A Combined Deep-Learning and Deformable-Model Approach to Fully Automatic Segmentation of the Left Ventricle in Cardiac MRI

M.R. Avendi, Arash Kheradvar, Hamid Jafarkhani

PII: S1361-8415(16)00012-8
DOI: 10.1016/j.media.2016.01.005
Reference: MEDIMA 1072

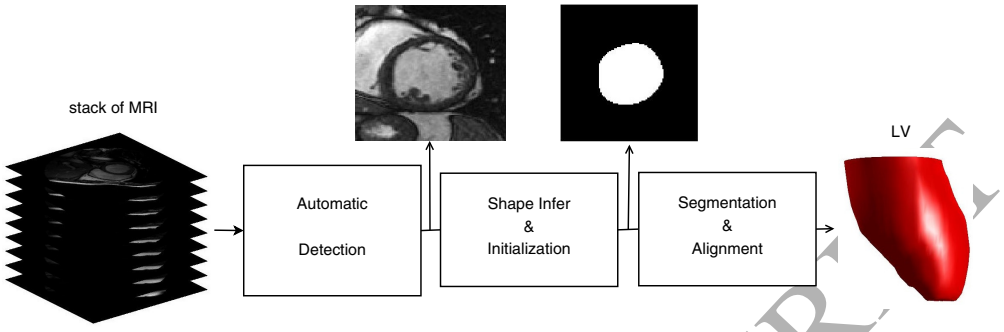
To appear in: *Medical Image Analysis*

Received date: 11 August 2015
Revised date: 3 January 2016
Accepted date: 18 January 2016

Please cite this article as: M.R. Avendi, Arash Kheradvar, Hamid Jafarkhani, A Combined Deep-Learning and Deformable-Model Approach to Fully Automatic Segmentation of the Left Ventricle in Cardiac MRI, *Medical Image Analysis* (2016), doi: 10.1016/j.media.2016.01.005

Highlights

- Deep learning for segmentation
- Excellent agreement
- High correlation for indices



A Combined Deep-Learning and Deformable-Model Approach to Fully Automatic Segmentation of the Left Ventricle in Cardiac MRI

M. R. Avendi[a,b], Arash Kheradvar[b], Hamid Jafarkhani [a,*]

[a] *Center for Pervasive Communications and Computing, University of California, Irvine, USA*

[b] *the Edwards Lifesciences Center for advanced cardiovascular technology, University of California, Irvine, USA*

Abstract

Segmentation of the left ventricle (LV) from cardiac magnetic resonance imaging (MRI) datasets is an essential step for calculation of clinical indices such as ventricular volume and ejection fraction. In this work, we employ deep learning algorithms combined with deformable models to develop and evaluate a fully automatic segmentation tool for the LV from short-axis cardiac MRI datasets. The method employs deep learning algorithms to learn the segmentation task from the ground true data. Convolutional networks are employed to automatically detect the LV chamber in MRI dataset. Stacked autoencoders are utilized to infer the shape of the LV. The inferred shape is incorporated into deformable models to improve the accuracy and robustness of the segmentation. We validated our method using 45 cardiac MR datasets taken from the MICCAI 2009 LV segmentation challenge and showed that it outperforms the state-of-the art methods. Excellent agreement with the ground truth was achieved. Validation metrics, percentage of good contours, Dice metric, average perpendicular distance and conformity, were computed as 96.69%, 0.94, 1.81mm and 0.86, versus those of $79.2\% - 95.62\%$, 0.87-0.9, 1.76-2.97mm and 0.67-0.78, obtained by other methods, respectively.

Keywords: Caridac MRI, LV segmentation, deep learning, machine

*Please address correspondence to H. Jafarkhani

Email addresses: m.avendi@uci.edu (M. R. Avendi), arashkh@uci.edu (Arash Kheradvar), hamidj@uci.edu (Hamid Jafarkhani)

learning, deformable models.

1. Introduction

Cardiac magnetic resonance imaging (MRI) is now routinely being used for the evaluation of the function and structure of the cardiovascular system (Yuan et al., 2002; Lima and Desai, 2004; Frangi et al., 2001; Petitjean and Dacher, 2011; Tavakoli and Amini, 2013; Heimann and Meinzer, 2009; Suinesiaputra et al., 2014). Segmentation of the left ventricle (LV) from cardiac MRI datasets is an essential step for calculation of clinical indices such as ventricular volume, ejection fraction, left ventricular mass and wall thickness as well as analyses of the wall motion abnormalities.

Manual delineation by experts is currently the standard clinical practice for performing the LV segmentation. However, manual segmentation is tedious, time consuming and prone to intra- and inter-observer variability (Frangi et al., 2001; Petitjean and Dacher, 2011; Tavakoli and Amini, 2013; Heimann and Meinzer, 2009; Suinesiaputra et al., 2014). To address this, it is necessary to reproducibly automate this task to accelerate and facilitate the process of diagnosis and follow-up. To date, several methods have been proposed for the automatic segmentation of the LV. A review of these methods can be found in (Frangi et al., 2001; Petitjean and Dacher, 2011; Tavakoli and Amini, 2013; Heimann and Meinzer, 2009; Suinesiaputra et al., 2014).

To summarize, there are several challenges in the automated segmentation of the LV in cardiac MRI datasets: heterogeneities in the brightness of LV cavity due to blood flow; presence of papillary muscles with signal intensities similar to that of the myocardium; complexity in segmenting the apical and basal slice images; partial volume effects in apical slices due to the limited resolution of cardiac MRI; inherent noise associated with cine cardiac MRI; dynamic motion of the heart and inhomogeneity of intensity; considerable variability in shape and intensity of the heart chambers across patients, notably in pathological cases, etc (Tavakoli and Amini, 2013; Petitjean and Dacher, 2011; Queiros et al., 2014). Due to these technical barriers the task of automatic segmentation of the heart chambers from cardiac MR images is still a challenging problem (Petitjean and Dacher, 2011; Tavakoli and Amini, 2013; Suinesiaputra et al., 2014).

Current approaches for automatic segmentation of the heart chambers can be generally classified as: pixel classification (Kedenburg et al., 2006;

Cocosco et al., 2008), image-based methods (Jolly, 2009; Liu et al., 2012), deformable methods (Billet et al., 2009; Ben Ayed et al., 2009; Chang et al., 2010; Pluempitiwiriyawej et al., 2005), active appearance and shape models (AAM/ASM) (Zhang et al., 2010; Assen et al., 2006) and atlas models (Zhuang et al., 2008; Lorenzo-Valdés et al., 2004). Pixel classification, image-based and deformable methods suffer from a low robustness and accuracy and require extensive user interaction (Petitjean and Dacher, 2011). Alternatively, model-based methods such as AAM/ASM and atlas models can overcome the problems with previous methods and reduce user interaction at the expense of a large training set to build a general model. However, it is very difficult to build a model that is general enough to cover all possible shapes and dynamics of the heart chambers (Petitjean and Dacher, 2011; Jolly et al., 2009). Small datasets lead to a large bias in the segmentation, which makes these methods inefficient when the heart shape is outside the learning set (e.g., congenital heart defects, post-surgical remodeling, etc).

Furthermore, current learning-based approaches for LV segmentation have certain limitations. For instance, methods using random forests (Margeta et al., 2012; Lempitsky et al., 2009; Geremia et al., 2011) rely on image intensity and define the segmentation problem as a classification task. These methods employ multiple stages of intensity standardization, estimation and normalization, which are computationally expensive and affect the success of further steps. As such, their performance is rather mediocre at basal and apical slices and overall inferior to the state-of-the-art. Also, methods that use Markov random fields (MRFs) (Cordero-Grande et al., 2011a; Huang et al., 2004), conditional random fields (CRFs) (Cobzas and Schmidt, 2009; Dreijer et al., 2013) and restricted Boltzman machines (RBMs) (Ngo and Carneiro, 2014) have been considered. MRF and RBM are generative models that try to learn the probability of input data. Computing the image probability and parameter estimation in generative models is generally difficult and can lead to reduced performance if oversimplified. Besides, they use the Gibbs sampling algorithm for training, which can be slow, become stuck for correlated inputs, and produce different results each time it is run due to its randomized nature. Alternatively, CRF methods try to model the conditional probability of latent variables, instead of the input data. However, they are still computationally difficult, due to complexity of parameter estimation, and their convergence is not guaranteed (Dreijer et al., 2013).

Motivated by these limitations, and given the fact that manual segmentation by experts is the ground truth in cardiac MRI, we tackle the complex

problem of LV segmentation utilizing a combined deep-learning (LeCun et al., 2015; Hinton and Salakhutdinov, 2006; Bengio, 2009; Bengio et al., 2013; Ng, accessed July., 2015; Deng and Yu, 2014; Baldi, 2012) and deformable-models approach. We develop and validate a fully automated, accurate and robust segmentation method for the LV in cardiac MRI. In terms of novelty and contributions, our work is one of the early attempts of employing deep learning algorithms for cardiac MRI segmentation. It is generally believed that since current practices of deep learning have been trained on huge amount of data, deep learning cannot be effectively utilized for medical image segmentation due to the lack of training data. However, we show that even with limited amount of training data, using artificial data enlargement, pre-training and careful design, deep learning algorithms can be successfully trained and employed for cardiac MRI segmentation. Nevertheless, we solve some of the shortcomings of classical deformable models, i.e., shrinkage and leakage and sensitivity to initialization, using our integrated approach. Furthermore, we introduce a new curvature estimation method using quadrature polynomials to correct occasional misalignment between image slices. The proposed framework is tested and validated on the MICCAI database (Radau et al., 2009). Finally, we provide better performance in terms of multiple evaluation metrics and clinical indices.

The remainder of the manuscript is as follows. In Section 2, the proposed method is described in detail. In Section 3, the implementation details are provided. Section 4 presents the validation experiments. The results are presented in Section 5. In Section 6 we discuss the results, performance and comparison with the state-of-the-arts methods. Section 7 concludes the paper.

2. Materials and Methods

2.1. Datasets

The MICCAI 2009 challenge database (Radau et al., 2009) is used in our study to train and assess the performance of the proposed methodology. The MICCAI database was obtained from the Sunnybrook Health Sciences Center, Toronto, Canada. The database is publicly available online (Radau et al., 2009) and contains 45 MRI datasets, grouped into three datasets. Each dataset contains 15 cases, divided into four ischemic heart failure cases (SC-HF-I), four non-ischemic heart failure cases (SC-HF-NI), four LV hypertrophy cases (SC-HYP) and three normal (SC-N) cases. Manual segmenta-

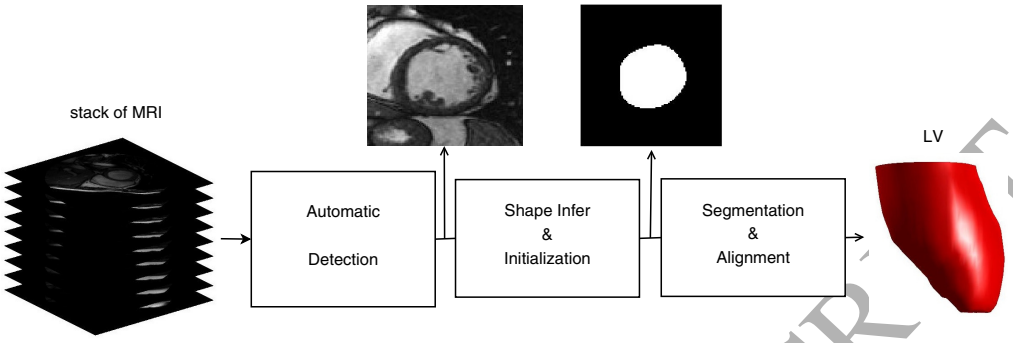


Figure 1: Block diagram of the developed algorithm.

tion of images by experts at the end diastole (ED) and the end systole (ES) cardiac phases is included in the database. A typical dataset contains 20 frames in 6-12 short-axis slices obtained from the base to the apex. Image parameters are: thickness=8 mm, image size = 256×256 pixels.

The training dataset of the MICCAI database (Radau et al., 2009) was used to train our method. The validation and online datasets were used for evaluation of the method.

2.2. Method

The block diagram of the proposed method is depicted in Fig. 1. A stack of short-axis cardiac MR images is provided as the input (Fig. 1). The method is carried out in three stages: (i) the region of interest (ROI) containing the LV is determined in the raw input images using convolutional networks (LeCun et al., 2010; Szegedy et al., 2013; Sermanet et al., 2014; Krizhevsky et al., 2012) trained to locate the LV; (ii) the shape of the LV is inferred using stacked autoencoders (Bengio et al., 2013; Bengio, 2009; Vincent et al., 2008; Baldi, 2012; Deng and Yu, 2014; Vincent et al., 2010) trained to delineate the LV; (iii) the inferred shape is used for initialization and also is incorporated into deformable models for segmentation. Contour alignment is performed to reduce misalignment between slices for 3D reconstruction. Each stage of the block diagram is individually trained during an offline training process to obtain its optimum values of parameters. After training, we deploy the system to perform the automatic segmentation task. The three stages are further elaborated as follows:

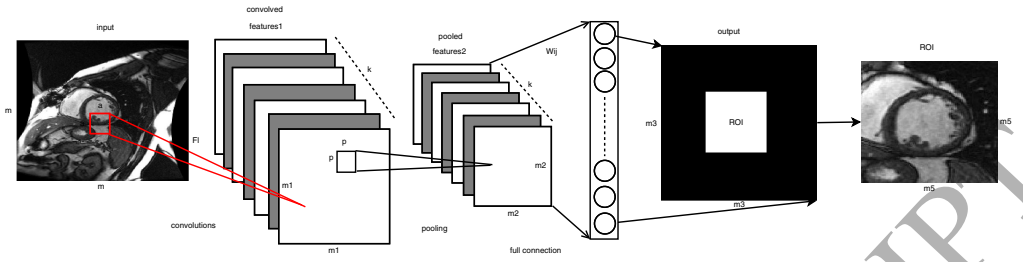


Figure 2: Block diagram of automatic detection of LV in MRI dataset.

2.2.1. Automatic Detection

The raw cardiac MRI datasets usually include the heart and its surrounding tissues within the thoracic cavity. To reduce the computational complexity and time, and improve the accuracy, the first step of the algorithm is to locate the LV and compute a ROI around it.

A block diagram of the automatic LV detection developed using convolutional networks is illustrated in Fig. 2. To reduce complexity, the original image size of 256×256 is down-sampled to 64×64 and used as the input.

Then, filters $\mathbf{F}_l \in \mathcal{R}^{11\times 11}, \mathbf{b}_0 \in \mathcal{R}^{100}$ are convolved with the input image to obtain the convolved feature maps. Denote the gray-value input image $\mathbf{I} : \Omega \rightarrow \mathcal{R}, \Omega \subset \mathcal{R}^2$ with size 64×64. $\mathbf{I}[i, j]$ represents a pixel intensity at coordinate $[i, j]$ of the image. Note that the pixel coordinates at the top left and bottom right of the image are $[1, 1]$ and $[64, 64]$, respectively. The convolved features are computed as $\mathbf{C}_l[i, j] = f\left(\mathbf{Z}_l[i, j]\right)$ where

$$\mathbf{Z}_l[i, j] = \sum_{k_1=1}^{11}\sum_{k_2=1}^{11} \mathbf{F}_l[k_1, k_2]\mathbf{I}[i + k_1 - 1, j + k_2 - 1] + \mathbf{b}_0[l], \tag{1}$$

for $1 \leq i, j \leq 54$ and $l = 1, \cdots, 100$. This results in 100 convolved features $\mathbf{Z}_l \in \mathcal{R}^{54\times 54}$. Here, $\mathbf{x}[i]$ denotes the i-th element of vector $\mathbf{x}$ and $\mathbf{X}[i, j]$ denotes the element at the i-th row and the j-th column of matrix $\mathbf{X}$.

Next, the convolved feature maps are sub-sampled using average pooling (Boureau et al., 2010). To this end, the average values over non-overlapping neighborhoods with size 6×6 are computed in each feature map as

$$\mathbf{P}_l[i_1, j_1] = \frac{1}{6} \sum_{i=(6i_1-5)}^{6i_1} \sum_{j=(6j_1-5)}^{6j_1} \mathbf{C}_l[i, j], \tag{2}$$

for $1 \leq i_1, j_1 \leq 9$. This results in 100 reduced-resolution features $\mathbf{P}_l \in \mathcal{R}^{9\times 9}$ for $l = 1, \cdots, 100$.

Finally, the pooled features are unrolled as vector $\mathbf{p} \in \mathcal{R}^{8100}$ and fully connected to a logistic regression layer with 1024 outputs to generate a mask of size 32×32 that specifies the ROI. The output layer computes $\mathbf{y}_c = f(\mathbf{W}_1\mathbf{p} + \mathbf{b}_1)$, where $\mathbf{W}_1 \in \mathcal{R}^{1024\times 8100}$ and $\mathbf{b}_1 \in \mathcal{R}^{1024}$ are trainable matrices. Note that the original MR image size is 256×256. Therefore, first, the output mask is up-sampled from 32×32 to the original MR image size. The center of the mask is then computed and used to crop a ROI of size 100×100 from the original image for further processing in the next stage.

Before using the network for localizing the LV, it should be trained. During training, the optimum parameters of the network $(\mathbf{F}_l, \mathbf{b}_0, \mathbf{W}_1, \mathbf{b}_1)$ are obtained as described in the next section.

Training Convolutional Network

Training the convolution network involves obtaining the optimum values of filters $\mathbf{F}_l, l = 1, \cdots, 100$ as well as other parameters $\mathbf{b}_0, \mathbf{W}_1, \mathbf{b}_1$. In common convolutional networks a large training set is usually available. Therefore, they initialize the filters $(\mathbf{F}_l)$ randomly and then train the convolutional network. The filters are constructed simultaneously during training. In our application, the number of training and labeled data is limited. As such, instead of random initialization, the filters are obtained using a sparse autoencoder (AE) which acts as a pre-training step. This leads us to train the network with the limited amount of data that we have while avoid overfitting.

We employ an AE with 121 input/output units and 100 hidden units as depicted in Fig. 3. To train the AE, $N_1 \approx 10^4$ small patches of size 11×11 are randomly selected from the raw input images of the training dataset. Each patch is then unrolled as vector $\mathbf{x}^{(i)} \in R^{121}, i = 1, \cdots, N_1$ and fed to the input layer of the AE. Denote the weights between the input layer and the hidden layer with $\mathbf{W}_1 \in \mathcal{R}^{100\times 121}$ and the weights between the hidden layer and output layer with $\mathbf{W}_2 \in R^{121\times 100}$. The hidden layer computes $\mathbf{a}_2^{(i)} = f(\mathbf{W}_2\mathbf{x}^{(i)} + \mathbf{b}_2)$ and the final output is $\mathbf{y}^{(i)} = f(\mathbf{W}_3\mathbf{a}_2^{(i)} + \mathbf{b}_3)$, where $f(x) = 1/(1 + e^{-x})$ is the sigmoid activation function and $\mathbf{b}_2 \in \mathcal{R}^{100}, \mathbf{b}_3 \in \mathcal{R}^{121}$ are trainable bias vectors. The task of AE is to construct $\mathbf{x}^{(i)}$ at the output from the hidden values. Thus, input values are used as the labeled data and no actual labeled data are required for training the AE.

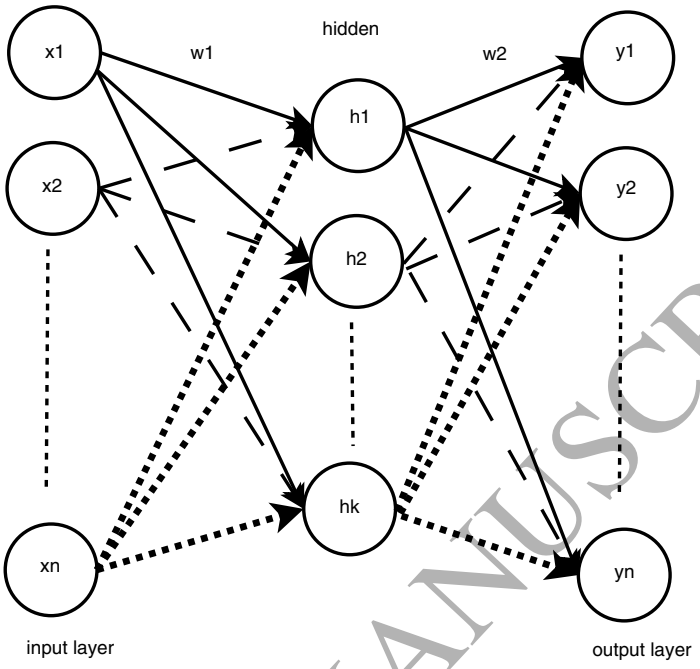


Figure 3: Sparse autoencoder is trained to initialize filters ($\mathbf{F}_l$).

The AE is optimized by minimizing the cost function

$$J(\mathbf{W}_2, \mathbf{b}_2) = \frac{1}{2N_1} \sum_{i=1}^{N_1} |\mathbf{y}^{(i)} - \mathbf{x}^{(i)}|^2 + \frac{\lambda}{2} \left(\|\mathbf{W}_2\|^2 + \|\mathbf{W}_3\|^2 \right) + \beta \sum_{j=1}^{k} \mathrm{KL}(\rho || \hat{\rho}_j). \tag{3}$$

Here, the first term computes the average squared-error between the final output $\mathbf{y}^{(i)}$ and the desired output $\mathbf{x}^{(i)}$. Furthermore, to avoid overfitting, the l_2 regularization/weight decay term is added to the cost function to decrease the magnitude of the weights. Also, to learn higher representation from the input data, a sparsity constraint is imposed on the hidden units. In this way, a sparse AE is built. Here, the Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951) constrains the mean value of the activations of the hidden layer $\hat{\rho}_j = (1/N_1) \sum_{i=1}^{N_1} \mathbf{a}_2^{(i)}[j],\ j = 1, \cdots, 100$, to be equal to the sparsity parameter ρ, which is usually a small value. The weight decay coefficient λ and the sparsity coefficient β control the relative importance of the three terms in the cost function. The optimization parameters are set as

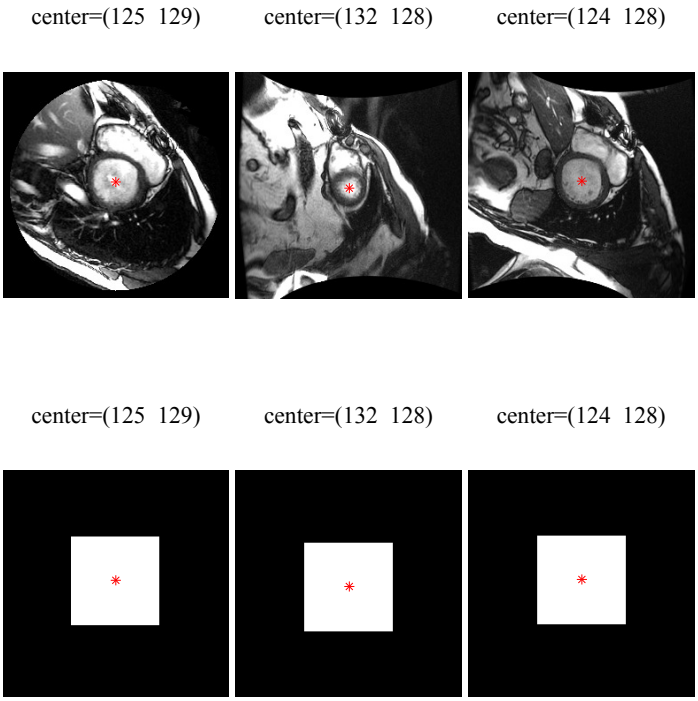


Figure 4: Typical input images (top) and corresponding binary masks (bottom) used for training of the automatic detection network. Note, the center of image (top) is the same as the center of corresponding ROI (bottom).

$\lambda = 10^{-4}, \rho = 0.1$ and $\beta = 3$. Once autoencodr is trained, $\mathbf{W}_2$ is configured as 100 filters $\mathbf{F}_l \in \mathcal{R}^{11\times 11}, l = 1, \cdots, 100$ and $\mathbf{b}_0 = \mathbf{b}_2$ for the next step.

Then, we perform a feed-forward computation using Eqs. 1-2 until the output layer. Next, the output layer is pre-trained by minimizing the cost function

$$J(\mathbf{W}_1, \mathbf{b}_1) = \frac{1}{2N_2} \sum_{i=1}^{N_2} |\mathbf{y}_{\mathrm{c}}^{(i)} - \mathbf{l}_{\mathrm{roi}}^{(i)}|^2 + \frac{\lambda}{2} \left(\|\mathbf{W}_1\|^2 \right), \tag{4}$$

where $\mathbf{l}_{\mathrm{roi}}^{(i)} \in \mathcal{R}^{1024}$ is the labeled data corresponding to the ith input image and N_2 is the number of training data. The labeled data at the output layer are binary masks, as shown in Fig. 4, generated based on manual training contours. As seen, a binary mask is an image with black background and

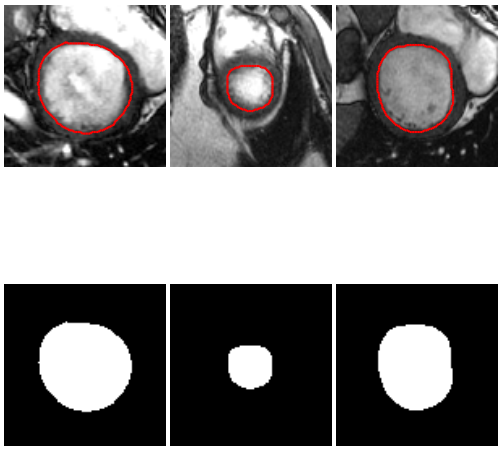


Figure 5: Typical sub-images with manual segmentation of LV (top) and corresponding binary masks (bottom) used for training the stacked autoencoder.

a white foreground corresponding to the ROI. The foreground is centered at the center of the LV contour, which is known from the training manual contours. Note that the binary mask is down-sampled to 32×32 and then unrolled as vector $\mathbf{l}_{\text{roi}}^{(i)}$ to be used for training.

Finally, the whole network is fine-tuned by minimizing the cost function

$$J(\mathbf{F}_l, \mathbf{b}_0, \mathbf{W}_1, \mathbf{b}_1) = \frac{1}{2N_2} \sum_{i=1}^{N_2} |\mathbf{y}_{\text{c}}^{(i)} - \mathbf{l}_{\text{roi}}^{(i)}|^2 + \frac{\lambda}{2} \left(\|\mathbf{W}_1\|^2 + \sum_{l=1}^{100} \|\mathbf{F}_l\|^2 \right) \quad (5)$$

The cost function can be minimized using the backpropagation algorithm. Here, $\lambda = 10^{-4}$. It should be mentioned that the training process is performed only once.

2.2.2. Shape Inferring

We utilize and train a stacked-AE, depicted in Fig. 6, to infer the shape of the LV. The stacked-AE has one input layer, two hidden layers, and one output layer. The sub-image obtained from the previous block is sub-sampled and unrolled as vector $\mathbf{x}_{\text{s}} \in \mathcal{R}^{4096}$ and fed to the input layer. The hidden layers build the abstract representations by computing $\mathbf{h}_1 = f(\mathbf{W}_4\mathbf{x}_{\text{s}} + \mathbf{b}_4)$

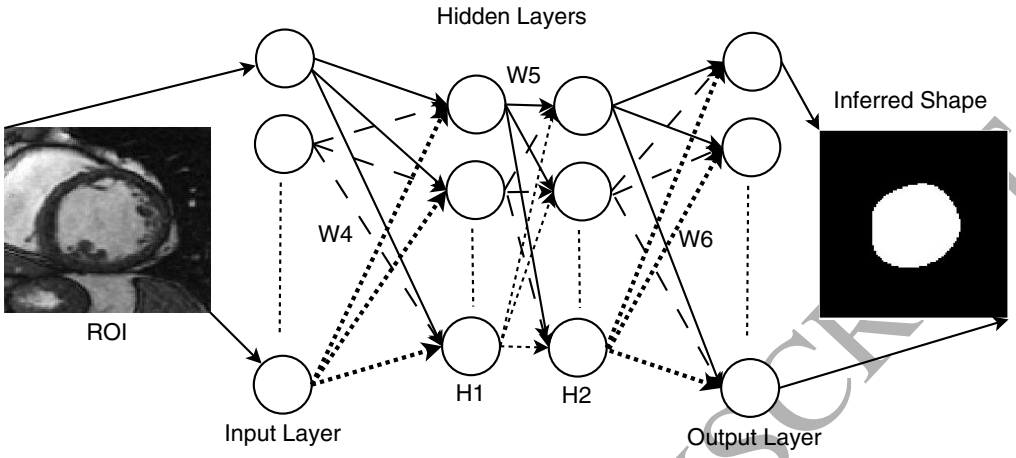


Figure 6: Stacked AE for inferring the shape of LV. The input is a sub-image and the output is a binary mask.

and $\mathbf{h}_2 = f(\mathbf{W}_5\mathbf{h}_1 + \mathbf{b}_5)$. The output layer computes $\mathbf{y}_s = f(\mathbf{W}_6\mathbf{h}_2 + \mathbf{b}_6)$ to produce a binary mask. The binary mask is black (zero) everywhere except at the borders of the LV. Here, $\mathbf{W}_4 \in \mathcal{R}^{100\times4096}$, $\mathbf{b}_4 \in \mathcal{R}^{100}$,$\mathbf{W}_5 \in \mathcal{R}^{100\times100}$, $\mathbf{b}_5 \in \mathcal{R}^{100}$ and $\mathbf{W}_6 \in \mathcal{R}^{4096\times100}$, $\mathbf{b}_6 \in \mathcal{R}^{4096}$ are trainable matrices and vectors that are obtained during the training process, as detailed in the next section.

Training stacked-AE

The training of the stacked-AE is performed in two steps: pre-training and fine-tuning. Since the amount of labeled data is limited in our application, a layer-wise pre-training is performed. The layer-wise pre-training helps to prevent overfitting, leading to a better generalization. During the pre-training step, parameters $\mathbf{W}_4, \mathbf{W}_5$ of the stacked-AE are obtained layer by layer with no labeled data. Parameter $\mathbf{W}_6$ of the stacked-AE is obtained using the labeled data. The details are as follows.

First, the input layer and the hidden layer H_1 are departed from the stacked-AE. By adding an output layer with the same size as the input layer to the two departed layers (input layer and H_1) a sparse AE is constructed (similar to Fig. 3). The sparse AE is trained in an unsupervised fashion as explained in 2.2.1 to obtain $\mathbf{W}_4$. The optimization parameters are set as $\lambda = 3 \times 10^{-3}, \rho = 0.1, \beta = 3$.

The training input/output data of the sparse AE are sub-images of size 100×100 centered at the LV extracted from the full-size training images.

The input image is resized to 64×64 to be compatible with the input size 4096 of the stacked-AE. Once training of the first sparse AE is complete, its output layer is discarded. The hidden units' outputs in the AE are now used as the input for the next hidden layer H_2 in Fig. 6.

Then, hidden layers H_1 and H_2 are departed from the stacked-AE to construct another sparse AE. Similarly, the second sparse AE is trained to obtain $\mathbf{W}_5$. Again, no labeled data is required. This step can be repeated depending on the number of hidden layers.

The last hidden layer's outputs are used as the input to the final layer, which is trained in a supervised fashion to obtain $\mathbf{W}_6$. The cost function to train the final layer computes

$$J(\mathbf{W}_6, \mathbf{b}_6) = \frac{1}{2N_2} \sum_{i=1}^{N_2} |\mathbf{y}_{\mathrm{s}}^{(i)} - \mathbf{l}_{\mathrm{lv}}^{(i)}|^2 + \frac{\lambda}{2} \|\mathbf{W}_6\|^2, \tag{6}$$

where $\mathbf{l}_{\mathrm{lv}}^{(i)} \in \mathcal{R}^{4096}$ is the labeled data corresponding to the ith image. The labeled data are binary masks created from manual segmentations drawn by experts. Fig. 5 depicts three examples of input images and corresponding labels used for training of the stacked-AE. Note that the binary mask is unrolled as vector $\mathbf{l}_{\mathrm{lv}}$ to be used during optimization.

It should be mentioned that the layer-wise pre-training results in proper initial values for parameters $\mathbf{W}_4, \mathbf{W}_5, \mathbf{W}_6$. In the second step, the whole architecture is fine-tuned by minimizing the cost function

$$J(\mathbf{W}_4, \mathbf{W}_5, \mathbf{W}_6, \mathbf{b}_4, \mathbf{b}_5, \mathbf{b}_6) = \frac{1}{2N_2} \sum_{i=1}^{N_2} |\mathbf{y}_{\mathrm{s}}^{(i)} - \mathbf{l}_{\mathrm{lv}}^{(i)}|^2 + \frac{\lambda}{2} \left(\|\mathbf{W}_4\|^2 + \|\mathbf{W}_5\|^2 + \|\mathbf{W}_6\|^2 \right), \tag{7}$$

using the back-propagation algorithm with respect to the supervised criterion. Here $\lambda = 10^{-4}$. As in the case of automatic detection the training process is performed only once.

2.2.3. Segmentation and Alignment

The final block employs a deformable model combined with the inferred shape for accurate segmentation. Deformable models are dynamic contours that evolve by minimizing an energy function. The energy function reaches its minimum when the contour lies on the boundary of the object of interest.

In most of conventional deformable methods, the output contours tend to shrink inward or leak outward due to presence of the papillary muscles in the LV and small contrast between surrounding tissues. We solve these issues by using the inferred shape from the previous stage as a good initialization. In addition, the shape is incorporated into the energy function to prevent the contour from shrinkage/leakage.

Denote the input sub-image with $I_{\mathrm{s}} : \Omega_{\mathrm{s}} \rightarrow \mathcal{R}, \Omega_{\mathrm{s}} \subset \Omega \subset \mathcal{R}^2$ and the coordinate of image pixels with (x, y). Let us define $\phi(x, y)$ as the level set function that returns negative values for the pixels inside a contour and positive values for the pixels outside. Also, denote the level set function corresponding to the inferred shape with $\phi_{\mathrm{shape}}(x, y)$. The energy function is defined as

$$E(\phi) = \alpha_1 E_{\mathrm{len}}(\phi) + \alpha_2 E_{\mathrm{reg}}(\phi) + \alpha_3 E_{\mathrm{shape}}(\phi), \tag{8}$$

which is a combination of the length-based energy function (Chan and Vese, 2001)

$$E_{\mathrm{len}}(\phi) = \int_{\Omega_{\mathrm{s}}} \delta(\phi) |\nabla \phi| \mathrm{d}x\mathrm{d}y, \tag{9}$$

region-based (Chan and Vese, 2001)

$$E_{\mathrm{reg}}(\phi) = \int_{\Omega_{\mathrm{s}}} |I_{\mathrm{s}} - c_1|^2 H(\phi) \mathrm{d}x\mathrm{d}y + \int_{\Omega_{\mathrm{s}}} |I_{\mathrm{s}} - c_2|^2 (1 - H(\phi)) \mathrm{d}x\mathrm{d}y, \tag{10}$$

and prior shape energy terms

$$E_{\mathrm{shape}}(\phi) = \int_{\Omega_{\mathrm{s}}} (\phi - \phi_{\mathrm{shape}})^2 \mathrm{d}x\mathrm{d}y. \tag{11}$$

Here, $\delta(\phi)$, $H(\phi)$ and $\nabla(\cdot)$ are the delta function, Heaviside step function and the gradient operation, respectively. Also c_1 and c_2 are the average of the input image I_{s} outside and inside the contour (Chan and Vese, 2001), respectively. The α_i's, $i = 1, \cdots, 3$ are the combining parameters, which were determined empirically during training as $\alpha_1 = 1, \alpha_2 = 0.5, \alpha_3 = 0.25$. In other words, given images and labels of the training dataset, the outcome of the combined deformable model was compared with the corresponding label and parameters $\alpha_1, \alpha_2, \alpha_3$ were tweaked to obtain the best evaluation metrics.

The deformable method seeks a unique contour denoted by C^* (or equivalently ϕ^*), which lies on the boundary of the object of interest. This is obtained by minimizing the energy function over ϕ as:

$$\phi^* = \arg\min_{\phi}\{E(\phi)\}, \tag{12}$$

that can be solved using the gradient descent algorithm. By letting ϕ be a function of time and using the Euler-Lagrange equation (Chan and Vese, 2001; Pluempitiwiriyawej et al., 2005), we obtain

$$\frac{\mathrm{d}\phi}{\mathrm{d}t} = -\frac{\mathrm{d}E}{\mathrm{d}\phi} = \delta(\phi)\left[\alpha_1 \mathrm{Div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) + \alpha_2(I_\mathrm{s} - c_2)^2 - \alpha_2(I_\mathrm{s} - c_1)^2 - 2\alpha_3(\phi - \phi_\mathrm{shape})\right], \tag{13}$$

where $\mathrm{Div}(\cdot)$ is the divergence operator.

The gradient descent starts with an initialization of $\phi^{(0)}$ obtained from the inferred shapes and is updated iteratively

$$\phi^{(k+1)} = \phi^{(k)} + \gamma\frac{\mathrm{d}\phi}{\mathrm{d}t}, \tag{14}$$

to obtain the final ϕ^* or contour C^*. Here, γ is the step size which is a small number. The stopping criterion checks whether the solution is stationary or not by computing the difference between the length of the contours in the current and previous iterations.

In case of 3D reconstruction of cardiac chambers, it is necessary to consider possible misalignment between the image slices. Misalignment occurs in cardiac MRI mainly due to respiratory and patient motions during MRI scans. Ignoring misalignment leads to jagged discontinuous surfaces in the reconstructed volume. To deal with this issue, we introduce a misalignment estimation and correction using quadratic polynomials.

To this end, the center coordinate of the LV contours is computed from the obtained LV segmentation in all image slices, denoted as $(\tilde{x}_i, \tilde{y}_i)$, for $i = 1, \cdots, n$, where n is the number of slices. Let us denote the actual center coordinate of the ith contour with (x_i, y_i). Then we can write

$$\tilde{x}_i = x_i + w_i, \tag{15}$$

$$\tilde{y}_i = y_i + v_i, \tag{16}$$

where $w_i \sim \mathcal{N}(0, \sigma_w^2)$, $v_i \sim \mathcal{N}(0, \sigma_v^2)$ are the misalignment values due to motion artifacts, modeled by independent Gaussian random variables.

Using quadratic assumption for the curvature, it follows that

$$x_i = a_1 i^2 + b_1 i + c_1, \tag{17}$$
$$y_i = a_2 i^2 + b_2 i + c_2. \tag{18}$$

Here $a_1, b_1, c_1, a_2, b_2, c_2$ are unknown parameters that are estimated by minimizing the mean squared error as

$$\hat{a}_1, \hat{b}_1, \hat{c}_1 = \arg \min_{a_1, b_1, c_1} \sum_{i=1}^{n} (\tilde{x}_i - a_1 i^2 - b_1 i - c_1)^2, \tag{19}$$

$$\hat{a}_2, \hat{b}_2, \hat{c}_2 = \arg \min_{a_2, b_2, c_2} \sum_{i=1}^{n} (\tilde{y}_i - a_2 i^2 - b_2 i - c_2)^2. \tag{20}$$

After estimating the unknown parameters, the actual center coordinates are estimated from Eqs. (17)-(18). Finally, the contours are registered, using an affine transformation with linear interpolation, according to the estimated center values to obtain an aligned stack of contours. Fig. 7 illustrates the centers of LV contours from the base to the apex for a typical MRI dataset with ten misaligned image slices. The estimated aligned centers using quadratic polynomials are depicted in red in the figure.

3. Implementation Details

Images and contours of all the cases in the training dataset of the MICCAI challenge (Radau et al., 2009) were collected and divided into the large-contour and small-contour groups. Typically, the large contours belong to image slices near the base/middle and the small contours belong to the apex of the heart since the contours near the apex of the heart are much smaller than the contours at the base. As such, there are around 135 and 125 images in each group, respectively. Then, we artificially enlarged the training dataset using techniques such as image translation, rotation and changing the pixel intensities based on the standard principal component analysis (PCA) technique as explained in (Koikkalainen et al., 2008). Using these techniques, we augmented the training dataset by a factor of ten. Eventually, we had around 1350 and 1250 images/labels in each group, respectively. Then, we

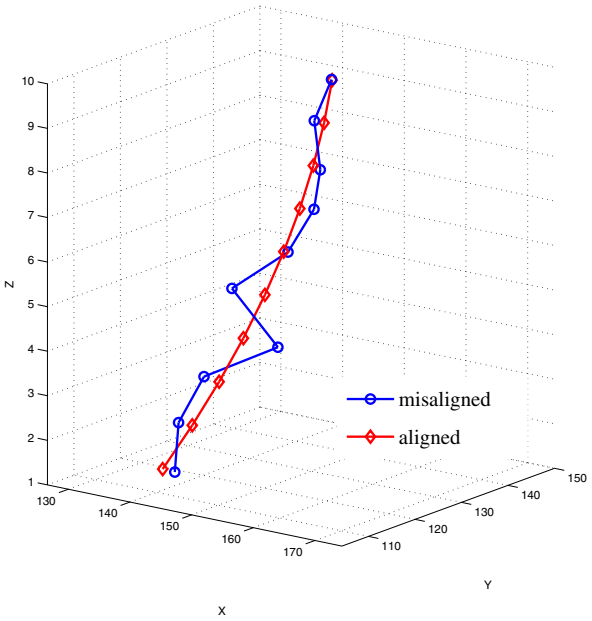


Figure 7: Misaligned centers of LV contours from the base to the apex (blue) and corresponding aligned centers (red) obtained from quadrature curve fitting for a typical MRI dataset with ten image slices.

built and trained two networks, one for the large-contour dataset and one for the small-contour dataset.

It is noted that considerable overfitting may happen in deep learning networks, due to the large number of parameters to be learned. We paid great attention to prevent the overfitting problem in our networks. To deal with this, we adopted multiple techniques including: layer-wise pre-training, l_2 regularization and sparsity constraints as explained in Sections 2.2.1 and 2.2.2. Although the lack of training data was a challenge, the use of layer-wise pre-training was greatly helpful. We also kept the number of units in the hidden layers small and did not go beyond three layers to ensure that the number of parameters is tractable. Furthermore, during the training process, we performed cross-validation by dividing the training dataset into 12 subjects for training and 3 subjects for validation, as well as early stopping

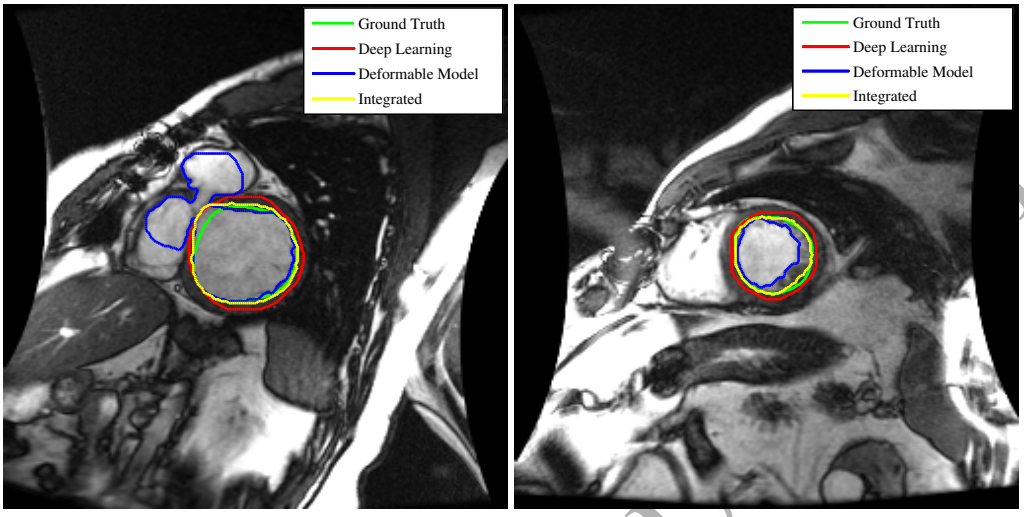


Figure 8: Outcomes of deformable model with no shape constraint ($\alpha_3 = 0$), deep learning (shape inference, step 2) and integrated deep learning and deformable model (final step), for two typical images.

to monitor and prevent overfitting. In addition, we artificially enlarged the training dataset as mentioned earlier in this section. The hyper-parameters of the networks, i.e., number of layers and units, number of filters, filter and pooling sizes, etc., are determined empirically during the cross-validation process.

In the current study, our method was developed in MATLAB 2014a, performed on a Dell Precision T7610 workstation, with Intel(R) Xeon(R) CPU 2.6 GHz, 32 GB RAM, 64-bit Windows 7. The method was trained using the training dataset and tested on the online and validation datasets of the MICCAI database (Radau et al., 2009).

4. Validation Process

We assess the performance of the proposed methodology by evaluating the accuracy of the proposed automated segmentation method compared with the gold standard (manual annotations by experts). To this end, the following measures are computed as: average perpendicular distance (APD), Dice metric, Hausdorff distance, percentage of good contours and the conformity coefficient (Chang et al., 2009). As recommended in (Radau et al., 2009), a segmentation is classified as good if the APD is less than 5mm. The

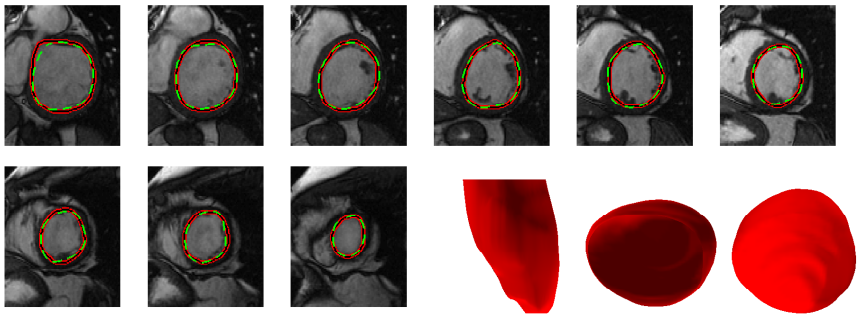


Figure 9: Automatic (red-black) and manual (dashed green) segmentation results of LV for an example cardiac MRI dataset of the MICCAI database (Radau et al., 2009) in 2D and 3D (right) representations.

average perpendicular distance measures the distance from the automatically segmented contour to the corresponding manually drawn expert contour, averaged over all contour points (Radau et al., 2009). A high value implies that the two contours do not match closely (Radau et al., 2009). Also, the Dice metric, DM $= 2(A_{\mathrm{am}})/(A_{\mathrm{a}} + A_{\mathrm{m}})$, is a measure of contour overlap utilizing the contour areas automatically segmented A_{a}, manually segmented A_{m}, and their intersection A_{am} (Radau et al., 2009). The Dice metric is always between zero and one, with higher DM indicating better match between automated and manual segmentations. The Hausdorff distance measures the maximum perpendicular distance between the automatic and manual contours (Queiros et al., 2014; Babalola et al., 2008). Finally, the conformity coefficient measures the ratio of the number of mis-segmented pixels to the number of correctly segmented pixels defined as CC=(3DM-2)/DM (Chang et al., 2009).

In addition, three clinical parameters, end-diastolic volume (EDV), end-systolic volume (ESV) and ejection fraction (EF) were computed using the automatic and manual LV segmentation results and used for the correlation and Bland-Altman analyses (Bland and Altman, 1986). The correlation analysis was performed using the Pearsons test to obtain the slope and intercept equation and the Pearson R-values. To assess the intra- and inter-observer variability the coefficient of variation (CV), defined as the standard deviation (SD) of the differences between the automatic and manual results divided by their mean values, and the reproducibility coefficient (RPC), defined as

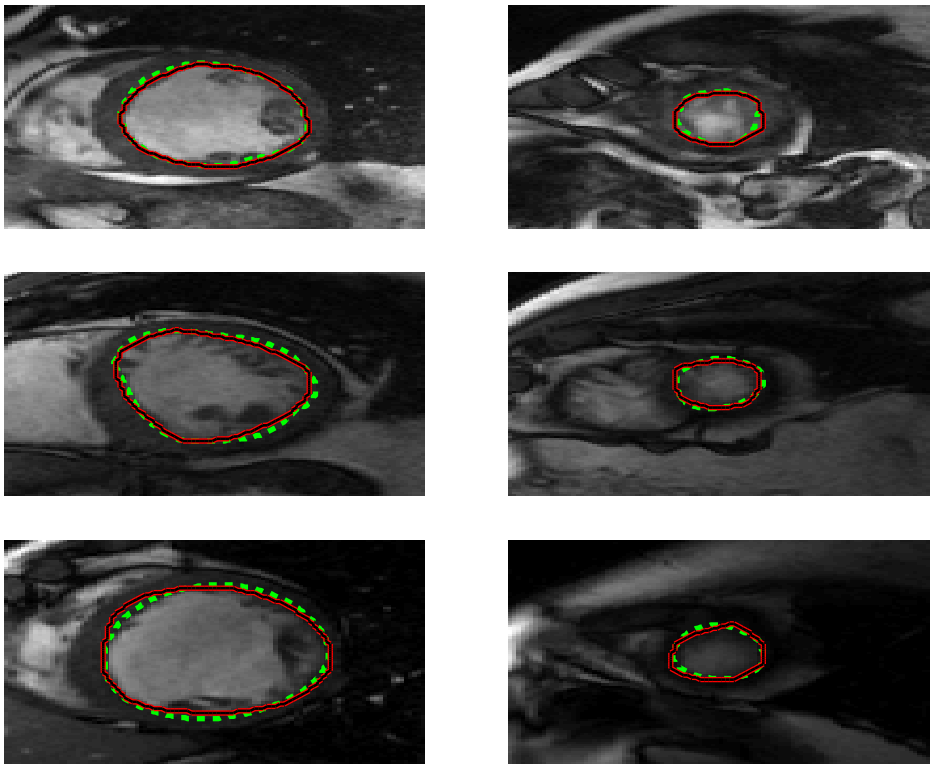


Figure 10: Automatic (red-black) and manual (green) segmentation results for challenging images at the apex (right) and mid LV (left) with presence of papillary muscles for three typical cardiac MRI datasets of the MICCAI database (Radau et al., 2009).

1.96∗SD, are computed.

The segmentation performance was assessed against reference contours using the evaluation code provided in (Radau et al., 2009). Each measure is computed slice by slice and a mean value and standard deviation for all slices of a dataset are calculated.

5. Results

5.1. Illustrative Results

To better understand the role of each step, the outcome of the deformable model with no shape constraint ($\alpha_3 = 0$), deep learning (shape inference, Step

Table 1: Evaluation metrics of our proposed method for the validation and online datasets of the MICCAI database (Radau et al., 2009). Numbers' format: mean value (standard deviation).

Dataset	I/F[3]	Good Contours (%)	Dice Metric	APD[1](mm)	HD[2](mm)	Conformity
Validation	I	90(10)	0.90(0.1)	2.84(0.29)	3.29(0.59)	0.78(0.03)
Validation	F	97.8(4.7)	0.94(0.02)	1.7(0.37)	3.29(0.59)	086(0.04)
Online	I	87(12)	0.89(0.03)	2.95(0.54)	4.64(0.76)	0.76(0.07)
Online	F	95.58(6.7)	0.93(0.02)	1.92(0.51)	3.62(1.1)	0.85(0.05)

[1] Average Perpendicular Distance (APD).
[2] Hausdorff Distance (HD).
[3] (I): Initial contour, (F) Final contour.

2) and the integrated deformable model and deep learning method (final step) for two typical images are shown in Fig. 8.

Fig. 9 illustrates the automatic and manual segmentation results of the LV for a typical cardiac MRI dataset from the base to the apex and three views of the reconstructed LV chamber (front, base and apex views). Also, segmentation results for image slices at the apex and mid LV, which are generally complex due to presence of papillary muscles and low resolution, are depicted in Fig. 10. In the figures, automatic segmentation results are shown in red. The ground truth manual segmentations drawn by experts are shown in green for comparison. Automatic and manual segmentation results for multiple datasets of the MICCAI database (Radau et al., 2009) are illustrated in Fig. 11. In the figure, each row corresponds to one patient/dataset which includes normal subjects (SC-N) and the ones with ischemic heart failure (SC-HF-I), non-ischemic heart failure (SC-HF-NI) and LV hypertrophy (SC-HYP).

5.2. Quantitative Results

In Table 1, the average values and the standard deviation of the computed metrics are listed for the validation and online datasets. For each dataset, two rows of results, corresponding to the initial contour (I) obtained from the inferred shape and the final contour (F), are listed. Table 2 presents a comparison of results between our method with the state-of-the-art methods that used the same database.

Moreover, Figs. 12-14 illustrate the correlation graphs (left) between the

Table 2: Comparison of segmentation performance between proposed method and state-of-the-art techniques using the MICCAI database (Radau et al., 2009). Numbers format: mean value (standard deviation).

Method	#[1]	Good Contours(%)	Dice Metric	APD[2](mm)	Conformity
Proposed	30	**96.69(5.7)**	**0.94(0.02)**	**1.81(0.44)**	**0.86**
(Queiros et al., 2014)	45	92.7(9.5)	0.9(0.05)	1.76(0.45)	0.78
(Ngo and Carneiro, 2014)	30	93.23(9.84)	0.89(0.03)	2.26(0.46)	0.75
(Hu et al., 2013)	45	91.06(9.4)	0.89(0.03)	2.24(0.4)	0.75
(Constantinides et al., 2012)	45	80(16)	0.86(0.05)	2.44(0.56)	0.67
(Liu et al., 2012)	45	91.17(8.5)	0.88(0.03)	2.36(0.39)	0.73
(Huang et al., 2011)	45	79.2(19)	0.89(0.04)	2.16(0.46)	0.75
(Schaerer et al., 2010)	45	–	0.87(0.04)	2.97(0.38)	0.70
(Jolly, 2009)	30	95.62(8.8)	0.88(0.04)	2.26(0.59)	0.73

[1] Number of datasets evaluated. 30 -validation and online datasets, 45- all datasets.
[2] Average Perpendicular Distance (APD).

automatic and manual results and the Bland-Altman graphs (right) of the differences, using the validation dataset, for EDV, ESV and EF, respectively. A correlation with the ground truth contours of 0.99, 0.99, 0.99 for EDV, ESV and EF was measured. The level of agreement between the automatic and manual results was represented by the interval of the percentage difference between mean±1.96SD. The mean and confidence interval of the difference between the automatic and manual EDV results were -13 cm^3 and (-36cm^3 to 10cm^3), respectively. The CV and RPC were 6.9% and 13%, respectively. The mean and confidence interval of difference between the automatic and manual ESV results were -3.5 cm^3, (-18cm^3 to 11cm^3) and CV=6.9%, RPC=14%. Also, the mean and confidence interval of the difference between the automatic and manual EF results were -2.4%, (-8% to 3.2%), CV=6.7%, RPC=13%.

Approximated elapsed times of the training process were as follows. Training autoencoder to obtain filters: 63.3 seconds, training convolutional network: 3.4 hours, training stacked-AE: 34.25 minutes. Once trained, the elapsed times of segmenting the LV in a typical MR image were as follows: ROI detection (convolution, pooling, and logistic regression): 0.25 seconds, shape inferring (stacked-AE): 0.002 seconds, segmentation (deformable model): 0.2 seconds.

6. Discussion

In this study, we developed and validated an automated segmentation method for the LV based on deep learning algorithms. We broke down the problem into localization, shape inferring and segmentation tasks. Convolutional networks were chosen for localization and extracting an ROI because they are invariant to spacial translation and changes in scale and pixels' intensity (LeCun et al., 2010; Sermanet et al., 2014). We also chose a stacked AE for shape inferring because of its simplicity in training and implementation yet showing to be powerful in different vision tasks. (Vincent et al., 2010). Ideally, a pure deep learning was desired. However, this was not possible due to several challenges including the limited amount of training data. Thus, we integrated deep learning with deformable models to bring more accuracy to the method.

As seen in the left side of Fig. 8, the outcome of the deformable model without shape constraint (blue) leaked to surrounding tissues due to low contrast at the borders. Clearly this is not acceptable. On the other hand, the deep learning network (shape inference) provided a close contour (red) to the ground truth (green) with no leakage. This is due to the fact that the network has been trained using the ground truth data to look for the overall shape of the LV and not the intensity difference at the border. Finally, the integrated deep learning and deformable models brought the contour (yellow) closer to the ground truth. Similar behavior can be seen in the right side of Fig. 8 when contours tend to shrink due to presence of papillary muscles in the LV.

From Figs. 9-11, it can be seen that the LV was accurately segmented from the base to the apex. The alignment process resulted in a smooth 3D reconstruction of the LV in Fig. 9. The first image on the left corner of Fig. 9 shows a slight leakage from the ground truth. This situation is one of the challenging cases that, due to the fuzziness of the LV border, contours tend to leak to surrounding tissues in pure deformable models. Luckily, by integrating the inferred shape into the deformable models, this leakage was significantly prevented in this image and also all similar cases in the dataset such as the first and second images in the fifth row of Fig. 11. In other challenging cases, such as in Fig. 10, that pure deformable models tend to shrink inward due to the presence of papillary muscles, or leak outward due to low resolution and small contrast of images at the apex, our method overcame these shortcomings.

Computed metrics in Table 1 showed that the inferred shape provided good initial contours with an overall accuracy of 90% (in terms of DM). Also, the integrated deformable model provided final contours with great agreement with the ground truth with an overall DM of 94% and improvements in other metrics. Table 2 revealed that our method outperformed the state-of-the-art methods and significant improvements were achieved in all metrics. Specifically, the DM and conformity were improved by 4% and 0.08 compared to the best DM and conformity reported by (Queiros et al., 2014).

The correlation analysis in Figs. 12-14 depicted a high correlation for the three clinical cardiac indices. The high correlation between the automatic and manual references shows the accuracy and clinical applicability of the proposed framework for automatic evaluation of the LV function. Also, the Bland-Altman analysis in the figures revealed a better level of agreement compared with that of (Queiros et al., 2014). On the other hand, the level of agreement of frameworks of (Cordero-Grande et al., 2011b; Eslami et al., 2013) are slightly better than that of our method, which can be related to the semi-automated property of these methods compared with our fully automatic approach.

The measured elapsed time revealed that the method can be trained within a reasonable time, which can be performed offline. The longest time was needed for the convolutional network, which required convolution of the filters with images. Nevertheless, these times can be even shortened by developing the algorithms into GPU-accelerated computing platforms instead of our current CPU-based platform. In testing, the average time to perform the LV segmentation in a typical image was found less than 0.5 seconds, of which mostly taken by the convolution network and the integrated deformable model. Yet, the integrated deformable model converges faster than pure deformable models because of the initialization and integration with the inferred shape. Some of the published works provide numbers for the computational time. However, since each method has been developed on a different computing platform, these values may not be reliable for comparison unless all the methods are developed on the same platform.

It is noted that, while 3D methods are becoming the state-of-the-art in many medical image analysis applications, we performed 2-dimensional (2D) processing in the present study. This choice was due to two known challenges in cardiac MRI that prevents one from direct 3-dimensional (3D) analysis. First, the gap between slices (vertical dimension) in most routine acquisitions is relatively large (around 7-8 mm) and the pixel intensities between

the slices cannot be reliably estimated (Petitjean and Dacher, 2011; Tavakoli and Amini, 2013; Petitjean et al., 2015; Queiros et al., 2014). Second, due to motion artifacts in cardiac MRI, misalignment between slices is common (Barajas et al., 2006; Chandler et al., 2008; Lötjönen et al., 2004; Zakkaroff et al.; Carminati et al., 2014; Liew et al., 2015). This means that the cavity center is not at the same position in different slices. Some of existing tools perform an initial 2D segmentation in the middle slice and later apply an alignment process and then convert from the Cartesian coordinate to the polar coordinate to be able to perform 3D processing (Queiros et al., 2014). Alternatively, atlas-based techniques build a reference 3D model from some training data and then register the new image to the reference model, which limits the accuracy of segmentation to the reference model. Accordingly, different approaches can be adapted for our method if 3D processing is sought. For instance, instead of feeding 2D images in the current method, 3D data can be fed to the deep learning networks and trained for a 3D reconstruction. This would require networks with higher number of nodes and layers. Considering these challenges and additional complexity burden, the possibility of performing 3D computation can be investigated in future. Nevertheless, cardiac chamber segmentation in clinical practice is mainly used to compute clinical indices such as the volume or ejection fraction. Our proposed method is able to provide these indices with high accuracy while performing 2D processing.

Finally, one of the difficulties in developing deep learning and machine learning approaches for cardiac MRI segmentation is the lack of adequate data for training and validation. Particularly for deep learning networks, access to more data helps to reach a better generalization and reduce the overfitting problem. For this work, we used a portion of the MICCAI dataset (Radau et al., 2009) and artificially enlarged the dataset for training. However, in this case the training data are highly correlated, which would limit the performance. Also, currently, there are no analytic approaches to design hyper-parameters (such as number of layers and units, filter size, etc.) in deep learning networks and they are mainly obtained empirically, as we performed in our study.

7. Conclusion

In summary, a novel method for fully automatic segmentation of the LV from cardiac MRI datasets was presented. The method employed deep

learning algorithms for automatic detection and inferring the shape of the LV. The shape was incorporated into deformable models and brought more robustness and accuracy, particularly for challenging basal and apical slices. The proposed approach was shown to be accurate and robust compared to the other state-of-the-art methods. Excellent agreement and a high correlation with reference contours were obtained. In contrast with other automated approaches, our method is based on learning several levels of representations, corresponding to a hierarchy of features and does not assume any model or assumption about the image or heart. The feasibility and performance of this segmentation method was successfully demonstrated through computing validation metrics with respect to the gold standard on the MICCAI 2009 database (Radau et al., 2009). Testing our method on a larger set of clinical data is subject of future research.

Acknowledgments

This work is partially supported by a grant from American Heart Association (14GRNT18800013).

References

Assen, H.C., Danilouchkine, M.G., Frangi, A.F., Ords, S., Westenberg, J.J., Reiber, J.H., Lelieveldt, B.P., 2006. Spasm: A 3d-asm for segmentation of sparse and arbitrarily oriented cardiac MRI data. Medical Image Analysis 10, 286 – 303.

Babalola, K.O., Patenaude, B., Aljabar, P., Schnabel, J., Kennedy, D., Crum, W., Smith, S., Cootes, T.F., Jenkinson, M., Rueckert, D., 2008. Comparison and evaluation of segmentation techniques for subcortical structures in brain MRI, Springer, pp. 409–416.

Baldi, P., 2012. Autoencoders, unsupervised learning, and deep architectures. ICML Unsupervised and Transfer Learning 27, 37–50.

Barajas, J., Caballero, K.L., Barnés, J.G., Carreras, F., Pujadas, S., Radeva, P., 2006. Correction of misalignment artifacts among 2-D cardiac mr images in 3-D space, in: First International Workshop on Computer Vision for Intravascular and Intracardiac Imaging, MICCAI 2006, pp. 114–121.

Ben Ayed, I., Li, S., Ross, I., 2009. Embedding overlap priors in variational left ventricle tracking. IEEE Transactions on Medical Imaging 28, 1902–1913.

Bengio, Y., 2009. Learning deep architectures for AI. Foundations and trends in Machine Learning 2, 1–127.

Bengio, Y., Courville, A., Vincent, P., 2013. Representation learning: A review and new perspectives. IEEE Trans. Patt. Anal. Mach. Intel. 35, 1798–1828.

Billet, F., Sermesant, M., Delingette, H., Ayache, N., 2009. Cardiac motion recovery and boundary conditions estimation by coupling an electromechanical model and cine-mri data. Functional Imaging and Modeling of the Heart , 376–385.

Bland, J.M., Altman, D., 1986. Statistical methods for assessing agreement between two methods of clinical measurement. The lancet 327, 307–310.

Boureau, Y.L., Ponce, J., LeCun, Y., 2010. A theoretical analysis of feature pooling in visual recognition. Proceedings of the 27th International Conference on Machine Learning (ICML-10) , 111–118.

Carminati, M.C., Maffessanti, F., Caiani, E.G., 2014. Nearly automated motion artifacts correction between multi breath-hold short-axis and long-axis cine cmr images. Computers in biology and medicine 46, 42–50.

Chan, T., Vese, L., 2001. Active contours without edges. IEEE Trans. Img. Proc. 10, 266–277.

Chandler, A.G., Pinder, R.J., Netsch, T., Schnabel, J.A., Hawkes, D.J., Hill, D.L., Razavi, R., 2008. Correction of misaligned slices in multi-slice cardiovascular magnetic resonance using slice-to-volume registration. Journal of Cardiovascular Magnetic Resonance 10, 1–9.

Chang, H.H., Valentino, D.J., Chu, W.C., 2010. Active shape modeling with electric flows. IEEE Transactions on Visualization and Computer Graphics 16, 854–869. doi:10.1109/TVCG.2009.212.

Chang, H.H., Zhuang, A.H., Valentino, D.J., Chu, W.C., 2009. Performance measure characterization for evaluating neuroimage segmentation algorithms. Neuroimage 47, 122–135.

Cobzas, D., Schmidt, M., 2009. Increased discrimination in level set methods with embedded conditional random fields. Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on , 328–335doi:10.1109/CVPR.2009.5206812.

Cocosco, C.A., Niessen, W.J., Netsch, T., Vonken, E.J., Lund, G., Stork, A., Viergever, M.A., 2008. Automatic image-driven segmentation of the ventricles in cardiac cine MRI. Journal of Magnetic Resonance Imaging 28, 366–374.

Constantinides, C., Roullot, E., Lefort, M., Frouin, F., 2012. Fully automated segmentation of the left ventricle applied to cine MR images: Description and results on a database of 45 subjects, pp. 3207–3210.

Cordero-Grande, L., Vegas-Sánchez-Ferrero, G., Casaseca-de-la Higuera, P., San-Román-Calvar, J.A., Revilla-Orodea, A., Martín-Fernández, M., Alberola-López, C., 2011a. Unsupervised 4d myocardium segmentation with a markov random field based deformable model. Medical image analysis 15, 283–301.

Cordero-Grande, L., Vegas-Snchez-Ferrero, G., de-la Higuera, P.C., San-Romn-Calvar, J.A., Revilla-Orodea, A., Martn-Fernndez, M., Alberola-Lpez, C., 2011b. Unsupervised 4d myocardium segmentation with a markov random field based deformable model. Medical Image Analysis 15, 283 – 301.

Deng, L., Yu, D., 2014. Deep Learning: Methods and Applications. Foundations and trends in signal processing, Now Publishers Incorporated.

Dreijer, J., Herbst, B., du Preez, J., 2013. Left ventricular segmentation from MRI datasets with edge modelling conditional random fields. BMC Medical Imaging 13. doi:10.1186/1471-2342-13-24.

Eslami, A., Karamalis, A., Katouzian, A., Navab, N., 2013. Segmentation by retrieval with guided random walks: Application to left ventricle segmentation in MRI. Medical Image Analysis 17, 236 – 253.

Frangi, A.F., Niessen, W., Viergever, M., 2001. Three-dimensional modeling for functional analysis of cardiac images, a review. IEEE Trans. Med. Imag. 20, 2–5.

Geremia, E., Clatz, O., Menze, B.H., Konukoglu, E., Criminisi, A., Ayache, N., 2011. Spatial decision forests for ms lesion segmentation in multi-channel magnetic resonance images. NeuroImage 57, 378–390.

Heimann, T., Meinzer, H.P., 2009. Statistical shape models for 3d medical image segmentation: A review. Medical Image Analysis 13, 543 – 563.

Hinton, G.E., Salakhutdinov, R.R., 2006. Reducing the dimensionality of data with neural networks. Science 313, 504–507. doi:10.1126/science.1127647.

Hu, H., Liu, H., Gao, Z., Huang, L., 2013. Hybrid segmentation of left ventricle in cardiac MRI using gaussian-mixture model and region restricted dynamic programming. Magnetic Resonance Imaging 31, 575 – 584.

Huang, R., Pavlovic, V., Metaxas, D.N., 2004. A graphical model framework for coupling MRFs and deformable models. Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on 2, II–739.

Huang, S., Liu, J., Lee, L.C., Venkatesh, S., Teo, L., Au, C., Nowinski, W., 2011. An image-based comprehensive approach for automatic segmentation of left ventricle from cardiac short axis cine MR images. Journal of Digital Imaging 24, 598–608. doi:10.1007/s10278-010-9315-4.

Jolly, M., 2009. Fully automatic left ventricle segmentation in cardiac cine MR images using registration and minimum surfaces. The MIDAS Journal-Cardiac MR Left Ventricle Segmentation Challenge 4.

Jolly, M., Xue, H., Grady, L., Guehring, J., 2009. Combining registration and minimum surfaces for the segmentation of the left ventricle in cardiac cine MR images, Springer Berlin Heidelberg. volume 5762, pp. 910–918.

Kedenburg, G., Cocosco, C.A., Köthe, U., Niessen, W.J., Vonken, E.P.A., Viergever, M.A., 2006. Automatic cardiac MRI myocardium segmentation using graphcut, International Society for Optics and Photonics. pp. 61440A–61440A.

Koikkalainen, J., Tolli, T., Lauerma, K., Antila, K., Mattila, E., Lilja, M., Lotjonen, J., 2008. Methods of artificial enlargement of the training set

for statistical shape models. IEEE Transactions on Medical Imaging 27, 1643–1654.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, pp. 1097–1105.

Kullback, S., Leibler, R.A., 1951. On information and sufficiency. Ann. Math. Statist. 22, 79–86. doi:10.1214/aoms/1177729694.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444.

LeCun, Y., Kavukcuoglu, K., Farabet, C., 2010. Convolutional networks and applications in vision, pp. 253–256.

Lempitsky, V., Verhoek, M., Noble, J.A., Blake, A., 2009. Random forest classification for automatic delineation of myocardium in real-time 3d echocardiography. Functional Imaging and Modeling of the Heart , 447–456.

Liew, Y., McLaughlin, R., Chan, B., Aziz, Y.A., Chee, K., Ung, N., Tan, L., Lai, K., Ng, S., Lim, E., 2015. Motion corrected lv quantification based on 3d modelling for improved functional assessment in cardiac mri. Physics in medicine and biology 60, 2715.

Lima, J.C., Desai, M.Y., 2004. Cardiovascular magnetic resonance imaging: Current and emerging applications. Journal of the American College of Cardiology 44, 1164–1171.

Liu, H., Hu, H., Xu, X., Song, E., 2012. Automatic left ventricle segmentation in cardiac MRI using topological stable-state thresholding and region restricted dynamic programming. Academic Radiology 19, 723 – 731.

Lorenzo-Valdés, M., Sanchez-Ortiz, G.I., Elkington, A.G., Mohiaddin, R.H., Rueckert, D., 2004. Segmentation of 4D cardiac MR images using a probabilistic atlas and the EM algorithm. Medical Image Analysis 8, 255–265.

Lötjönen, J., Pollari, M., Kivistö, S., Lauerma, K., 2004. Correction of movement artifacts from 4-d cardiac short-and long-axis mr data, in: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2004. Springer, pp. 405–412.

Margeta, J., Geremia, E., Criminisi, A., Ayache, N., 2012. Layered spatio-temporal forests for left ventricle segmentation from 4d cardiac mri data. Statistical Atlases and Computational Models of the Heart. Imaging and Modelling Challenges , 109–119.

Ng, A., accessed July., 2015. the deep learning tutorial. URL: `http://deeplearning.stanford.edu/tutorial/`.

Ngo, T.A., Carneiro, G., 2014. Fully automated non-rigid segmentation with distance regularized level set evolution initialized and constrained by deep-structured inference, pp. 3118–3125.

Petitjean, C., Dacher, J.N., 2011. A review of segmentation methods in short axis cardiac MR images. Medical Image Analysis 15, 169 – 184.

Petitjean, C., Zuluaga, M.A., Bai, W., Dacher, J.N., Grosgeorge, D., Caudron, J., Ruan, S., Ayed, I.B., Cardoso, M.J., Chen, H.C., Jimenez-Carretero, D., Ledesma-Carbayo, M.J., Davatzikos, C., Doshi, J., Erus, G., Maier, O.M., Nambakhsh, C.M., Ou, Y., Ourselin, S., Peng, C.W., Peters, N.S., Peters, T.M., Rajchl, M., Rueckert, D., Santos, A., Shi, W., Wang, C.W., Wang, H., Yuan, J., 2015. Right ventricle segmentation from cardiac mri: A collation study. Medical Image Analysis 19, 187 – 202.

Pluempitiwiriyawej, C., Moura, J.M.F., Wu, Y.J.L., Ho, C., 2005. Stacs: New active contour scheme for cardiac MR image segmentation. IEEE Trans. Med. Img. 24, 593–603.

Queiros, S., Barbosa, D., Heyde, B., Morais, P., Vilaca, J.L., Friboulet, D., Bernard, O., D'hooge, J., 2014. Fast automatic myocardial segmentation in 4d cine CMR datasets. Medical Image Analysis 18, 1115 – 1131.

Radau, P., Lu, Y., Connelly, K., Paul, G., Dick, A., Wright, G., 2009. Evaluation framework for algorithms segmenting short axis cardiac MRI. MIDAS J. Caridac MR Left Ventricle Segmentation Challenge .

Schaerer, J., Casta, C., Pousin, J., Clarysse, P., 2010. A dynamic elastic model for segmentation and tracking of the heart in MR image sequences. Medical Image Analysis 14, 738 – 749.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y., 2014. Overfeat: Integrated recognition, localization and detection using

convolutional networks. International Conference on Learning Representations (ICLR2014) .

Suinesiaputra, A., Cowan, B.R., Al-Agamy, A.O., Elattar, M.A., Ayache, N., Fahmy, A.S., Khalifa, A.M., Gracia, P.M., Jolly, M.P., Kadish, A.H., Lee, D.C., Margeta, J., Warfield, S.K., Young, A.A., 2014. A collaborative resource to build consensus for automated left ventricular segmentation of cardiac MR images. Medical Image Analysis 18, 50 – 62.

Szegedy, C., Toshev, A., Erhan, D., 2013. Deep neural networks for object detection. Advances in Neural Information Processing Systems 26 , 2553–2561.

Tavakoli, V., Amini, A.A., 2013. A survey of shaped-based registration and segmentation techniques for cardiac images. Computer Vision and Image Understanding 117, 966 – 989.

Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A., 2008. Extracting and composing robust features with denoising autoencoders. Proceedings of the 25th international conference on Machine learning , 1096–1103.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A., 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. The Journal of Machine Learning Research 11, 3371–3408.

Yuan, C., Kerwin, W.S., Ferguson, M.S., Polissar, N., Zhang, S., Cai, J., Hatsukami, T.S., 2002. Contrast-enhanced high resolution MRI for atherosclerotic carotid artery tissue characterization. Journal of Magnetic Resonance Imaging 15, 62–67.

Zakkaroff, C., Radjenovic, A., Greenwood, J., Magee, D., . Stack alignment transform for misalignment correction in cardiac mr cine series .

Zhang, H., Wahle, A., Johnson, R.K., Scholz, T.D., Sonka, M., 2010. 4-d cardiac MR image analysis: left and right ventricular morphology and function. IEEE Trans. Med. Img. 29, 350–364.

Zhuang, X., Hawkes, D.J., Crum, W.R., Boubertakh, R., Uribe, S., Atkinson, D., Batchelor, P., Schaeffter, T., Razavi, R., Hill, D.L.G., 2008. Robust

registration between cardiac MRI images and atlas for segmentation propagation, International Society for Optics and Photonics. pp. 691408–691408.

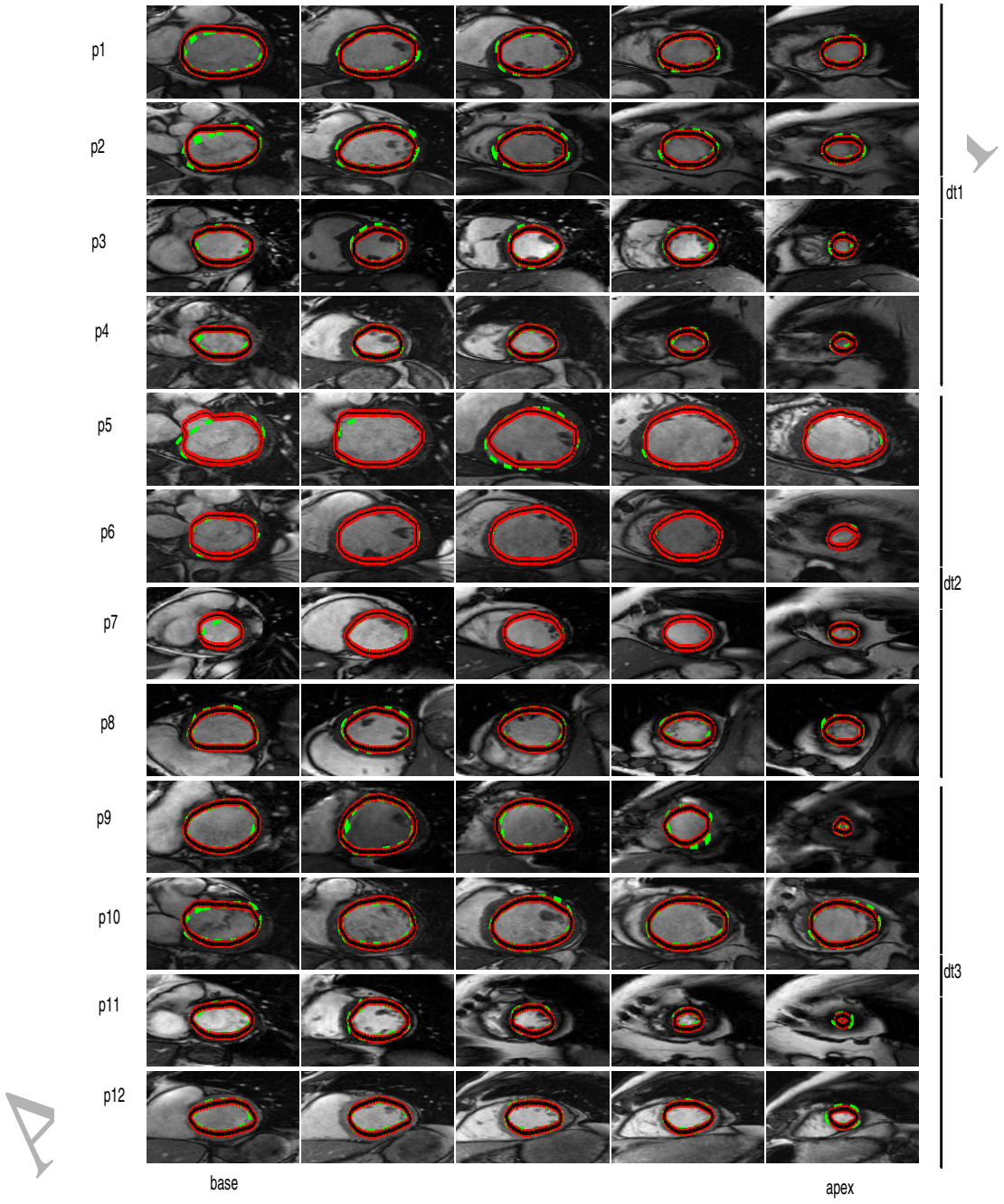


Figure 11: Automatic (red-black) and manual (dashed green) segmentation of LV in the base (left), mid-ventricular (middle) and the apex (right) slices for multiple cases of the MICCAI database (Radau et al., 2009). Each row corresponds to one patient, ischemic heart failure (SC-HF-I), non-ischemic heart failure (SC-HF-NI), LV hypertrphies (SC-HYP) and normal (SC-N)

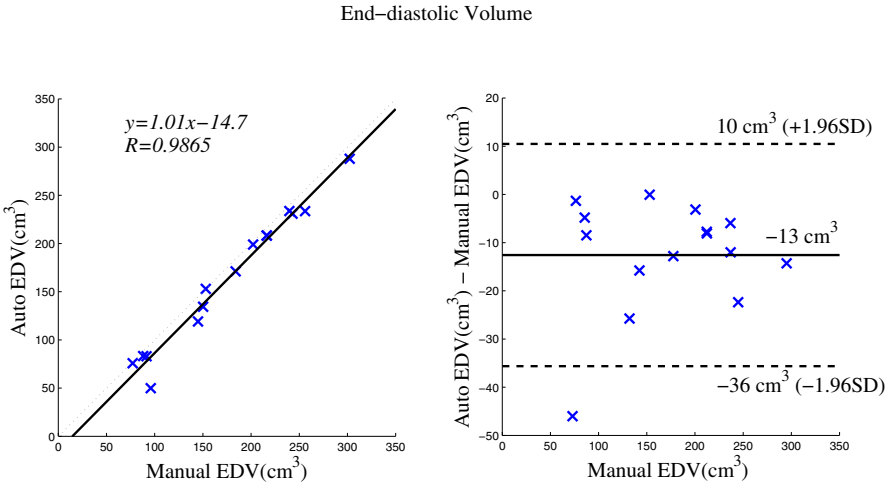


Figure 12: Correlation graph (left) and Bland-Altman(right) for end-diastolic volume (EDV).

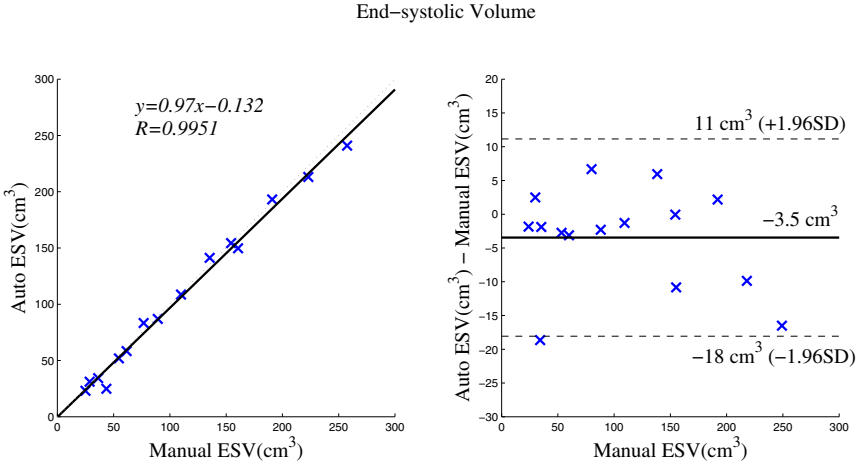


Figure 13: Correlation graph (left) and Bland-Altman(right) for end-systole volume (ESV).

Ejection Fraction

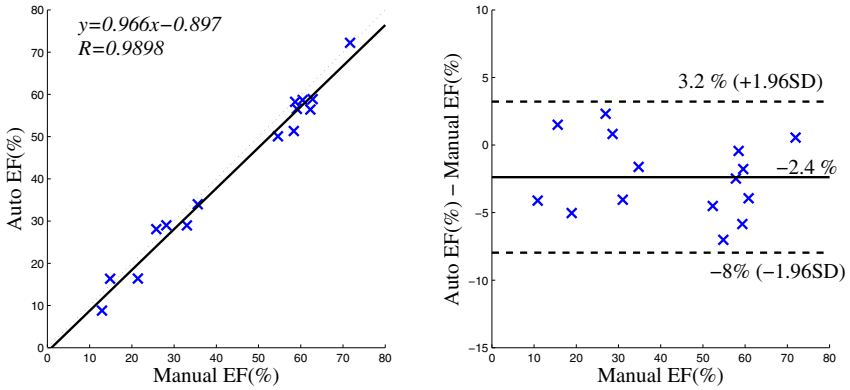


Figure 14: Correlation graph (left) and Bland-Altman(right) for the ejection fraction (EF).