utilizes the kernels based on the L1 norm distance and the correlation distance and it only needs to calculate those two similarities with respect to 33 of the training cases (around $15\%$ of the available training data).

Table 1: Testing set results including the AUC and the number of nonzero elements for both vectors $a$ and $\upsilon$

| View | PSVM<br>nnz($a$)<br>nnz($\upsilon$) | AK-PSVM<br>nnz($a$)<br>nnz($\upsilon$) | SAK-SVM<br>nnz($a$)<br>nnz($\upsilon$) |
|---|---|---|---|
| A4C | 77.2 %<br>6<br>224 | 80.0%<br>6<br>224 | 78.8 %<br>**2**<br>**16** |
| A2C | 74.3%<br>6<br>223 | 83.9%<br>6<br>223 | 80.8%<br>**2**<br>**33** |

## Conclusions

This paper presents a novel automatic technique to detect the heart wall motion abnormality. Given a sequence of endocardial contours extracted from LV ultrasound images, the contours moving through can be interpreted as a 3D surface. Therefore, the problem of detecting heart wall motion abnormalities is converted to the problem of classification of 3D surfaces. This 3D-surface-based formulation of this problem is novel, as far as we know. In contrast to other recent approach to solve this problem (Qazi *et al.* 2007), our method requires few or no prior knowledge about the problem and it only relies on the estimation of the movement of the walls extracted from echocardigrams. Based on six different well-known measurements, we defined several initial similarity functions between the surfaces. Since these similarity functions can be thought of as kernel functions, we proposed an algorithm to optimally combined these kernel functions to design a final kernel function that is optimal for the heart wall motion classification problem. Moreover, in contrast with other state-of-the-art methods our algorithm provide sparse solutions that lead to kernels that depend in fewer of the initial kernel functions and fewer basis functions. This is very important when creating a classifier that is going to be used in a real-time system, since it can reduced the testing computational time considerable. We compared the proposed approach to other state-of-the-art algorithms on a real dataset and the reported results show that our method is competitive performance-wise while achieving sparsity in the resulting kernels. Even when this paper is intended to show the performance on the heart wall motion abnormality application, we believe that our methodology is very general and that it can be applied to any classification problem involving 3D surfaces. We plan to perform further experiments to confirm this.

## References

Bach, F.; Lanckriet, J.; and Jordan, M. 2004. Fast kernel learning using sequential minimal optimization. Technical

Report CSD-04-1307, Division of Computer Science, University of California, Berkeley.

Bezdek, J., and Hathaway, R. 2002. Some notes on alternating optimization. In *Proceedings of the 2002 AFSS International Conference on Fuzzy Systems*, 288–300. Springer-Verlag.

Bradley, P. S., and Mangasarian, O. L. 1998. Feature selection via concave minimization and support vector machines. In Shavlik, J., ed., *Machine Learning Proceedings of the Fifteenth International Conference(ICML '98)*, 82–90. San Francisco, California: Morgan Kaufmann.

Catherine M. Otto, M. 2000. *Textbook of Clinical Echocardiography, 2nd edition*. Philadelphia, PA: W.B. Saunders Company. ISBN 0-7216-7669-3.

Cerqueira, M.; Weissman, N.; Dilsizian, V.; Jacobs, A.; Kaul, S.; Laskey, W.; Pennell, D.; Rumberger, J.; Ryan, T.; and Verani, M. 2002. Standardized myocardial segmentation and nomenclature for tomographic imaging of the heart. *American Heart Association Circulation* 105:539 – 542. URL: http://circ.ahajournals.org/cgi/content/full/105/4/539.

Comaniciu, D.; Zhou, X. S.; and Krishnan, S. 2004. Robust real-time tracking of myocardial border: An information fusion approach. *IEEE Trans. Medical Imaging* 23, NO. 7:849 – 860.

Comaniciu, D. 2003. Nonparametric information fusion for motion estimation. *Proc. IEEE Conf. Computer Vision and Pattern Recognition* I:59 – 66.

Cover, T., and Thomas, J. 1991. *Elements of Information Theory*. Wiley Interscience.

Dorai, C., and Jain, A. 1997. Cosmos-a representation scheme for 3d free-form objects. *IEEE Trans. Pattern Analysis and Machine Intelligence* 19(10):1115–1130.

Flynn, P., and Jain, A. 1989. On reliable curvature estimation. *Proc. IEEE Conf. Computer Vision and Pattern Recognition* 110–116.

Fung, G., and Mangasarian, O. L. 2001. Proximal support vector machine classifiers. In Provost, F., and Srikant, R., eds., *Proceedings KDD-2001: Knowledge Discovery and Data Mining, August 26-29, 2001, San Francisco, CA*, 77–86. New York: Asscociation for Computing Machinery. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/01-02.ps.

Fung, G.; Dundar, M.; Bi, J.; and Rao, B. 2004. A fast iterative algorithm for fisher discriminant using heterogeneous kernels. In *ICML '04 Proceedings*. ACM Press.

Georgescu, B.; Zhou, X. S.; Comaniciu, D.; and Gupta, A. Database-guided segmentation of anatomical structures with complex appearance. *IEEE Conf. Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, 2005*.

Golub, G. H., and Van Loan, C. F. 1996. *Matrix Computations*. Baltimore, Maryland: The John Hopkins University Press, 3rd edition.

Guyon, I., and Elisseeff, A. 2003. An introduction to variable and feature selection. *JMLR* 3:1157–1182.

Hamers, B.; Suykens, J.; Leemans, V.; and Moor, B. D. 2003. Ensemble learning of coupled parmeterised kernel models. In *International Conference on Neural Information Processing*, 130–133.

Hoffmann, R.; Marwick, T.; Poldermans, D.; Lethen, H.; Ciani, R.; van der Meer, P.; Tries, H.-P.; Gianfagna, P.; Fioretti, P.; Bax, J.; Katz, M.; Erbel, R.; and Hanrath, P. 2002. Refinements in stress echocardiographic techniques improve inter-institutional agreement in interpretation of dobutamine stress echocardiograms. *European Heart Journal* 23, issue 10:821 – 829. doi:10.1053/euhj.2001.2968, available online at http://www.idealibrary.com.

ILOG CPLEX Division, 889 Alder Avenue, Incline Village, Nevada. 2004. *CPLEX Optimizer*. `http://www.cplex.com/`.

Kim, S.-J.; Magnani, A.; and Boyd, S. 2006. Optimal kernel selection in kernel fisher discriminant analysis. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, 465–472. New York, NY, USA: ACM Press.

Koenderink, J. J., and Doorn, A. V. 1992. Surface shape and curvature scales. *Image Vision Computing* 10(8):557–565.

Lanckriet, G.; Cristianini, N.; Bartlett, P.; Ghaoui, L. E.; and Jordan, M. 2003. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research* 5:27–72.

Lee, P. M. 2004. *Bayesian Statistics*. New York, NY: Oxford University press, 3rd edition.

Qazi, M.; Fung, G.; Krishnan, S.; Rosales, R.; Steck, H.; Rao, B.; Poldermans, D. D.; and Chandrasekaran, D. 2007. Automated heartwall motion abnormality detection from ultrasound images using bayesian networks. In *IJCAI 2007 Proceedings*. URL: http://www.ijcai.org/papers07/contents.php.

Schiele, B., and Crowley, J. 2000. Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision* 36(1):31–50.

Therrien, C. 1989. *Decision, Estimation, and Classification*. John Wiley and Sons.

World Health Organization. 2004. The atlas of global heart disease and stroke. URL:http://www.who.int/ cardiovascular_diseases/ resources/atlas/.