Greedy Algorithm for Error Correction in Automatically Produced Boundaries from Low Contrast Ventriculograms

Jasjit S. Suri,† Robert M. Haralick,† Florence H. Sheehan ‡

†Intelligent Systems Laboratory
Department of Electrical Engineering
University of Washington
Seattle, WA 98195

‡Cardiovascular Research & Training Center
University of Washington Medical Center
University of Washington
Seattle, WA 98195

Poor contrast in the apex zone and non-homogeneous mixing of the dye with the blood in the left ventricle causes the left ventricle *pixel-based* classifiers operating on ventriculograms [1] to yield boundaries which are not close to *ground truth* boundaries as delineated by the cardiologist. They have a mean boundary error of 6.4 mm and an error of appoximately 12.5 mm in the apex zone. These errors have a systematic positional and orientational bias, the boundary being under-estimated in the apex zone.

This paper discusses two calibration methods: the *Identical Coefficient* and the *Independent Coefficient* to remove these systematic biases. From these methods, we constitute a fused algorithm which reduces the boundary error compared to either of the calibration methods. The algorithm, in a Greedy way, computes *which and how many* vertices of the left ventricle boundary can be taken from the computed boundary of each method in order to best improve the performance.

The corrected boundaries have a mean error of less than 3.5 millimeters with a standard deviation of 3.4 mm over the approximately 6×10^4 vertices in the data set of 291 studies. Our method reduces the mean boundary error by 2.9 millimeters over the boundary produced by the classifier. We also show that the calibration algorithm performs better in the apex zone where the dye is unable to propagate. For End-Diastole, the algorithm reduces the error in the apex zone by 8.5 millimeters over the *pixel-based* classifier boundaries.

Greedy Algorithm for Error Correction in Automatically Produced Boundaries from Low Contrast Ventriculograms

Abstract

Poor contrast in the apex zone and non-homogeneous mixing of the dye with the blood in the left ventricle causes the left ventricle *pixel-based* classifiers operating on ventriculograms [1] to yield boundaries which are not close to *ground truth* boundaries as delineated by the cardiologist. They have a mean boundary error of 6.4 mm and an error of appoximately 12.5 mm in the apex zone. These errors have a systematic positional and orientational bias, the boundary being under-estimated in the apex zone.

This paper discusses two calibration methods: the *Identical Coefficient* and the *Independent Coefficient* to remove these systematic biases. From these methods, we constitute a fused algorithm which reduces the boundary error compared to either of the calibration methods. The algorithm, in a Greedy way, computes *which and how many* vertices of the left ventricle boundary can be taken from the computed boundary of each method in order to best improve the performance.

The corrected boundaries have a mean error of less than 3.5 millimeters with a standard deviation of 3.4 mm over the approximately 6×10^4 vertices in the data set of 291 studies. Our method reduces the mean boundary error by 2.9 millimeters over the boundary produced by the classifier. We also show that the calibration algorithm performs better in the apex zone where the dye is unable to propagate. For End-Diastole, the algorithm reduces the error in the apex zone by 8.5 millimeters over the *pixel-based* classifier boundaries.

Key Words: Motion, Left Ventricle, Low Contrast, Boundaries, Training, Calibration, Optimization, Polyline.

1 Introduction

Accurate and automatic boundary estimation of the left ventricle (LV) and its quantitative analysis is important for diagnostic purposes. One of the popular methods for studying cardiac disorders is by catheterization. In this procedure, the contrast medium (a Barium compound dye) is injected into the heart where it mixes with the blood during the heart cycle. Because this mixing takes place in a non-homogeneous manner and because of the papillary muscle structure of the heart, very little dye reaches the *apex zone* (See Figs. 1, 2, and 3). Consequently, the gray scale contrast of the *apex zone* is poor. In addition left ventriculograms (LVG) have a high level of noise due to the scattering by tissue volumes which are not related to the left ventricle as explained by Tanaka, Sezan and Nakajima [6].

A *pixel-based* classification procedure [1] followed by motion morphology was developed at the Intelligent Systems Laboratory (ISL) in conjunction with the Cardiovascular Research and Training Center (CVRTC), University of Washington, Seattle, to estimate the borders of the left ventricle. These algorithms however do not yield boundaries close to the hand-drawn boundaries as delineated by the cardiologist. Of the many reasons for the failure of *pixel-based* image processing algorithms, the foremost is the poor contrast in the *apex zone* that causes the image processing boundaries to fall short of ground truth boundaries or to be *under-estimated* [1] (see Fig. 1 and Fig. 2). Furthermore, irregularities of the *inferior* walls, variations in the left ventricle shape, size, orientation, and heart rate cause the boundaries from the *pixel-based* classifier output to push outwards or to be *over-estimated* (see Fig. 1).

Due to all of the above complications, a reliable left ventricle boundary detection algorithm must make use of as much knowledge of the left ventricle shape, size, position and orientation as possible. This knowledge can be learned from a population of left ventricle boundaries manually traced by the cardiologist or a trained technician. Our database population consisted of 291 patient studies. 135 studies had acute myocardial infarction with a top 50% in quality.

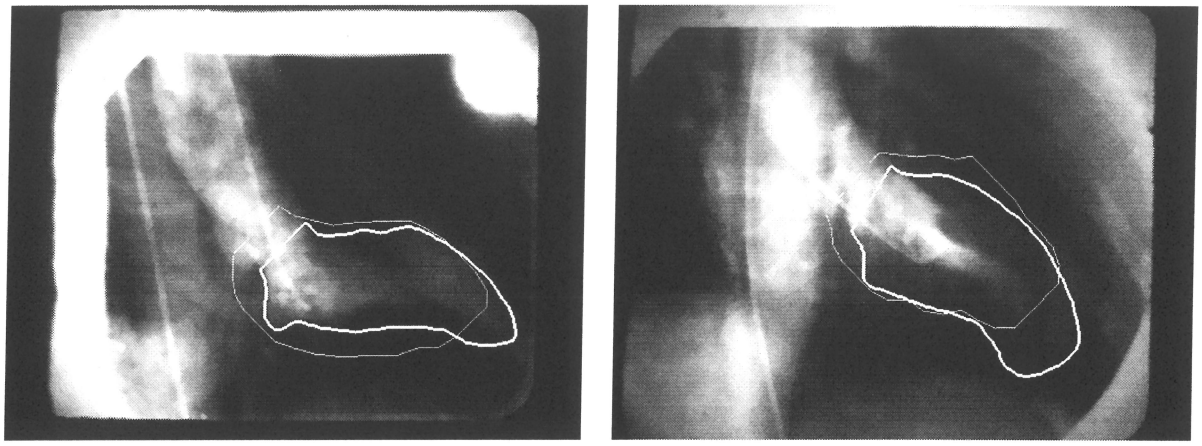


Figure 1: **Left**: End-Systole (ES) frame showing very little dye in the apex zone of the LV. ES frame is that frame of the heart cycle when the LV has fully contracted. It is named so because it is the last frame of the Systole (contraction) cycle. **Right**: End-Systole frame showing with little dye and interference by ribs with the LV. Thick lines represent the border drawn by the Cardiologist. Thin lines are the border computed by the *pixel-based* classifier or boundaries produced by image processing algorithms. Background consists of gray scale ventriculograms (LVG) of size 384 $\times$ 512.

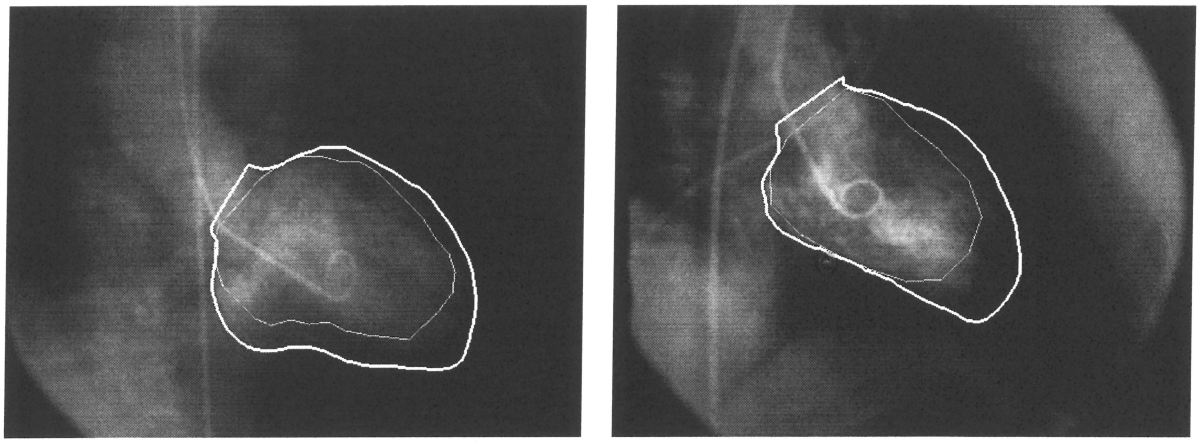


Figure 2: **Left**: End-Diastole (ED) frame showing very little dye in the apex zone of the LV. ED frame is that frame of the heart cycle when the LV has fully expanded. It is named so because it is the last frame of the Diastole (expansion) cycle. **Right**: End-Diastole frame showing the *pixel-based* classifier boundaries are *under-estimated* in anterior, inferior walls and apex zone. Thick lines represent the border drawn by the Cardiologist. Background consists of gray scale left ventriculograms (LVG) of size 384 $\times$ 512.

35 patients were studied during acute infarction, who then underwent follow-up studies over the course of one year. 27 studies from Japan were normal and had a diagnostic cardiac catheterization, but were found to have normal anatomy. The Japanese patient LV images represented the top 50% in quality; the remainder of the 94 studies were from Catheterization Laboratory, at the University of Washington, Seattle, Washington. These studies represented a top 30% in quality. Using this knowledge of traced boundaries from this population and the information about the *pixel-based* classifier boundaries, it was possible to construct an *off-line* training system which can train two sets of boundaries: the boundaries produced by an image processing algorithm and the boundaries traced by the physicians. The trained system can then be applied *on-line* to new patient boundaries generated by an image processing algorithm.

In this paper we show that the aforementioned training procedure is a very promising technique for measuring the accuracy of left ventricle borders of the left ventricle, particularly in the *apex zone* and *walls* where the dye has not mixed well with the blood. We show that when the trained system is applied to new left ventricle test patient boundaries generated from the *pixel-based* classifier, the system reduces the systematic boundary biases and provides automatic left ventricle boundaries that are within 3.5 mm of the *physician traced* boundaries. Our procedure can be thought of as a *calibration* procedure: calibrating the initial *pixel-based* boundaries closer to the *ground truth* boundaries by removing any systematic bias in shape, position, and orientation.

Calibration in general is very popular in vision, image processing, and statistical medical imaging. Wunderlich and Fischer [2] developed an *isocenter calibration* technique by which they transformed the cardiac structures in digitized biplane angiograms to absolute dimensions calculating their radiological magnification and video transformation. Most imaging systems have an inherent magnification which requires a calibration to calculate the number of pixels per millimeter. Our calibration is of a different kind, one where we remove systematic shape, positional, and orientational errors from boundaries produced by an image processing

algorithm operating on low contrast heart images.

The boundary calibration system employs two different techniques: the *Identical Coefficient Method* and the *Independent Coefficient Method*, named after the way each technique estimates the parameters. Each method produces estimates of the vertices of the polygon bounding the left ventricle and is independent for each frame of the heart cycle. We then *fuse* these vertex sets to form a final boundary. We select in a *Greedy* manner that fixed subset of estimated *vertex positions* from each method which, when fused together, most minimizes the resulting errors between the final estimated polygon boundary and the *physician traced* left ventricle boundary.

We used a *cross-validation* procedure for estimating the error of the *calibration system.* The procedure took a database of N patient studies and partitioned this database into K equal sized subsets. For all the K *choose* L combinations, we trained the system using L subsets, and applied the estimated transformation on the remaining $(K - L)$ subsets. The mean error of the transformed boundary was then computed from these $(K - L)$ subsets coming from all K choose L combinations. Our experiments consisted of varying the calibration parameters: N, K, L, P. We chose six different sets of K values (corresponding to each protocol) for training the system.

Because of the small number of available patient studies in our database, (N=291) and the large number of parameters (about 200 times N) in the transformation, there was a danger of *memorization* rather than *generalization* in the estimation of the *transformation parameters.* Therefore, it was essential that the number of vertices (P in the left ventricle polygon) be carefully chosen. As P decreased, the generalization improved but the representation of the *true LV shape* got worsend, thereby causing higher error with respect to the *ground truth.* As P increased, generalization was lost but representation of the *true LA shape* improved. With the other parameters K, L and N fixed, there was *optimal* number of boundary vertices P^* balancing the *representation error* with the *memorization error.* Our protocol found the

optimal number.

We measured the error between an automatically constructed boundary produced by an image processing algorithm and a *ground truth boundary* as traced by physician using a *polyline distance metric.* The polyline distance metric was symmetrically defined as the average distance between a vertex of one polygon to the boundary of the other polygon. Using this distance *metric*, we analyzed the left ventricle estimated boundary data in seven different ways: The main four are: *cumulative distribution* of the average of the End-Diastole (ED) and End-Systole (ES) errors $(\frac{ED+ES}{2})$, *error by frames* of the systolic heart cycle, *error per vertex* and *error per arc length* along the left ventricle contour, starting from the anterior aspect of the aortic valve. We showed that the calibration improves the estimation of the boundary in the *apex zone* where there is *low contrast* in ventriculograms. For the End-Diastole frame, there is an improvement of about 8.5 millimeters (approximately 20 pixels) and for the End-Systole frame about 3 millimeters (approximately 7.5 pixels) over the initial *image processing* algorithm boundaries. The Greedy algorithm that fused the Identical Coefficient Method and Independent Coefficient Method boundaries yielded the best results, thereby further reducing the error by 0.3 millimeters in the *apex zone* . We showed 81% of the patient boundaries had a mean $(\frac{ED+ES}{2})$ less than 4.0 millimeters using the Greedy calibration technique.

2 Left Ventricle Related Literature

This section discusses how the previous research work is related to current research and significant reviews which use automatic algorithms for LV boundary extraction. Segmentation of medical organs for medical diagnosis began when the imaging systems became available for imaging of the body organs [3], [4]. In medical applications, the shape of the organs vary considerably between patient studies and through time. The foundation of most of the boundary segmentation techniques have been histogram-based. Tanaka, *U.S. Patent number 4,952,805*, developed a histogram-based division method followed by statistical shape analysis. Shimura, *U.S. Patent number 4,028,079*, developed an edge detection technique along the radial lines

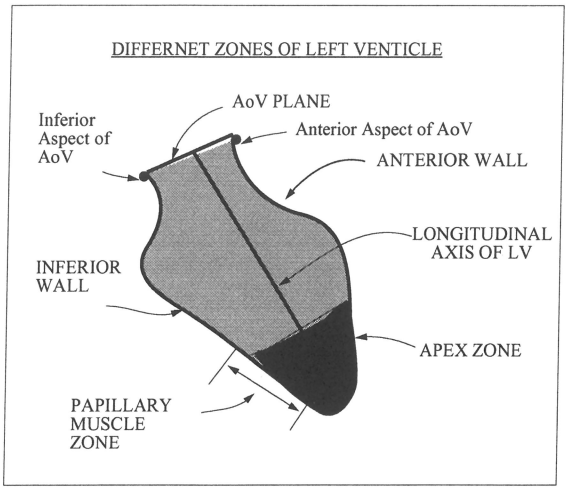


Figure 3: Labelling of the Left Ventricle (ED frame): The anterior and inferior side walls, the apex zone, the AoV plane (joining AAV and IAV). Bold line joining mid of AoV and the apex point is the longitudinal axis of the left ventricle, shown, about one-third region of the left ventricle region is the *apex zone.*

from the center of the image followed by thresholding. All these methods are either semi-automatic or are not robust enough to produce reliable boundaries.

Cootes [5] attempted using an active shape model to infer the position of boundary parts where there was missing data (top of the ventricle); he used the knowledge of the expected shape combined with information from the areas of the image where good evidence of wall could be found. Cootes had used Least Squares method to estimate shape parameters. For the final shape estimation, Cootes then used a weighted interactive algorithm where weights were proportional to the standard deviation of the shape parameter over the training set. Cootes used the Mahalanobis distance for performance.

van Bree attempted to find the LV borders using a combination of probability surfaces and dynamic programming. van Bree's algorithm consists of three major steps. The first step is building the probability surfaces from a database of hand-drawn boundaries after the left ventricle boundaries have been corrected for translation and rotation. The second step consists of generation of extraction lines (also called search lines) for the design of the search matrix. These search lines are the lines which originate from the interior of the left ventricle

and extend radially outward to the outer region (background) of the left ventricle. The third step consists of smoothing the search matrix followed by dynamic programming to search for the optimal path of the left ventricle border.

Lee [1] used a *pixel-based* bayesian approach for the left ventricle boundary extraction where he looked at the gray scale value of the location throughout the cardiac cycle as a vector: Lee assumed that the distribution was bi-variate normal. For each observed vector, one class is assigned according to the ground truth, which is available by filling the LV region surrounded by the ground truth LV boundary. Ground truth of a pixel through F frames generates 2^F possible classes but, far less than 2^F classes are actually required in order to represent all existent classes. Due to the different heart rates (i.e. variable number of frames in the cardiac cycle for different studies) Lee allowed different dimension of the observation vector. Lee simultaneously segmented all the frames from End-Diastole through End-Systole to the next End-Diastole instead of segmenting frame by frame separately, assuming that each pixel was statistically independent in the same frame. Lee later modified his assumption by considering the nearest neighbor. For performance and error computation of the detected LV boundary, Lee compares classifier boundary with manually traced boundary using the Hausdorf distance measure. Because the gray scale contrast in the region of apex zone is very low and there is interference of diaphragm with inferior walls, the classifier produces boundaries which are not close to ground truth boundaries; furthermore, the boundaries produced by the classifier do not reflect true LV shape. In this paper, we discuss three sets of algorithms which takes these classifier boundaries and transform them closer to the boundaries as delineated by cardiologist.

The apex is harder to identify in gray scale LVG's because of the inadequate dye propagation; therefore, Suri, Haralick and Sheehan are developing an integrated directional derivative approach to determine the directional (angle) patterns in the ventriculograms which are very useful for apex estimation in the LVG.

Quantification of LV wall motion particularly interested [7], [8] researchers since the quan-

tification relates directly to cardiac disorders. These methods are computer based and differ in geometrical approach for quantification of wall motion. Putting markers on the mid-wall for quantification of wall dynamics was first suggested by Gibson [34]. Gibson's paper discusses five methods for wall quantification. Sheehan developed the Centerline Method for measurement of wall motion [7]. Sheehan measures the motion along 100 chords constructed perpendicular to a Centerline drawn midway between the End-Diastole and End-Systole LV contours. Suri, Haralick and Sheehan developed the Polyline Distance Method for estimating mean distances between two LV boundary shapes [9]. We discuss our method in the performance section.

Regression-based approaches offer an alternate way to fit noisy boundary data. For example, some techniques which can be applied are: the median Least Squares approach developed by Kim [22] for fitting the noisy data. Hampel and Holland [23], [24], [25] developed an Iterative Reweighted Least Square (IRLS) for robust estimation of the outliers. Suri, Haralick and Sheehan developed an *Iterative Reweighted Least Square* scheme for computing the *robust apex coordinates* of the left ventricle and then computing the serial number of the apex point starting from the anterior aspect of aortic valve. Performance of the robust procedure can be determined by estimating its efficiency which is defined as the *ratio* of root mean square error computed using *inlier data* alone in the the regression to the root mean square error using the *combined* inlier as well as outlier data in the regression.

3 Two Coefficient Methods: IdCM & InCM

This section gives the mathematical statements of the two calibration methods used by the Greedy algorithm for estimation of the left ventricle boundaries. Ground truth boundaries refer to the hand delineated boundaries drawn by the cardiologist or the trained technician. Classifier boundaries (also called as raw boundaries) refer to the boundaries produced by the initial pixel based automated classification procedure. In the *Identical Coefficient Method*, each vertex is associated with a set of coefficients. The calibrated x-coordinate for that vertex is computed as the linear combination of raw x-coordinates of the LV boundary using the

coefficients associated with that vertex. The calibrated y-coordinate of that vertex is similarly computed as the *same* linear combination of raw y-coordinates of the LV boundary. In the *Independent Coefficient Method*, the calibrated x-coordinate is computed as the linear combination of raw x- *and* raw y-coordinates of the LV boundary, using the coefficients associated with that vertex. The calibrated y-coordinate of that vertex is computed with a *different* linear combination of raw x- *and* y-coordinates. The problem of calibration then reduces to a problem of determining the coefficients of the linear combination which can be accomplished by solving a regression problem. The initial (x, y) coordinates of the left ventricle are converted from pixels to millimeter using *magnification correction factor*. This factor is computed by keeping a grid of lead wire of known millimeter size or a kugel of known diameter (in mm) over the ventriculograms. The kugel is approximately the same size as the left ventricle (approximately 70 millimeters). These input raw and ground truth boundaries are initially in a 100 vertices polygon format with unit dimensions in millimeters, we therefore *resample and interpolate* each of these polygons into *equally spaced* vertices before it undergoes the calibration procedure discussed below.

3.1 Identical Coefficient Method (IdCM) for any frame

Let $g_n^{'}$ and $h_n^{'}$ be the row vectors of x-coordinates and y-coordinates respectively for the ground truth boundaries for patient n. Let $r_n^{'}$ and $s_n^{'}$ be the row vectors of x-coordinates and y-coordinates respectively for the classifier boundary for any patient n, where $n = 1, ..., N$. For the calibrated boundary estimation in left ventriculograms using the *Identical Coefficient Method* , we are

- **Given**: Corresponding pairs of ground truth boundaries $\mathbf{R}$ $[2N \times P]$, and the classifier boundaries $\mathbf{Q}$ $[2N \times (P+3)]$, respectively:

$$\mathbf{R} = \begin{pmatrix} g_1^{'} \\ h_1^{'} \\ \cdots \\ g_N^{'} \\ h_N^{'} \end{pmatrix} \qquad \mathbf{Q} = \begin{pmatrix} r_1^{'} \underbrace{1\, u_{11}\, u_{21}} \\ s_1^{'} \underbrace{1\, v_{11}\, v_{21}} \\ \cdots \\ r_N^{'} \underbrace{1\, u_{1N}\, u_{2N}} \\ s_N^{'} \underbrace{1\, v_{1N}\, v_{2N}} \end{pmatrix}$$

where, (u_{11}, v_{11}), (u_{1N}, v_{1N}) and (u_{21}, v_{21}), (u_{2N}, v_{2N}) are the coordinates for the anterior aspect and inferior aspect of the AoV plane of the left ventricle (see figure 1) from ground truth boundary.

- Let $\mathbf{A}$ $[(P+3) \times P]$ be the unknown regression coefficients matrix.
- The problem is to estimate the coefficient matrix $\mathbf{A}$, to minimize $\| \mathbf{R} - \mathbf{Q}\,\mathbf{A} \|^2$. Then for any classifier boundary matrix $\mathbf{Q}$, the calibrated vertices of the boundary are given by $\mathbf{Q}\hat{\mathbf{A}}$, where $\hat{\mathbf{A}}$ is the estimated coefficients.

Note that from the problem formulation, the coefficients that multiply $g_n^{'}$ also multiply $h_n^{'}$, hence the name *Identical Coefficient Method.* Also note that the new x-coordinates for the n^{th} boundary depend only on the old x-coordinates from the n^{th} boundary, and the new y-coordinates from the n^{th} boundary depend only on the old y-coordinates from the n^{th} boundary.

3.2 Independent Coefficient Method (InCM) for any frame

As before, let $g_n^{'}$ and $h_n^{'}$ be the row vectors of x- and y-coordinates for any patient n. Let $r_n^{'}$ and $s_n^{'}$ be the row vectors of x- and y-coordinates of the classifier boundary. For the calibrated boundary estimation in ventriculograms using the *Independent Coefficient Method*, we are:

- **Given**: Corresponding ground truth boundaries $\mathbf{R}$ $[N \times 2P]$, classifier perturbed boundaries $\mathbf{Q}$ $[N \times (2P+5)]$ respectively:

$$\mathbf{R} = \begin{pmatrix} g_1^{'}\ h_1^{'} \\ \cdots \\ \cdots \\ g_N^{'}\ h_N^{'} \end{pmatrix} \qquad \mathbf{Q} = \begin{pmatrix} r_1^{'}\ s_1^{'}\ \underbrace{1\ u_{11}\ v_{11}\ u_{21}\ v_{21}} \\ \cdots \\ \cdots \\ r_N^{'}\ s_N^{'}\ \underbrace{1\ u_{1N}\ v_{1N}\ u_{2N}\ v_{2N}} \end{pmatrix}$$

where, (u_{11}, v_{11}), (u_{1N}, v_{1N}) and (u_{21}, v_{21}), (u_{2N}, v_{2N}) are the coordinates of the anterior aspect and inferior aspect of the AoV plane of the left ventricle (see figure 1) from ground truth boundary.

- Let $\mathbf{A}$ $[(2P+5) \times 2P]$ be unknown regression coefficient matrix.

- The problem is to estimate the coefficient matrix $\mathbf{A}$, to minimize $\| \mathbf{R} - \mathbf{Q}\,\mathbf{A} \|^2$. Then for any classifier boundary matrix $\mathbf{Q}$, the calibrated vertices of the boundary are given by $\mathbf{Q}\hat{\mathbf{A}}$, where $\hat{\mathbf{A}}$ is the estimated coefficients.

Note that the new (x, y)-coordinates of the vertices of each boundary is a *different* linear combination of the old (x, y)-coordinates for the polygon, hence the name *Independent Coefficient Method.* The above two methods are different in the way the calibration model is set up. The classifier boundary matrix $\mathbf{Q}$ in IdCM is of size $\underbrace{2N \times (P+3)}$ while in InCM is of size $\underbrace{N \times (2P+5)}$. For IdCM, the number of coefficients estimated in the $\hat{\mathbf{A}}$ matrix is $\underbrace{(P+3) \times P}$. For InCM, the number of coefficients estimated is $\underbrace{(2P+5) \times 2\,P}$. Thus the Independent Coefficient Method requires around 4 times the number of coefficients of the Identical Coefficient Method to be estimated, and this difference represents a significant factor in the ability of the technique to *generalize* rather than *memorize* for our data size.

4 Greedy Algorithm for LV Calibration by Vertex

Section 4.1 discusses the boundary data preparation before it is fed to the boundary calibrator. Section 4.2 discusses the estimation of the training coefficients and the estimated boundaries for both boundary calibrations. Section 4.3 discusses the Greedy algorithm and vertex pooling process.

4.1 Data Preparation & Correspondence

Figure 4 shows the overall boundary calibration system, where the heart of the system is the *IdCM or InCM calibrator.* The input to the calibrator is the LV boundary data that consists of N studies, F frames and $P_1 = 100$ vertices. The input boundary data is resampled and interpolated so that each polygon has equally spaced P_2 vertices. Thus, every vertex number of ground truth boundary corresponds to the same vertex of the classifier boundary. This resampling and linear interpolation is done in an automatic way, where the user needs to only specify the number of boundary vertices to be sampled and which frames of the cardiac cycle

to be sampled. In our case, we choose two frames for sampling, the End-Diastole and the End-Systole, and the number of sampled boundaries vertices are P_2. Note that the AoV plane is known. This means the (x, y)-coordinates for the anterior aspect of aortic valve and inferior aspect of aortic valve is known.

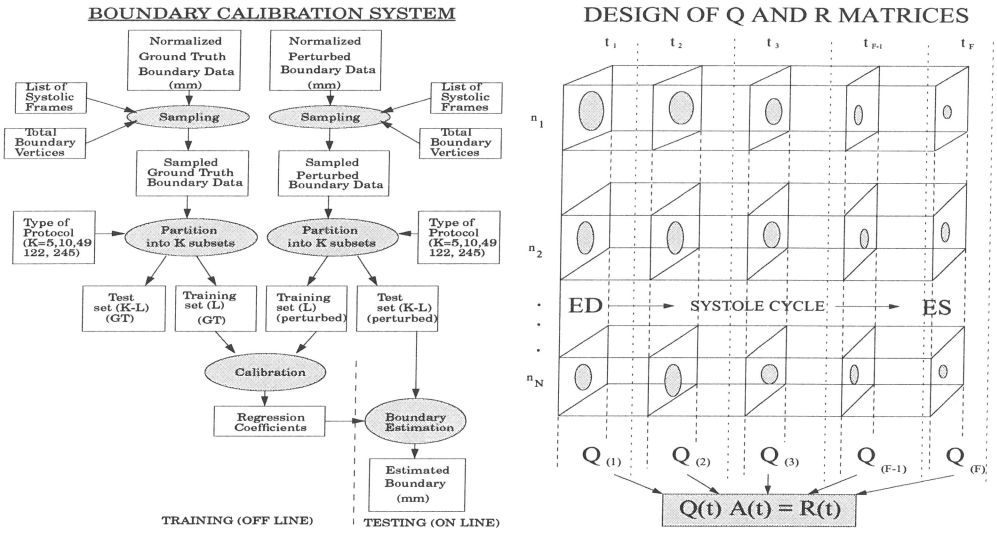


Figure 4: Object process diagram for the IdCM and InCM calibrations for *any frame* of the heart cycle. We sample and interpolate the input normalized data, followed by the partitioning the data into K subsets, L used for training and $K - L$ used for the testing set. Magnification conversion factors are used to convert pixels to millimeter where 1 pixel = 0.39 mm. **Right**: figure showing design of system matrices using the database for each frame independently.

4.2 IdCM and InCM Calibrations: Estimation Process

In either case, the regression model is to find the off-line coefficient matrix $\mathbf{A}(t)$ to minimize

$$\epsilon^2_{ifr} \quad = \quad \| \, \mathbf{R}(t) - \mathbf{Q}(t) \, \hat{\mathbf{A}}(t) \, \|^2 \tag{1}$$

Generalizing for any frame t of the systolic cycle, minimizing $\hat{\mathbf{A}}$ is given by the normal equation:

$$\hat{\mathbf{A}}_{tr} = \underbrace{(\, \mathbf{Q}^T \, \mathbf{Q} \,)^{-1} \, \mathbf{Q}^T \, \mathbf{R}} \tag{2}$$

The above equation is solved using singular value decomposition (SVD) [30]. Given the test set $(\mathbf{Q}_{te})$ or training set $(\mathbf{Q}_{tr})$, we can estimate the calibrated boundary as:

$$\underbrace{\hat{\mathbf{R}}_{te} = \mathbf{Q}_{te} \, \hat{\mathbf{A}}_{tr}}, \quad \& \quad \underbrace{\hat{\mathbf{R}}_{tr} = \mathbf{Q}_{tr} \, \hat{\mathbf{A}}_{tr}} \tag{3}$$

Thus the estimated matrices for the IdCM and InCM test sets are:

$$\underbrace{\hat{\mathbf{R}}_{id} = \mathbf{Q}_{te}\,\hat{\mathbf{A}}_{id}}, \quad \& \quad \underbrace{\hat{\mathbf{R}}_{in} = \mathbf{Q}_{te}\,\hat{\mathbf{A}}_{in}} \tag{4}$$

Note, if P_2 are the sampled vertices, then $\hat{\mathbf{R}}_{id}$ is of dimension $2N \times P_2$ and $\hat{\mathbf{R}}_{in}$ is of dimension $N \times 2P_2$.

4.3 Best InCM index selection: Greedy Algorithm

The Greedy algorithm is shown in the object process diagram 5. We select a fixed subset of estimated vertex positions from IdCM and InCM technique which when fused together, minimizes the resulting error between the final estimated polygon boundary and the physician traced LV boundary. Let S, S_{id} and S_{in} be three sets consisting of *all* vertices, IdCM pool vertices and InCM pool vertices respectively. Let $\hat{\mathbf{R}}_{id}$ $(2N \times P_1)$ and $\hat{\mathbf{R}}_{in}$ $(N \times 2P_2)$ be the estimated boundary matrices from IdCM and InCM techniques with P_2 samples vertices. Let $\mathbf{R}_{gt}$ $(N \times 2P_2)$ and $\mathbf{R}$ $(N \times 2P_2)$ be the matrices consisting of (x, y)-coordinates from original ground truth with P_1=100 and sampled P_2 vertices respectively. Initially all the vertices are considered in the IdCM pool and the error is computed. Denote its error by ϵ_{id}. Now we select that vertex from IdCM pool which when fused with InCM pool vertices yields an estimated boundary error lower than ϵ_{id}. This procedure is repeated until there is no further improvement. If ϵ be the error at any time in the Greedy *do-while loop*, and $\mathcal{F}$ be the set consisting of all the frames, then the Greedy *do-while loop* for any frame t in the set $\mathcal{F}$, consists of the following steps.

Greedy Boundary Calibration()

For each $t \in \mathcal{F}$

S_{id}=S; S_{in}=ϕ, ϵ=0 greedyCounter=0

While $(\epsilon \leq \epsilon_{id})$ **do**

greedyCounter++

For each $i \in S_{id}$, /* total vertices are P_{id} */

$S_{id} = S_{id} - \{i\}$; $S_{in} = S_{in} \bigcup \{i\}$

Combine IdCM ($\hat{\mathbf{R}}_{id}$) and InCM ($\hat{\mathbf{R}}_{in}$) Using S_{id} and S_{in}

$\hat{\mathbf{R}}_{com}$ = Combine($\mathbf{R}_{id}, \mathbf{R}_{in}, N, P_2, S_{id}, S_{in}$, greedyCounter)

Performance Evaluation using Original GT: Error for index i

ϵ_i = PolyPerformance($\hat{\mathbf{R}}_{com}, \mathbf{R}_{gt}, N, P_1, P_2$)

end /* end of the for loop */

ArgMin Computation: Minimum error and best vertex j selection

(ϵ_{min}, j) = ArgMin($\epsilon[i]$, $P_{id} - greedyCounter$)

if ($\epsilon_{min} < \epsilon$) **then** $S_{id} = S_{id} - \{j\}$; $S_{in} = S_{in} \bigcup \{j\}$ **else** break; **end**

end /* end of the while loop */

end /* end of all the frames of systolic heart cycle */

Note there are three steps in the Greedy algorithm. First, fusion of IdCM and InCM boundary data, second, polyline performance to compute the errors and third, the vertex selection. The input to the fusion process are 2 sets of boundaries $\hat{\mathbf{R}}_{id}$ and $\hat{\mathbf{R}}_{in}$ which need to be fused. The fusion is done by selection of that vertex (or column) of InCM boundary which contributed to a reduction in error. The Polyperformance() function takes 2 sets of boundaries: the fused boundary $\hat{\mathbf{R}}_{com}$ and the original ground truth $\mathbf{R}_{gt}$ consisting of P_1=100 vertices and computes the mean error (as discussed in section 5). The third function is the argmin() function or vertex selection function that takes the error associated with P_2 vertices and finds that vertex number from InCM LV boundary that yielded the least error in each Greedy cycle. One such Greedy cycle is shown in fig. (5),right. Consider 2 baskets say b_1 and b_2 containing same number of balls P_2 but of different color, say white and black. These balls can be imagined to represent the vertices of the LV boundary. The first cycle consists of searching of that black ball from basket b_2 which when combined with P_2-1 white balls in basket b_1 will yield lower error when no ball was transferred. Finally 1 such ball is transferred from basket b_2 to basket b_1 and cycle is repeated till no more balls are found which improves the performance of the system.

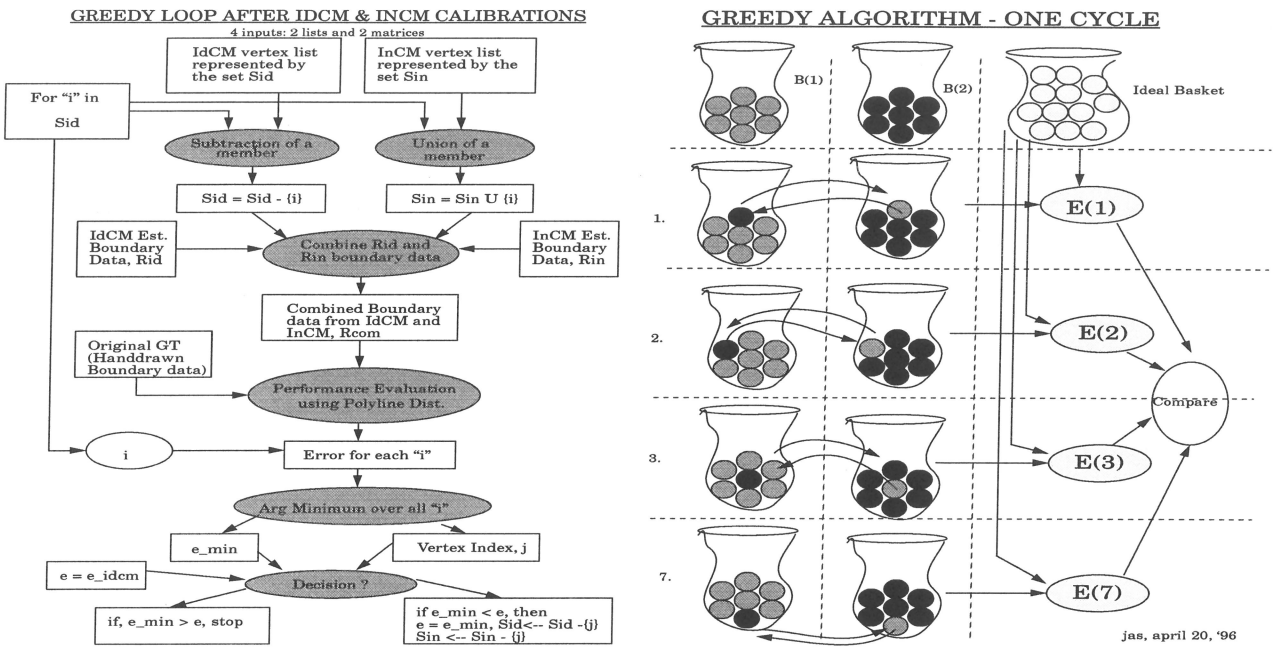


Figure 5: **Left**: *For loop* for combining the IdCM and InCM calibrations used in the Greedy do-while loop. The data structure of the do-whlie loop is implemented in the form of lists where the best vertex is computed and empted from the IdCM pool and unioned in the InCM pool. The do-while loop is implemented independently for ED and ES frames. **Right**: One Greedy cycle where best vertex (ball) is selected out of basket b_2.

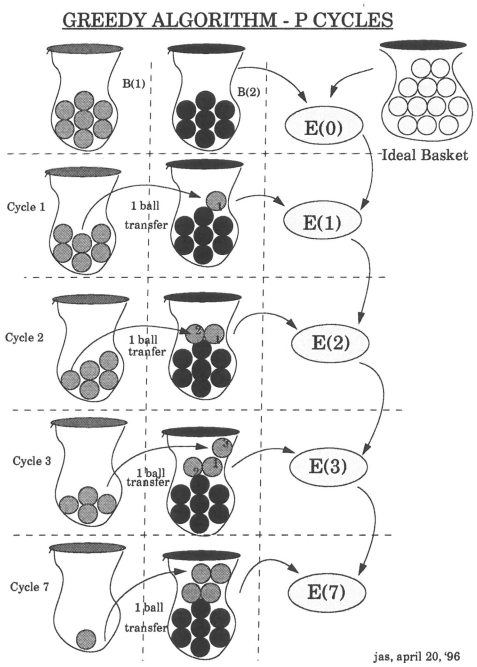


Figure 6: The figure shows basket number b_2 filling up after each Greedy cycle.

5 Polyline Distance Measure and Performance Terms

The performance of the calibration algorithms is evaluated by computing the boundary error on the test data set ($\mathbf{Q}_{te}$). IdCM and InCM calibration algorithms compute the error in a suboptimal way; i.e., for each vertex there is error between the ground truth LV boundary to the corresponding vertex on the estimated LV boundary. But the cardiologists are interested in finding the closest distance of a vertex on one polygon boundary to another LV polygon. Sheehan [7] developed a centerline method for quantitative assessment of ventricular boundary. She measured the motion along 100 chords constructed perpendicular to a centerline drawn midway between the ED and ES LV contours. The Centerline method was developed to find the extent of local LV wall motion. The algorithm consists of the following principle: If the two LV polygons are ED and ES boundaries, ED being having larger number of points on it, then first linearly interpolate the larger contour to get 200 points and then for each tuple of three points on this contour a perpendicular is drawn to the tangent of the circle passing through these 3 points. Centers of these perpendicular constitute the centerline. On the similar basis of computing perpendicular from one vertex to another polygon, polyline distance metric is based. We give the formal definition as follows: The polyline distance $D_s(B_1 : B_2)$ between two polygons representing boundary B_1 and B_2 is symmetrically defined as the average distance between a vertex of one polygon and the boundary of the other polygon. To define this measure precisely, we first need to define a distance $d(v, s)$ between a point v and a line segment s. The distance $d(v, s)$ between a point v having coordinates (x_o, y_o), and a line segment having end points (x_1, y_1) and (x_2, y_2) is:

$$d(v,s) = \begin{cases} \min\{d_1, d_2\}; & if \ \lambda < 0, \lambda > 1 \\ |d^{\perp}|; & if \ 0 \leq \lambda \leq 1, \end{cases} \tag{5}$$

where

$$\begin{aligned} d_1 &= \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} \\ d_2 &= \sqrt{(x_0 - x_2)^2 + (y_0 - y_2)^2} \\ \lambda &= \frac{(y_2 - y_1)(y_0 - y_1) + (x_2 - x_1)(x_0 - x_1)}{(x_2 - x_1)^2 + (y_2 - y_1)^2} \\ d^{\perp} &= \frac{(y_2 - y_1)(x_1 - x_0) + (x_2 - x_1)(y_0 - y_1)}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \end{aligned} \tag{6}$$

The distance $d_b(v, B_2)$ measuring the polyline distance from vertex v to the boundary B_2 is defined by:

$$d_b(v, B_2) = \min_{s \in \text{ sides } B_2} d(v, s) \tag{7}$$

The distance $d_{vb}(B_1, B_2)$ between the vertices of polygon B_1 and the sides of polygon B_2 is defined as the sum of the distances from the vertices of the polygon B_1 to the closest side of B_2.

$$d_{vb}(B_1, B_2) = \sum_{v \in \text{ vertices } B_1} d(v, B_2)$$

Reversing the computation from B_2 to B_1, we can similarly compute $d_{vb}(B_2, B_1)$. Using Eq. 7, the polyline distance between polygons, $D_s(B_1 : B_2)$ is defined by:

$$D_s(B_1 : B_2) = \frac{d_{vb}(B_1, B_2) + d_{vb}(B_2, B_1)}{(\#vertices \in B_1 + \#vertices \in B_2)} \tag{8}$$

5.1 Mean error (e^{poly}_{NFP}) or Measure of Goodness

Using the definition of the polyline distance between 2 polygons, we can now compute the mean error of the overall calibration system. It is denoted by e^{poly}_{NFP} and defined by:

$$e^{poly}_{NFP} = \frac{\sum_{t=1}^{F} \sum_{n=1}^{N} D_s(G_{nt}, C_{nt})}{F \times N} \tag{9}$$

where, $D_s(G_{nt}, C_{nt})$ is the polyline distance between the ground truth G_{nt} and calibrated polygons C_{nt} for patient study n and frame number t. Using the definition of the polyline distance between 2 polygons, the standard deviation can be computed as:

$$\sigma^{poly}_{NFP} = \sqrt{\frac{\sum_{t=1}^{F} \sum_{n=1}^{N} \{\sum_{v \in \text{ vertices } G_{nt}} \left(d_b(v, C_{nt}) - e^{poly}_{NFP}\right)^2 + \sum_{v \in \text{ vertices } C_{nt}} \left(d_b(v, G_{nt}) - e^{poly}_{NFP}\right)^2\}}{N \times F \times (\#vertices \in B_1 + \#vertices \in B_2)}} \tag{10}$$

5.2 Error per vertex and Error per Arc Length

Using the polyline distance formulaes, we can compute the error per vertex from one polygon (ground truth) to another polygon (calibrated). This is defined as the mean error for a vertex v over all the patients and all the frames. The *error per vertex* for a fixed vertex v when computed between ground truth and calibrated boundary, defined by:

$$e^{GC}_{v} = \frac{\sum_{t=1}^{F} \sum_{n=1}^{N} d_b(v, G_{nt})}{F \times N} \tag{11}$$

Similary we can compute the *error per vertex* between calibrated and ground truth using eq. 7. *Error per arc length* is computed in the following way: For the values e_v^{GC} where v=1, 2, 3, ... P_1, we construct a curve f^{GC} defined on the interval [0,1] which takes the value e_v^{GC} at point x which is the *normalized arc length* to vertex v and whose in between values are defined by linear interpolation. We compute the curve f^{CG}, between calibrated boundary and ground truth boundary in a similar way. We then add algebraically these two curves to yield the final *error per arc length*, given as: $f = \frac{f^{GC}+f^{CG}}{2}$.

6 Data Analysis Using IdCM, InCM & Greedy Method

The performance of the system can be judged by evaluating the error measures on the test data $\mathbf{Q}_{te}$ set. This section discusses the performance of IdCM, InCM and the Greedy methods. All our performance is with respect to the original ground truth boundaries having P_1=100 vertices.

6.1 Data Analysis 1: Vertex Optimization

We find the mean error as a function of the number of polygon boundary vertices on the left ventricle contour. The optimization curve and operating point is shown in figure 7. The data

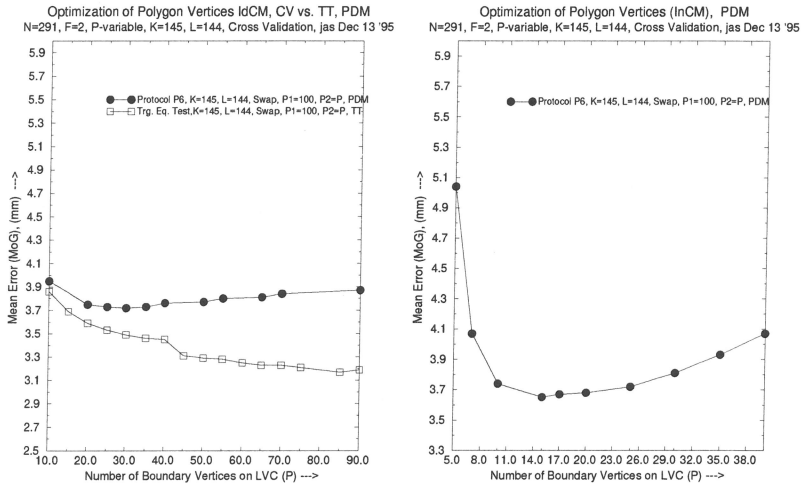


Figure 7: Vertex Optimization using the polyline distance metric. **Left**: IdCM (identical coefficient method), cross validation vs. TT, **Right**: InCM (independent coefficient method). Note the IdCM operating point is 30 vertices and InCM operating point is 15 vertices.

base consists of N=291 patient studies and the selected number of partitions K=145. We now vary the number of vertices P_2 on the left ventricle polygon varying it from 10 vertices to 90 vertices with 5 vertex increments. If L=144 are the training sets then for each combination there are $K - L$ test set boundaries on which the error is computed. We choose the number of vertices P_2 to minimize the error on the test set. Since there are ${}^{K}C_{L}$=145 trials, each trial has $(K - L)$ subsets, each subset consists of $\frac{N}{K}$ patients (in a protocol if $\frac{N}{K}$ is not a perfect division, then for the last trial in ${}^{K}C_{L}$ combinations, we have $(K - L + r_p)$ patients as testing set, where r_p is a remainder number of $\frac{N}{K}$) and each patient consists of P_2 vertices and F=2 frames. We thus get the total number of points as: $F \times {}^{K}C_{L} \times (K - L) \times \frac{N}{K} \times P_2$, resulting in: $N \times F \times P_2 \times \frac{(K-1)!}{(K-L-1)!L!}$ points for each (N, K, L, P_2) tuple. Since we are computing the polyline distances, the number of operations is $N \times F \times P_1^2 \times P_2 \times \frac{(K-1)!}{(K-L-1)!L!}$. We see from the plot 7 that the *optimal* number of vertices in InCM is about *half* the optimal number of vertices in IdCM, the reason being that the number of coefficients that have to be estimated in InCM is about *four* times the number of coefficients that have to be estimated in IdCM.

6.2 Data Analysis 2,3: Cumulative distribution of $(\frac{ED+ES}{2})$ errors and Error Per Arc Length along LVC

We show here the cumulative distribution of end frame errors $(\frac{ED+ES}{2})$ from both the calibration methods using IdCM and InCM and as shown in figure 8. Figure 9 demonstrates the mean *error per arc length* along the LVC. The abssisa shows the length of the arc starting from AAV. The ordinate shows the error at each vertex in mm. As seen in the plot, the mean *error per vertex* is largest near the middle of the normalized arc length which is close to the apex of the left ventricle. Thus the error is maximum in the apex region.

We see that Greedy algorithm does *best* in the apex zone compared to the IdCM and InCM methods. The *error per vertex* in figure 9 shows that in the ED frame, the apex zone error is reduced by **8.5** mm (from **12.5** mm to about **4** mm), while in ES frame, the apex zone error is reduced by **3 mm** (from **9** mm to **6 mm**). The corresponding mean error over ED and ES frames of the pixel-based boundaries was **6.4** mm which is reduced to **3.8** mm in IdCM and

3.5 mm in Greedy. As per our assumption, the error is least at the end points of the LVC since the AoV plane is known, thus the *error per vertex* curve drops at both ends.

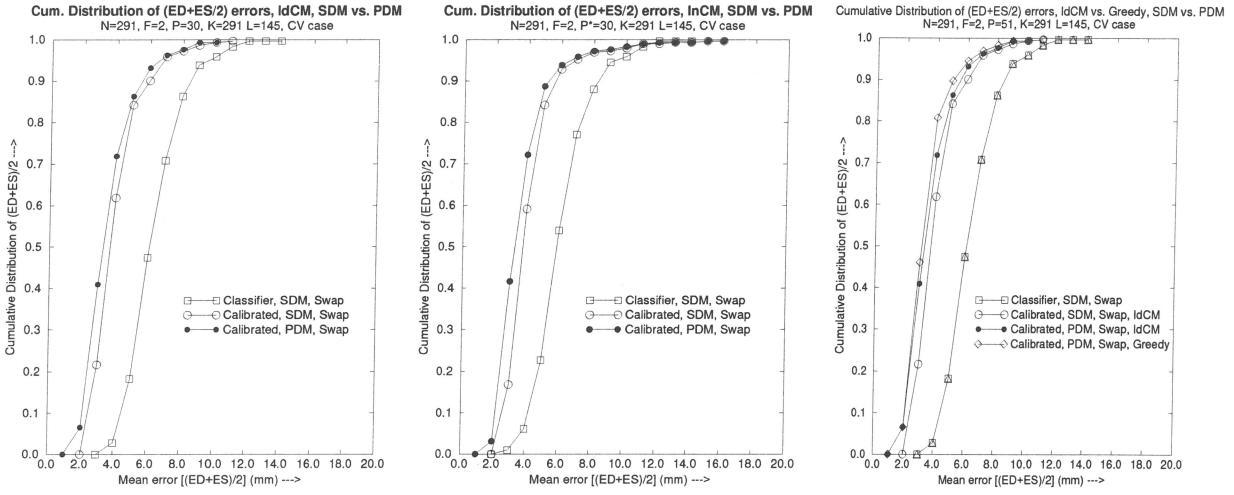


Figure 8: Cum. distribution vs. mean error of $(\frac{ED+ES}{2})$ errors. **Left**: Identical coefficient method. **Middle**: Independent coefficient method. **Right**: Greedy vs. IdCM. The curves shows that 80% patient estimated boundaries have error $\leq$ 4 milli-meters in IdCM, while 72% of the patients have error $\leq$ 4 mm in InCM. and 81% of the patients have error $\leq$ 4 mm in Greedy. Partition Protocol Parameters: N=291, F=2, K=145, L=144, P_1=100, P_2=30.

6.3 Data Analysis 4: ED and ES errors Vs. InCM Pool vertices

Here we show the effect of the Greedy calibration scheme. Figure 10 shows the drop in ED and ES frame errors when the IdCM pool vertices are transferred to InCM pool. This is implemented using the Greedy *do-while loop* where some columns (or vertices) of IdCM matrix $\hat{\mathbf{R}}_{id}$ are replaced by corresponding columns (or vertices) of InCM matrix, $\hat{\mathbf{R}}_{in}$. Plot 10 (right) shows that the Greedy algorithm reduces the error by **0.3** mm over IdCM. We also observe that the best number of vertices for IdCM is 30 while for InCM it is 15. The best performance over all the three techniques is by the Greedy algorithm with the number of vertices being 30. In the Greedy calibration technique, the error does not rise very sharply after 30 vertices but rises gradually by $(\frac{1}{100})^{th}$ of a milli-meter from 30 vertices to 40 vertices.

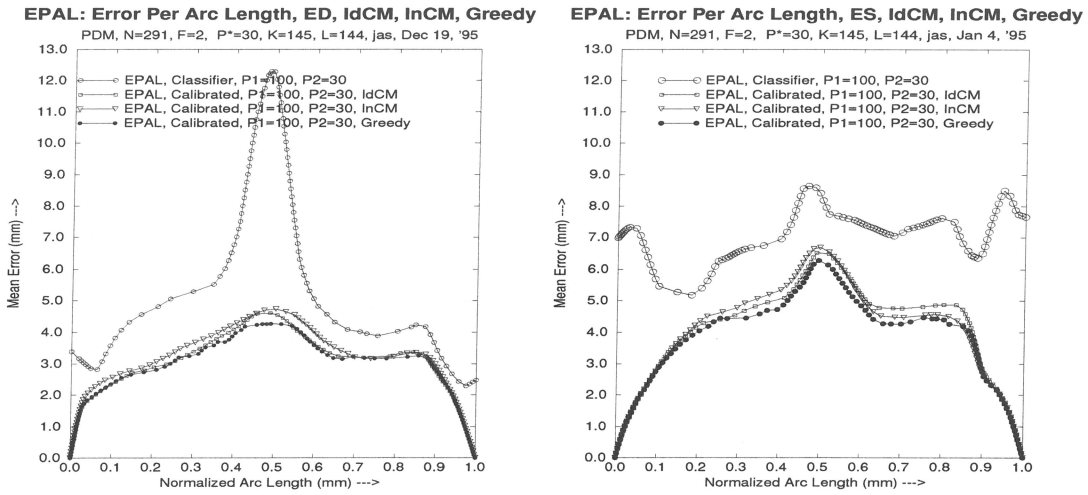


Figure 9: Mean error per arc length (EPAL) using polyline distance method where we superimpose 4 curves, the initial raw (perturbed), identical coefficient method, independent coefficient method and greedy calibration method. **Left**: ED frame **Right**: ES frame Greedy does the best out of all the 3 techniques of calibration in both ED and ES frames. Greedy method also does better in the apex zone (where the dye was unable to reach) compared to all other vertices. In ED frame, apex error reduces by **8.5** mm and in ES frame, apex error reduces by **3** mm.

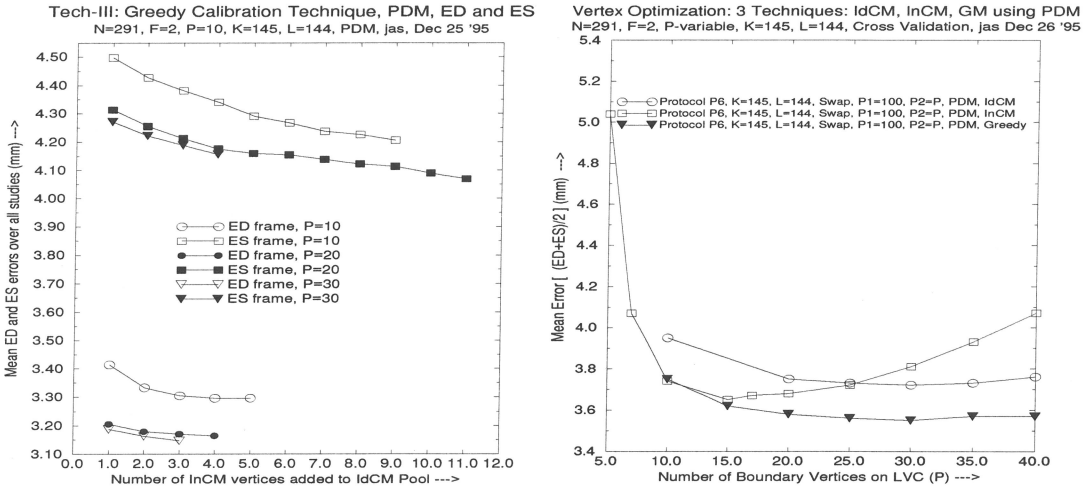


Figure 10: Greedy Performance: **Left**: Plot showing the reduction in the error for ED and ES frames when some vertices are calibrated using IdCM and others using InCM. With the increase InCM pool, the error drops. **Right**: Comparion of 3 calibation techniques, Greedy does the best. Partition Protocol Parameters: N=291, F=2, K=145, L=144, P_1=100, P_2=30. Mean error for IdCM=**3.8** mm, InCM=**3.9** mm and Greedy=**3.5** mm. So Greedy improve by **0.3** mm over IdCM method.

7 Reliability Algorithm and Validation: Rejection Analysis

In this section we develop a general and automatic validation technique that can detect the left ventricle boundaries whose mean end frame boundary errors ($\frac{ED+ES}{2}$) are above a given threshold, R_{th}. This validation scheme has the following features and advantages: 1. It determines those left ventricle boundaries from the database (which could be coming from any source) whose ($\frac{ED+ES}{2}$) error is above a given threshold error. 2. The scheme provides feedback to the boundary calibration system so that the system knows which left ventricle boundaries can be rejected. 3. It estimates the overall performance of the boundary calibration system without taking the rejected boundaries into consideration. 4. It provides a check for consistency and reliability of the output estimation algorithms (for example: Classification algorithm, IdCM, InCM or Greedy). For this test we need three inputs: the left ventricle boundary coordinates (x, y), the gray scale left ventriculograms, and the binary indicator for the left ventricle region (1 for inside the left ventricle region and 0 for outside the LV region). This boundary rejection scheme is based on the gray scale information near the boundary of the left ventricle. Using a mutually exclusive window of a fixed size centered on the left ventricle boundary vertex and along the left ventricle contour, we compute the mean gray level value for part of the window which is inside the LV region and part of the window which is outside the LV region (see fig. 11, left). We then associate this difference in the mean gray scale intensities (also called as contrast values) to the corresponding vertex of the LV observed boundary. Since we know the observed boundary errors ($\frac{ED+ES}{2}$) for patient study n estimated from any technique, we can regress these contrast values against the observed errors to compute the rejection training coefficients. These training coefficients can then be used to find the predicted errors on the test contrast boundary data (CBD). The predicted errors which are above the threshold, we state them to be rejected boundaries or have unacceptable boundary errors. This rejection system is a reliability test because this system helps in determining the procedure as to how reliable the estimated boundaries are. In section

7.1, we give the mathematical statement for computing the predicted errors. Section 7.2 discusses the reliability test algorithm and training system, section 7.3 gives the mathematical formulae for computing the probability of false alram, probability of mis-detection, mean predicted errors when n patients are rejected from the database. Finally we discuss the resultant reliability curves from our experiments.

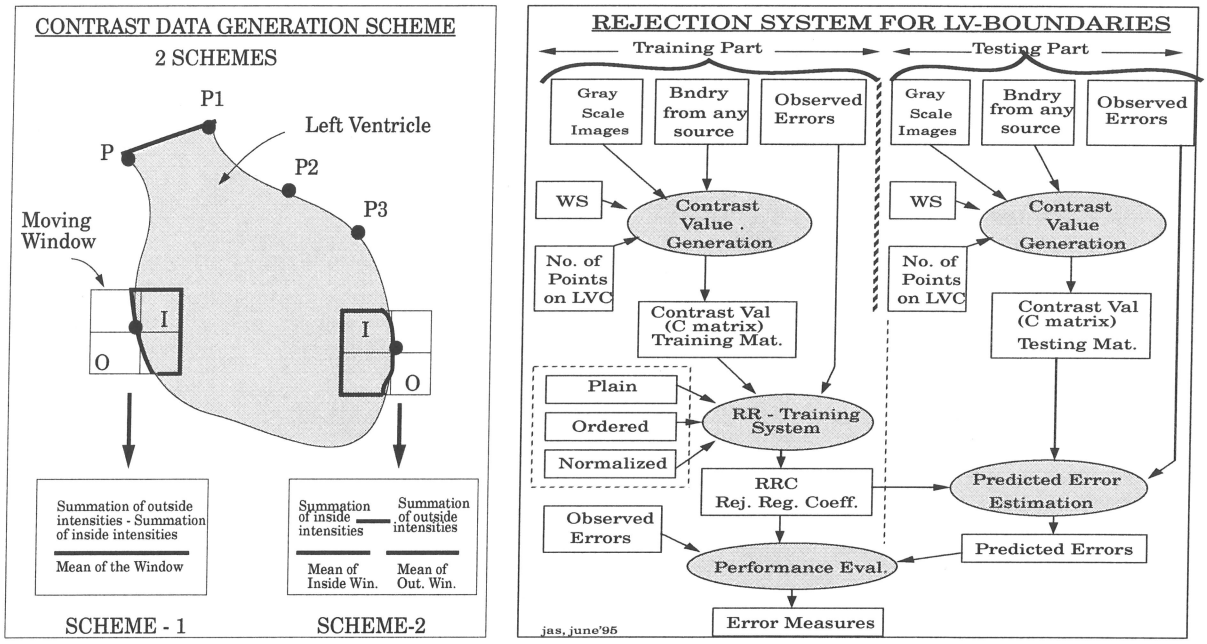


Figure 11: **Left**: Contrast Data generation process showing 2 schemes. (i) Non-Separate Contrast data and (ii) Separate Contrast Data. **Right**: Overall system for reliability and validation test. It has 2 parts. First part is the generation of the training rejection coefficients from the contrast data set. Second part consists of applying the training coefficients on the test contrast data to generate the predicted errors. These predicted errors undergo the performance evaluation of the system. WS is the window size which moves along the LVC. We use 3 statistical techniques for estimating the rejection regression coefficients (RRC), depending upon the layout of the constrast data matrix (C).

7.1 Problem Statement: Reliability Equation

We now present the mathematical statement for the estimation of the prediced errors for the left ventricle boundaries given the contrast data matrix **C** and observed end frame error ($\frac{ED+ES}{2}$) vector **e**. The predicted errors are used for spotting the left ventricle boundaries that are above the given error threshold.

Boundary Contrast data refers to the gray scale data generated along the left ventricle boundary by superimposing the left ventricle boundary over the gray scale left ventriculograms. At each chosen vertex of the LVC, there is a corresponding contrast value. The contrast value for a given window is the difference between the mean gray scale intensities inside and outside the window. This superimposed boundary can be from any given boundary estimation algorithm. *Observed errors* are the mean end frame boundary errors ($\frac{ED+ES}{2}$) for each patient study n. Let c'_n=$[c_1, \ldots, c_P]$ be the row vector of contrast values of dimension P for patient study n, where, n=$1, \ldots, N$. Let $\mathbf{e}$=$[e_1, \ldots, e_N]$ be the vector of the observed end frame errors, ($\frac{ED+ES}{2}$) for N patient studies. For the predicted errors of the patient study n, we are:

- **Given**: Corresponding pairs of contrast data matrix $\mathbf{C}$ $[N \times (P+3)]$, and the observed errors $\mathbf{e}$ $[N \times 1]$, respectively as:

$$\mathbf{C}^{N\times(P+3)} = \begin{pmatrix} c'_1 \underbrace{1\, s_1\, m_1} \\ \ldots \\ \ldots \\ \ldots \\ c'_N \underbrace{1\, s_N\, m_N} \end{pmatrix} \qquad \mathbf{e}^{N\times 1} = \begin{pmatrix} e_1 \\ \ldots \\ \ldots \\ \ldots \\ e_N \end{pmatrix}$$

  where s_n, m_n is the standard deviation and mean of the P contrast values for the patient study n along the LVC.

- Let $\mathbf{a}$ $[N \times 1]$ be the vector of unknown regression coefficients.

- The problem is to estimate the coefficient vector $\mathbf{a}$, to minimize $\| \mathbf{e} - \mathbf{C}\,\mathbf{a} \|^2$. Then for any boundary contrast data matrix $\mathbf{C}$, the predicted error for the boundary is given as: $\mathbf{C}\hat{\mathbf{a}}$, where $\hat{\mathbf{a}}$ is the estimated coefficients.

7.2 Reliability Algorithm: C-e Relation

Following are algorithm steps for estimating the predicted errors which form the basis for the reliability of the boundaries estimated from any boundary estimation algorithm.

1. Boundary Contrast matrix generation ($\mathbf{C}$): The contrast matrix can be generated by first superimposing the LV boundaries over the gray scale images (LVG). We use two

methods for generating the contrast data. The mathematical statements for expressing the contrast value at a vertex i are given as follows: Let g_p be the gray scale value for pixel p. Let I and O be the sets which contain the pixels inside and outside the mutually exclusive moving window. Let G_I and G_O be the sum of all the gray scale intensities for pixels which are inside and outside the window given by:

$$G_I = \sum_{p \in I} g_p \;\; \& \;\; G_O = \sum_{p \in O} g_p \tag{12}$$

Let f_I and f_O be the cardinality of the sets I and O. Using the above notations, we give the expression for the contrast value at vertex i using the two methods.

$$c_i = \frac{(G_I - G_O)_i}{(f_I + f_O)} \;\; (method - I) \tag{13}$$

$$c_i = (\frac{G_I}{f_I})_i - (\frac{G_O}{f_O})_i \;\; (method - II) \tag{14}$$

Note, the difference between the above equations lies in the way the sums of the gray scale intensities G_I and G_O are subtracted in the two cases. In method I, we compute the gray scale difference between the total gray scale values inside and outside the window and then divide the difference by the total number of pixels in the window. In second case, we find the mean values for inside and outside separately and then subtract it. The first one is named as non-separate contrast data while the later separate contrast data. This is shown in fig. 11. We also compute the mean and standard deviation of the contrast values for each patient study n. Now, using Eq's. 12 and 13 we find the contrast values for P vertices on LVC which yields the contrast vector c'_n for patient study n. Repeating this process for all the studies N, we get the contrast data matrix $\mathbf{C}$ $[N \times (P+3)]$.

2. Estimation of the training coefficients ($\hat{\mathbf{a}}$): We compute the training coefficients using the standard Least Squares to minimize the error function ϵ^2_{rr} given as:

$$\epsilon^2_{rr} = \| \mathbf{C}_{tr}\,\mathbf{a} - \mathbf{e} \|^2 \;\; \& \;\; \hat{\mathbf{a}} = \underbrace{(\mathbf{C}^T_{tr}\,\mathbf{C}_{tr})^{-1}\,\mathbf{C}^T_{tr}\,\mathbf{e}} \tag{15}$$

Note, $\mathbf{C}_{tr}$ has dimension of $\underbrace{N_{tr} \times (P+3)}$, $\hat{\mathbf{a}}$ has dimension $\underbrace{(P+3) \times 1}$, and $\mathbf{e}$ has dimension of $N_{tr} \times 1$. N_{tr} are the number of training studies.

3. Estimating the predicted errors $\hat{\mathbf{e}}$ on test boundaries (N_{te})

$$\hat{\mathbf{e}} = \mathbf{C}_{te}\,\hat{\mathbf{a}} \tag{16}$$

$\mathbf{C}_{te}$ has dimension $N_{te} \times (P+3)$.

4. Statistical techniques and window sizes: Here we repeat the rejection regression coefficient estimation process for three different set of statistical techniques. They are: 1. Plain, 2. Ordered, 3. Normalized and Ordered. Plain, because we take the contrast values are arranged as per the vertex number of the LVC, Ordered, because we arrange the contrast values in increasing order and then use it for calibration. Normalized because, we normalize the constrast values by its standard deviation. For each of the above techniques, we also change the window size for generation of the contrast data. two sets of window sizes were taken into consideration namely, 11×11 and 22×22.

	Assigned	
True	m_{rr}	m_{rs}
True	m_{sr}	m_{ss}

Table 1: Contengency Table

7.3 Performance:Mathematical Formulaes for Probability of Mis-Detection, False Alram, and Mean Predicted Errors

Let $(s_{(i)}, e_{(i)})$ be the name and the error for a patient from the list of ideal errors and corresponding names. This list is the output of the boundary estimation algorithm. Similarly let $(q_{(j)}, p_{(j)})$ be the name and the error for a patient from the list of the predicted errors and corresponding names. This list of errors results from the output of boundary rejection scheme. Note that the errors in both lists are sorted in increasing order and hence the names of the patient boundaries are not in same order in both the lists. These two lists will be used for finding the probabilities of mis-detection vs. probabilities of false alram, mean errors for rejected and non-rejected patient boundaries.

Let us define mathematically the terms of the contingency table shown in table (1). m_{rr} is the number of patient studies truly rejected and was assigned to be rejected. m_{rs} is the number of patient studies truly rejected but was assigned to be non-rejected (selected). m_{sr} are the number of patient studies truly non-rejected (selected) but was assigned to be rejected. m_{ss} is number of patient studies truly non-rejected (selected) and were assigned to be non-rejected (selected). These four terms can be mathematically expressed using our two input lists as:

$$m_{rr}(n) = \#\{ i \mid \exists\, j, (s_{(i)} = q_{(j)}),\, j > (N-n),\, i > (N-n) \tag{17}$$

$$m_{rs}(n) = \#\{ i \mid \exists\, j, (s_{(i)} = q_{(j)}),\, j \leq (N-n),\, i > (N-n) \tag{18}$$

$$m_{rs}(n) = \#\{ i \mid \exists\, j, (s_{(i)} = q_{(j)}),\, j > (N-n),\, i \leq (N-n) \tag{19}$$

$$m_{rs}(n) = \#\{ i \mid \exists\, j, (s_{(i)} = q_{(j)}),\, j \leq (N-n),\, i \leq (N-n) \tag{20}$$

Note that the elements of the contigency table are function of each set of patient studies rejected n. Using these definitions we can express the probability of mis-detection $\mathbf{P}_{md}(n)$ and probability of false alram $\mathbf{P}_{fa}(n)$ as a function of the total number of patient studies rejected n:

$$\mathbf{P}_{md}(n) = \frac{m_{rr}(n)}{m_{rr}(n) + m_{rs}(n)} \quad \& \quad \mathbf{P}_{fa}(n) = \frac{m_{sr}(n)}{m_{sr}(n) + m_{ss}(n)} \tag{21}$$

7.3.1 Errors for rejected $\bar{p}_r(n)$ and non-rejected studies $\bar{p}_{non}(n)$

Given the above lists, we can express the mean error of the rejected and non-rejected patient boundaries for the ideal and cross-validation cases as follows: Let $\bar{e}_r(n)$ and $\bar{e}_{non}(n)$ be the errors for rejected and non-rejected (selected) patient boundaries for the ideal case. Let $\bar{p}_r(n)$ and $\bar{p}_{non}(n)$ be the errors for rejected and non-rejected (selected) patient boundaries for the cross-validation case. For the ideal case:

$$\bar{e}_r(n) = \frac{1}{n} \sum_{i=N-n+1}^{N} e_{(i)} \quad \& \quad \bar{e}_{non}(n) = \frac{1}{N-n} \sum_{i=1}^{N-n} e_{(i)} \tag{22}$$

For the cross-validation (CV) case:

$$\bar{p}_r(n) = \frac{1}{n} \sum_{i=N-n+1}^{N} p_{(j)} \quad \& \quad \bar{p}_{non}(n) = \frac{1}{N-n} \sum_{i=1}^{N-n} p_{(j)} \tag{23}$$

The rejection threshold for n patient studies is computed using the list of predicted errors and is given as: $R_{th}(\mathrm{n})=p_{N-n}$.

7.4 Relationships developed: Reliability Curves

1. $\bar{e}_{non}(n)$ and $\bar{p}_{non}(n)$ vs. n: This is the relation between the mean error for the non-rejected patients for both ideal and cross-validation cases and the total number of patients rejected n. $\bar{e}_{non}(n)$ and $\bar{p}_{non}(n)$ can be computed using the Eq.'s (22) and (23). We took a data set of 245 studies for these experiments. In the first case n is made to increase from 1 to 50 and second case we increase n from 1 to all the patients in the data base (N). These 2 plots are shown in figure (12). We see as we rejected 20% percent of the patient boundaries the mean error for $\bar{e}_{non}(n)$ and $\bar{p}_{non}(n)$ drops down from 4.4 milli-meter to 4 milli-meters (for CV case) and 3.85 millimeters (for Ideal case) respectively. For the plot when all the patients (N) are taken into account, there is a steep drop of the mean error dropping to 1.98 milli-meter. This experiment was done when the ideal errors are estimated from IdCM.

2. $\bar{e}_{non}(n)$ and $\bar{p}_{non}(n)$ vs. R_{th}: The is the relation between the mean error for the non-rejected patients for both ideal and cross-validation cases and the rejection threshold R_{th}. $\bar{e}_{non}(n)$ and $\bar{p}_{non}(n)$ can be computed using the Eq.'s (22) and (23). This is show in fig. 12. With the reduction in the rejection threshold R_{th}, mean error drops down gradually.

3. $\mathbf{P}_{md}(n)$ vs. $\mathbf{P}_{fa}(n)$: This is the relation for the probability of mis-detection versus the probability of false alram for the set of n studies rejected. The curve is shown in fig. 13. With increase in $\mathbf{P}_{md}(n)$, $\mathbf{P}_{fa}(n)$ decreases. Note both scales are from 0 to 1. (see Eq. (21)).

4. $\mathbf{P}_{md}(n)$ vs. n: This is the relation for the probability of mis-detection versus the number of patients rejected n. (see Eq. (21)). This plot demonstrates that when more patients are rejected then the probability of mis-detection falls down. (see fig. (12)).

5. $\mathbf{P}_{fa}(n)$ vs. n: This is the relation the probability of false alram and the number of patients rejected n, computed using the relations expressed in Eq. (21). We see from the plot, with increase in n, the probability of false alram ($\mathbf{P}_{md}(n)$) increases. We have shown two cases, first when the WS is 11 x 11 and second when the WS is 22 x 22. When n is small, that is when small number of patient boundaries are rejected, the $\mathbf{P}_{fa}(n)$ is more or less the same for both window sizes, but when large number of patients are rejected, then the $\mathbf{P}_{fa}(n)$ is higher for larger window size.

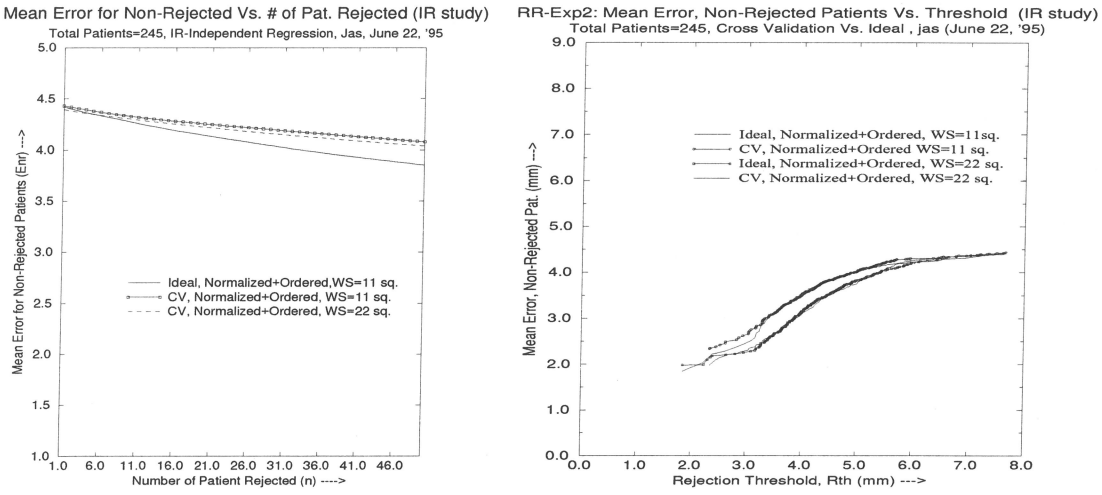


Figure 12: **Left**: Plot of mean of non-rejected patient studies vs. number of patients rejected (n). **Right**: Plot of mean of non-rejected patient vs. rejection threshold.

8 Discussions

The Greedy algorithm for error correction fuses two sets of estimated boundaries: boundaries produced by the Identical Coefficient Method and boundaries produced by Independent Coefficient Method. If these two estimated boundaries have large errors with respect to the ground truth (ideal or hand-drawn boundary), then the Greedy algorithm rapidly identifies those vertices which are too far from the ideal vertices, significantly reducing the boundary error. Nevertheless, the true shape of the left ventricle depends upon the number of optimized boundary vertices selected on the left ventricle contour. If the number of vertices is less than the optimum number of vertices, the true LV shape may not be represented. On the other

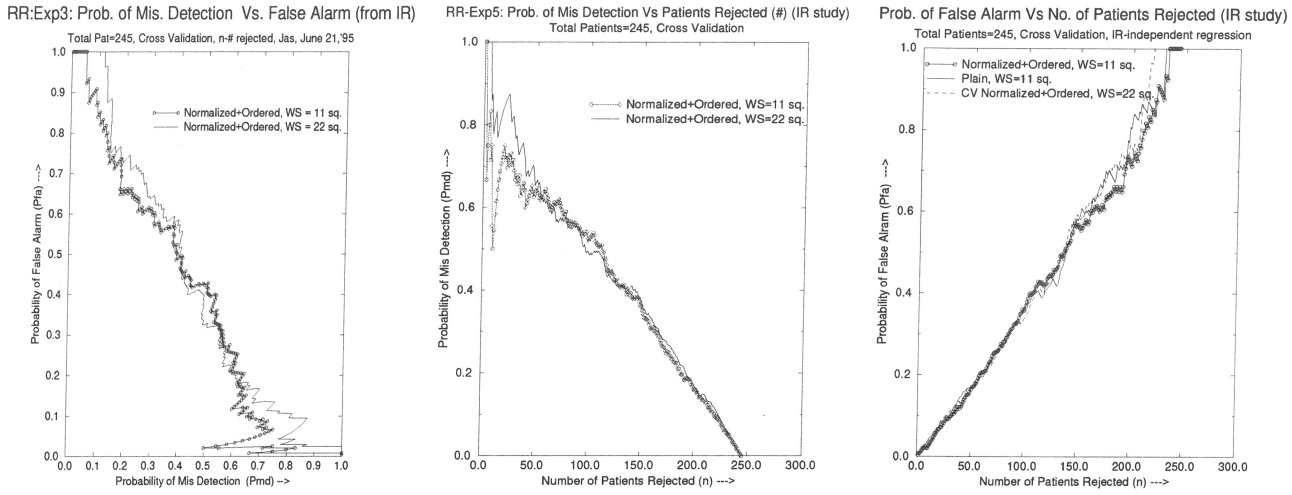


Figure 13: **Left**: With the increase in probability of mis-detection $\mathbf{P}_{md}(n)$ the probability of false-alarm $\mathbf{P}_{fa}(n)$ decreases. Note that each point on the curve corresponds to the total number of patients rejected (n). There are 2 cases shown in this plot. First when the window size (WS) is 11×11, and second case when the WS is 22×22. **Middle**: Plot of $\mathbf{P}_{md}(n)$ Vs. n **Right**: Plot of $\mathbf{P}_{fa}(n)$ Vs. n

hand, if the number of vertices are larger than the optimized boundary vertices, the shape becomes more accurate, but generalization is lost. Therefore, we first optimize the techniques: IdCM and InCM and then fuse the best results, Consequently, the error will always improve, but the drop in the error will depend upon the following factors: 1. the number of vertices on LVC, 2. the initial errors of IdCM and InCM boundary data before the fusion starts, which in turn depends upon the number of data vectors N for training the calibration model. 3. the starting error value (ϵ) before the Greedy loop starts. (In our case, starting value is the best error for IdCM boundary data).

These preliminary results indicate that the three sets of calibration algorithms significanly reduce the boundary error over the image processing algorithms. However, we can make the left ventricle boundary calibration system robust by padding information or features like the apex information to the $\mathbf{Q}$ matrix (classifier data) to improve the accuracy. Our algorithm requires no operator assistance; further, the algorithm is relatively simple and can be implemented on any commercial imaging system.

9 Conclusions

We presented three sets of calibration algorithms, the Identical Coefficient Method, the Independent Coefficient Method and the Greedy Calibration Method. The *Greedy calibration algorithm* for calibrating the initial *pixel-based* classifier boundaries takes the best of the other two calibration Methods: the Identical Coefficient Method and the Independent Coefficient Method. The mean error over ED and ES frames using a *cross-validation* protocol and polyline distance metric is **3.5** millimeters over the database of **291** patient studies. The Greedy algorithm is a considerable improvement over the Identical Coefficient Method by **0.3** millimeters which is significant for the accuracy of the overall calibration system. The Greedy algorithm performs best in the apex zone of the left ventricle where the dye is unable to propagate, reducing the error by approximately **8.5** mm. Thus we see that the calibration constitutes a significant last step for boundary estimation.

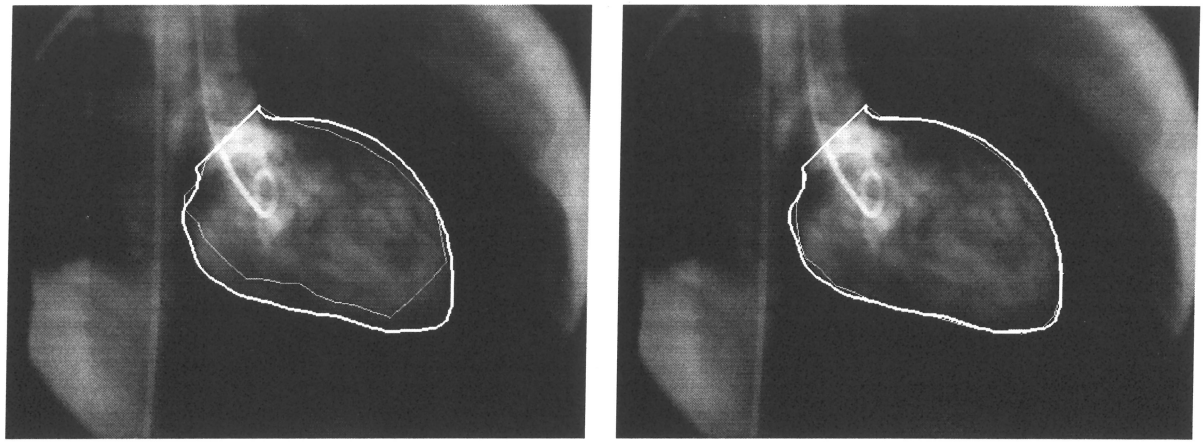
10 Acknowledgements

The authors would like to thank Drs. Poggio (MIT), Linda G. Shapiro, Zick, Lytle, Meldrum, Stuetzle for their assistance in discussions. We also thank Ms. Nan and others from CVRTC, University of Washington Medical Center, Seattle, for tracing the left ventricle borders in Cardioangiograms. Thanks are also due to Dr. Kathy S. Ewing, University of Washington for editing the manuscript.

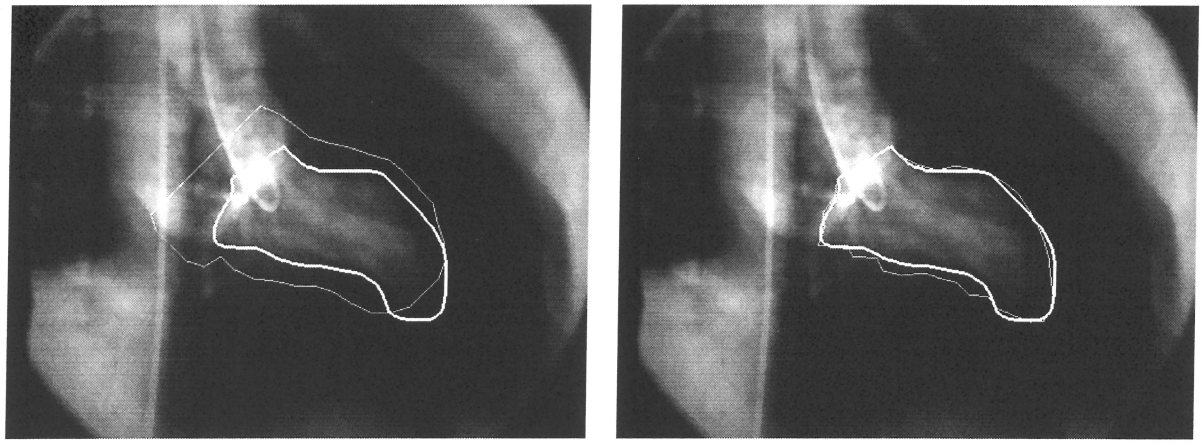
References

[1] C. K. Lee, *Automated Boundary Tracing Using Temporal Information*, PhD Thesis, Dept. of Electrical Engineering, Universtiy of Washington, Seattle, 1994.

[2] W. Wunderlich and F. Fischer and T. Linderer and R. L. Kirkeeide, *Analytic isocenter calibration. A new approach for accurate x-ray gantries*, ANGIOLOGY, volume 46 (7), p577-82, July, 1995.

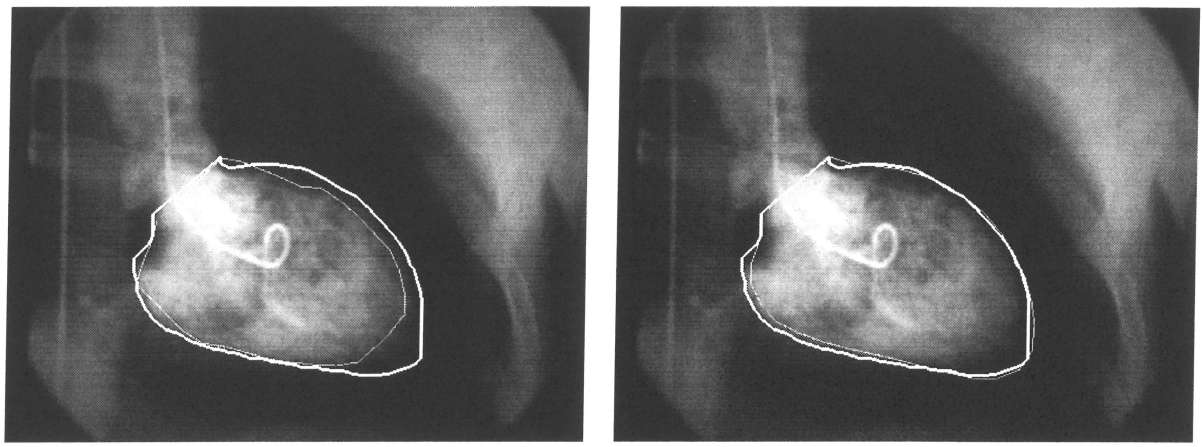


(a1) ED Frame: GT and Classifier (a2) ED Frame: GT and Estimated

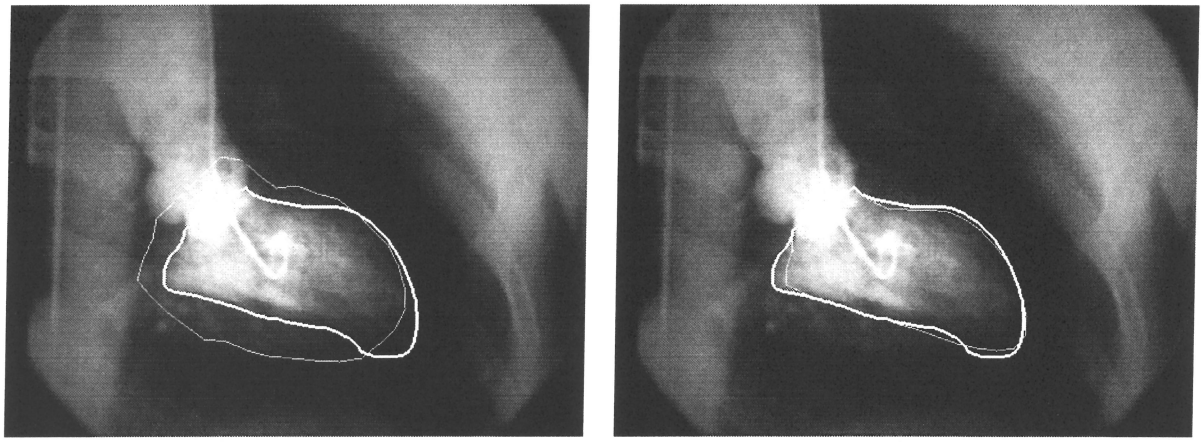


(b1) ES Frame: GT and Classifier (b2) ES Frame: GT and Estimated

Figure 14: Classifier vs. Estimated boundaries with gray scale in the back ground using Greedy Calibration Method. Quartile 1, **Thick LV contour** -Ground Truth, **Thin LV contour**-Classifier, and Estimated, **Upper**: (a1) Uncalibrated ED frame boundary with ground truth. (a2) Calibrated ED frame boundary with ground truth. **Bottom**: (b1) Uncalibrated ES frame boundary with ground truth. (b2) Calibrated ES frame boundary with ground truth. Calibration Parameters: N=291, K=145, L=144, F=2, P_1=100, P_2=35, Mean end frame error $(\frac{ED+ES}{2})$ = 1.16 mm, Mean error (e_{NFP}^{poly}) = 3.5 mm

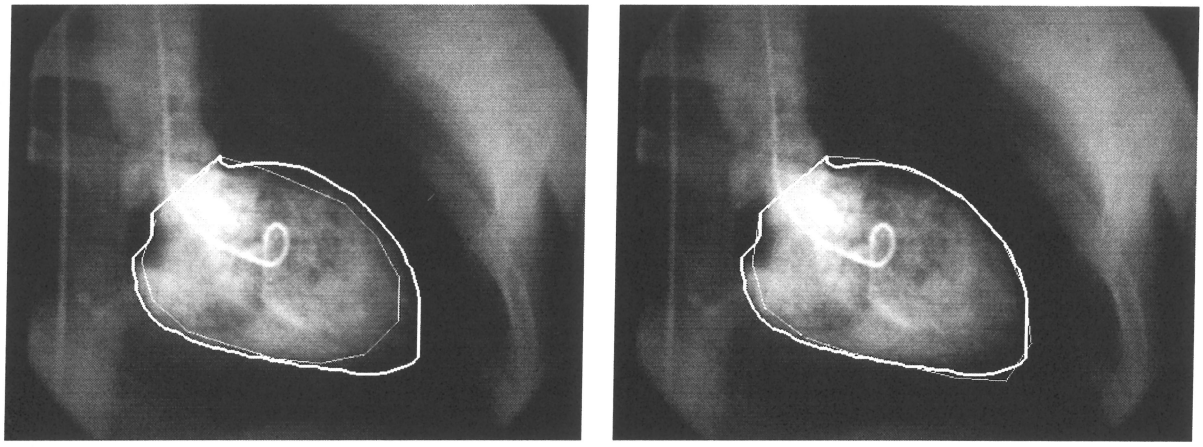


(a1) ED Frame: GT and Classifier (a2) ED Frame: GT and Estimated

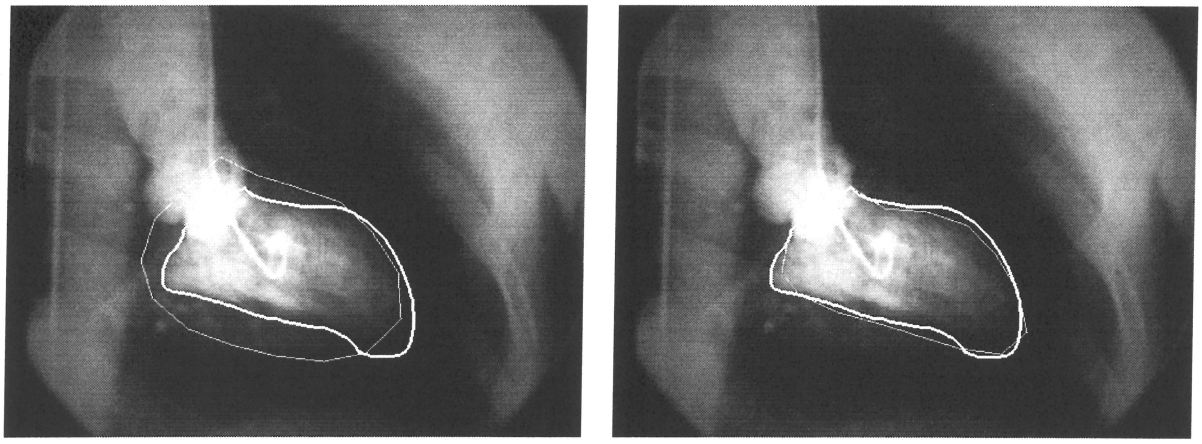


(b1) ES Frame: GT and Classifier (b2) ES Frame: GT and Estimated

Figure 15: Classifier vs. Estimated boundaries with gray scale in the back ground using Identical Calibration Method (IdCM) . Quartile 1, **Thick LV contour** -Ground Truth, **Thin LV contour**-Classifier, and Estimated, **Upper**: (a1) Uncalibrated ED frame boundary with ground truth. (a2) Calibrated ED frame boundary with ground truth. **Bottom**: (b1) Uncalibrated ES frame boundary with ground truth. (b2) Calibrated ES frame boundary with ground truth. Calibration Parameters: N=291, K=145, L=144, F=2, P_1=100, P_2=30, Mean end frame error $(\frac{ED+ES}{2})$ = 1.30 mm, Mean error (e_{NFP}^{poly}) = 3.7 mm



(a1) ED Frame: GT and Classifier (a2) ED Frame: GT and Estimated



(b1) ES Frame: GT and Classifier (b2) ES Frame: GT and Estimated

Figure 16: Classifier vs. Estimated boundaries with gray scale in the back ground using Independent Coefficient Method (InCM). Quartile 1, **Thick LV contour** -Ground Truth, **Thin LV contour**-Classifier, and Estimated, **Upper**: (a1) Uncalibrated ED frame boundary with ground truth. (a2) Calibrated ED frame boundary with ground truth. **Bottom**: (b1) Uncalibrated ES frame boundary with ground truth. (b2) Calibrated ES frame boundary with ground truth. Calibration Parameters: N=291, K=145, L=144, F=2, P_1=100, P_2^*=15, Mean end frame error $(\frac{ED+ES}{2})$ = 1.57 mm, Mean error (e_{NFP}^{poly}) = 3.65 mm

[3] Robert J. Moore, *Imaging Pinciples of Cardiac Angiography*, Aspen Publishers, Rockville Maryland, pages p1-258, 1990.

[4] Duglas S. Moodi and John Yiannikas, *Digial Subtraction Angigraphy of the heart and lungs*, Grune and Stratton Inc., Harcourt Brace Jovanovich Publishers, 1986.

[5] T.F. Cootes, C.J. Taylor, D.H. Cooper and J. Graham, *Computer Vision and Image Understanding*, Vol 61., No. 1, January, pp 38-59, 1995.

[6] (i) *U.S. Patent No.: 4,952,805*, (ii) *U.S. Patent No.: 5,164,993*, (iii)*U.S. Patent No.: 5,268,967*, (iv) *U.S. Patent No.: 4,731,863*, (v) *U.S. Patent No.: 5,028,782*, (vi) *U.S. Patent No.: 4,028,079*, (vii)*U.S. Patent No.: 4,962,539.*

[7] Florence H. Sheehan, Douglas K. Stewart, Harold T. Dodge, Suzanne Mitten, Edward L. Bolson and B. Greg Brown, *Variability in the measurement of regional left ventricular wall motion from contrast angiograms*, CIRCULATION, Ventriculography, vol. 68, number 3, p550-p559, 1983.

[8] Beier J, T Joerke, S Lempert, E Wellnhofer and H Oswald and E Fleck, *A Comparision of 7 Different Volumetry Methods of Left and Right Ventricle using Post-mortem Phantoms*, IEEE Computers in Cardiology, p33-p36, 1993.

[9] Jasjit S. Suri and Robert M. Haralick, *Systematic Error Correction in automatically produced boundaries in Low Contrast Ventriculograms*, International Conference in Pattern Recognition, Austria, 1996.

[10] T. S. Huang, *Modeling, analysis and visualization of nonrigid object motion,* International Conference on Pattern Recognition (ICPR), p361-p364, June, 1990.

[11] T. Chen, W. C. Lin and C. T. Chen, *Artificial Neural Networks for 3-d nonrigid motion analysis*, In Artificial Neural Networks in Engineering, St. Louis, Missouri, Nov, 1992.

[12] Chang Wen Chen and Thomas S. Huang, *Modelling, Analysis, and Visualization of Left ventricle Shape and Motion by Hierarchical Decomposition*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, No. 4, p342-p356, 1994.

[13] Harold T. Dodge, Harold Sandler, Donald W. Ballew and John D. Lord, *The use of Biplane Angiocardiolgy for the Measurement of Left Ventricular Volume in Man* The American Heart Journal, volume 41, p762-p776, November, 1960.

[14] Joshua Wynne, Laurence H. Greeen, Tift Mann, David Levin and Willam Grossman, *Estimation of Left Ventricular Volumes in Man from Biplane Cineangiograms Filmed in Oblique Projections*, The American Journal of CARDIOLOGY, volume 41, p726-p732, April, 1978.

[15] H. Lehmkuhl and Th Machnig and B Eicker and Karth and K Reynen and KBachmann, *Digital Subtraction Angiography:Feasibility of Densitometric Evaluation of Left Ventricular Volumes and Comparision to Measurements obtained by the monoplane Area-Length-Method*, IEEE Computers in Cardiology (CinC), p29-p32, 1993.

[16] Richard L. Griffith, Colin Grant and Howard Kaufman, *An Algorithim for locating the Aortic valve and the apex in left ventricular angiocardiograms*, IEEE Transactions on Biomedical Engineering, volume 21, No. 5, p345-p349, 1974.

[17] D. Geman, S. Geman, C. Graffigne and P.Dong, *Boundary Detection by Constrained Optimization*, IEEE Pattern Recognition and Machine Intelligence, volume 12, p609-p628, 1990.

[18] Silvio E. Papapietro and L. Richard Smith and William P. Hood and Richard O. Russell and Charles E. Rackley and William J. Rogers, *An Optimal method for angiographic definition and quantification of regional left ventricular contraction*, Computers in Cardiology, p294-p295, September, 1978.

[19] Anthony Rickards, Ricardo Seabra-Gomes and Paul Thurston, *The assessment of regional abnormalities of the left ventricle by angiography*, European Journal of Cardiology, p167-p182, volume 5, No. 2, 1977.

[20] Steven C. Klausner, Tarlton J. Blair, Walter F. Bulawa, Gary M. Jeppson, Robert L. Jensen and Paul D. Clayton, *Quantitative Analysis of Segmental Wall Motion Throughtout Systole and Diastole in the Normal Human Left Ventricle* CIRCULATION, vol. 65, number 3, March, 1982.

[21] G. B. John Mancini, *Clinical applications of cardiac digital angiography*, Raven Press, NY , p1-324, 1988.

[22] Dong Yoon Kim, John J. Kim, Peter Meer, Doron Mintz and Azriel Rosenfeld, *Robust Computer Vision: A Least Median of Squares based approach*, Center for Automation Research, University of Maryland,College Park, MD 20742, 1989.

[23] Frank R. Hampel, Elvezio M. Ronchetti, Peter J. Ronsseeuw and Werner A. Stahel, *Robust Statistics-The Approach Based on Influence functions,* John Wiley and sons, 1986.

[24] P.W. Holland and R. E. Welsch, *Robust Regression using iteratively reweighted least squares*, Communication Statistics, volume 6, p813-p828, Aug, 1977.

[25] Richard W. Hill and Paul W. Holland, *Two Robust Alternatives to Least Squares Regression*, Journal of American Statistical Assocation, volume 72, p828-p833, Dec, 1977.

[26] C. K. Chow and T. Kaneko, *Border Detection based on Local Characterstics* Compu. Biomed Res, volume 5, p388-p410, Aug, 1972.

[27] H.Azhari and I. Gath and R. Beyar and M . L. Marcus and S. Sideman, *Discrimination of healthy and diseased hearts by spectral decomposition of their ventricular three dimensional geometry,* IEEE Transactions on Medical Imaging, volume 10, no.2, p207-p215,1991.

[28] Jasjit S. Suri and Robert M. Haralick, *3 Methods of performance evaluation of the Calibrated LV boundaries*, Techinical report Number: ISL-TR-96-01, Dept. of EE, ISL, UW, Seattle, 1996.

[29] Florence H. Sheehan, *Cardiac Angiography*, W. B. Saunders Company, p109-p147, 1991.

[30] William H. Press and Brain P. Flannery and Saul A. Teukolsky and William T. Vetterling, *Numerical Recipes in C*, Cambridge University Press, 1988.

[31] P.J. Huber, *Robust Estimation of a location parameter* Annals Math Statis", volume 35, p73-p101, Aug, 1964.

[32] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, Addison-Wesley, 1991.

[33] R. W. Brower, *Evalutation of Pattern Recognition for the Apex of the Heart* Catheterization and Cardiovasular Diagnosis, volume 6, p145-p157, 1980.

[34] D. G. Gibson and T. A . Prewitt and D. J. Brown, *Analysis of left venticle wall movement during isovolumic relaxation and its relation to coronary artery disease*, Journal of Heart Br., volume 38, p1010, 1976.