Left Ventricle (LV) Segmentation using K-means and Deep Learing and volumetric calculation

Azadeh Hadadi[1], Adeyemi Abdulnafiu Olalekan[1], Alain Lalande[2],
[1]Condorcet University Centert, Universite Bourgogne et Franche-Compte, 71200 Le Creusot, France
[2]Le2I - Laboratoire de Biophysique, Faculté de Médecine Université de Bourgogne, 21079 Dijon, France

The paper focuses on the automatic cardiac diagnostic challenge (ACDC) and more specifically on LV segmentation in MRI images. Two fully automatic segmentation methods, i.e., k-means and deep learning, will be presented, and discussed in detail. The implementation of each method will be described and both methods will be unified in a GUI tool for better usability. The tool is a user interface application which developed in Python and wrapped under MATLAB to give better access for further development. The objective of the tool is to segment the LV during diastole and systole (only the endocardium) in MRI images and calculate ventricle cavity volume and ejection fraction. The methods were tested on MRI images of a database made of 100 patients. The experimental evaluation demonstrates promising results and significant precision for deep learning algorithm on entire database while k-means might prone to failure if intensity changes significantly. The deep learning algorithm using TensorFlow backbone was trained several times with different training dataset and the results were compared. The findal findings shows that if the number of the images used in training increases and the diversity of the selected images covers broad specturm of the scenarios then the precsion in practice increases significantly comparing other LV segementation methods. The result shows 50% accuracy improvement when database size increases four times.

***Index Terms*—Deep Learning, K-means, fully autmatic segmenation, Left Ventricle, MRI, cardiac diagnostic challenge.**

I. Introduction

Cardiovascular diseases cause 17.5 million deaths every year in all around the world [1]. Cardiac MRI image sequences, covering of one full period of cardiac cycle or over several periods,is for evaluating cardiac function. Evaluation of cardiac function requires calculation of different cardiac parameters (i.e. ejection fraction (EF), left ventricle mass (LVM), left ventricle volume, wall thickness, or wall thickening). All of these parameters can be acquired from segmented endocardium. Traditionally, cardiologists do manual segmentation of these contours in all dataset (i.e. all time frames in all slices) which is a very time consuming task. Therefore, the automatic segmentation of the LV is considered as a challenging task [2]. Several semi-automatic and automatic algorithms were proposed. Definitely automatic algorithm attracts more attention although due to more complicated scenarios in practice achieving high precision is usually hard. Recently, repaid development in deep learning has motivated many researchers to use this approaches. Therefore this research will focus on using deep learning for the LV segmentation too.

this paper will present a tool to segment the LV in diastole and systole only the endocardium using dataset aquired from **https://www.creatis.insa-lyon.fr/Challenge/acdc/** and calculate the ventricular cavity volume in diastole and systole. for this purpose, we will set up an easy training dataset generation, training and evaluation procedure which allows not only to be used in the LV segmentation but also can be repeated on other cardiac segmentation (or other organs) and analysis using a Mask-RCCN [3]. The Mask-RCCN employs TensorFlow backbone to train a given model of conventional neural network for detection and segmentation. Different training dataset and parameters lead to different procession. However, the higher precision needs more time which can go beyond few days even. In this work, we developed a tool which simplified data selection and make procedure and evaluation fully automatic with minimum user intervention. Beside, an existing successfully tested CNN architecture will be used to get training weights (gain, bias and so on) which significantly decrease to whole CNN training time. It has been practically proved that our proposed tool will help the user and researchers as well as developers to save a lot time and fairly high precision.

The remaining part of the report is organized as follows: In section II, literatures of the related works will be reviewed and summarized. then, LV structure and cardiac MRI will be explained. Section III will be dedicated to selected algorithm and theory which briefly presents different parts of the algorithm. Section IV will detail implementation, dataset preparation, training, evaluation for both K-means and Deep learning. Results will be presented in section V and discussed. The paper will end up to conclusion and references.

Manuscript submited May 22, 2020 for evaluation. Corresponding author: Azadeh Hadadi/Adeyemi Abdulnafiu Olalekan.

II. Review and Related Works

A. Literature Review

All algorithms presented for the LV segmentation can be divided into two categories, i.e., semi-automatic and automatic. Here in this literature only automatic segmentation will be reviewed. Hisham et al. [4] proposed a novel deep learning approach for the automated segmentation and quantification of the LV from cardiac cine MRI images using two consecutive F-CNNs, the first network for localization of ROI and the second network for precise segmentation. Dong et al. [5] uses two-networks architecture as [4] to segment the LV directly in 3D. They introduced a new network for the second stage,

which called AtlasNet and the paper is all about how to design AtlasNet to get precise 3D LV. Al Noman et al. [6] used parametric active contour model (PACM) to segment LV. They introduced an artificial neural network based regression model to predict the initial contour to detect the LV area in CMRI image. Kristanto et al. [7] submitted a US patient for LV segmentation by contrast enhancement. Moreno et al. [8] employed a combination of two convolutional neural networks (CNN) to develop a fully automatic LV segmentation method for Short Axis MRI datasets as of [4] but with little difference. The first CNN defines the region of interest (ROI) of the cardiac chambers based on You Only Look Once (YOLO) network. The output of YOLO net is used to filter the image and feed the second CNN, based on UNet network, which segments the myocardium and the blood pool. Wei et al. [9] proposed model leverages channel attention mechanism and shape correction auto-encoder, to adaptively enhanced the feature maps of different receptive field and correct the prediction shape that is inconsistent with the prior s, respectively. While most existing LV segmentation methods focus on cardiac images of single modality or multi-modality, few have been devoted to images of mixed-modality. Cong et al [10] presented a mixed-modality adaptive approached using MRI and CT image of LV for training and testing. Lan et al. [11] presented a deep learning based on double snake segmentation.
Moradi et al. [12] presented a new MFP-Unet CNN for LV segmentation using ultra-sound images and the result sound significantly precise. Segmentation of the LV continues to be a challenging task despite significant evolution of techniques and network architectures in last ten years. [13] describes fully automated iterative thresholding method for LV segmentation. [14] talks about a deep convolutional encoder-decoder model seg-net for image segmentation and [15] uses Fully Convolutional Neural Network (FCN) for cardiac image segmentation task. U-Net architecture [12] is another encoder-decoder architecture that has performed well on biomedical image segmentation tasks. In recent years several papers have been published on direct LV volume predictions without segmentation. [16] describes the approach using deep convolutional network and compares the performances of volume prediction using VGG, Google-net and Resnet architectures. Several solutions [17] [18] that were presented during Data Science Bowl 2 Challenge used different algorithms and network architectures to perform LV segmentation and volume prediction.

B. LV Structure

LV cavity's shape is ellipsoid and mayocardium surrounded it which a normal range for its thickness is 6-16 mm. On the other hand, Right ventricle's shape is more complex and also faces lower pressure than LV for ejecting blood to the lungs .Due to these reasons, most of the research efforts are on LV since its function is of greater importance than RV. The standard imaging plane is perpendicular to the long (apex-base) axis and called short axis plane as is shown in Figure 1. Imaging of the heart in MRI covers the whole organ with about 8–10 short-axis slices, distance between two adjacent slices

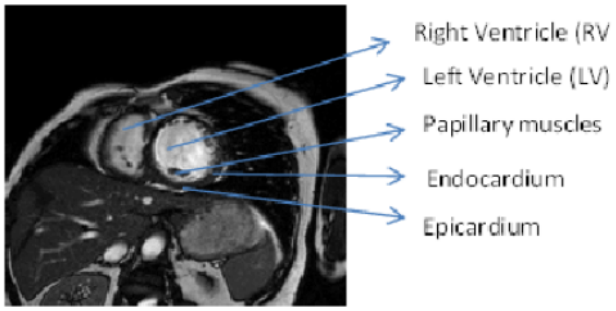

Fig. 1. Left ventricle in a short-axis MRI image

ranging from 10 to 20 mm. Normal MR image demonstrate blood pools as bright whereas myocardium and surrounding structures are shown in dark. Heart segmentation of MR images consist of outer wall called epicardium and inner wall called endocardium. The epicardial wall is at the frontier between the myocardium and surrounding tissues (fat, lung), which does not show a good contrast with the myocardium. On the other hand, for endocardium which surrounds the LV cavity, MRI demonstrates quite good contrast between myocardium and the blood flow even without contrast medium. in overall, since endocardium is less difficult to be segmented than the other one and also for computing ventricular volume, it is the only contour which is required, most of the works focus only on that.

C. Cardiac MRI

the MRI has unique features for accurate quantification of anatomy; function, flow, and perfusion of the cardiovascular system at rest and under stress conditions. Using multi-slice cine MRI techniques, the three-dimensional (3D) geometry of the heart can be imaged at high temporal and spatial resolution. As a 3D tomographic imaging technique, volumetric measurements do not rely on any geometrical assumption and therefore are accurate for both normal and abnormal ventricular anatomies; both left and right ventricular dimensions can be assessed with small margins of error.

III. Algorithm and Theory

There are several techniques used in the LV segmentation. In general, we can classify these techniques according to the algorithm that used in the segmentation process to image-based methods, pixel classification methods , deformable models , model-based methods, deep learning, and finally the atlas guided method [2].

A. K-means Segmentation

The first approach for automatic LV segmentation is the K-means clustering algorithm and graph-searching on cardiac MRI [19]. Since Clustering is to divide a given data into groups having similar properties between classe members which is genearlly called similarity criteria. the K-means clustering is one of the most common iterative algorithm which iteravtively adjust K-means center of mass. K-means clustering

works usually based on squared error minimization. The algorithm is simple and fast, and applies an iterative optimization process in most of the slices, the LV has the highest intensity value. The detil of the algorithm was presented in the work published by Lee et al. [19].

B. Deep Learning LV Segmentation

The second implementation of the LV segmentation is based on Mask R-CNN.

1) Mask R-CNN

This Mask-RCNN is simple, flexible, and a general framework for object instance segmentation [3]. This framework extends Faster R-CNN by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition. The mask branch is a small FCN applied to each ROI, predicting a segmentation mask in a pixel-to pixel manner.

Although Fast/Faster RCNN [20] [21] and Fully Convolutional Network (FCN) [21] are frameworks for object detection and semantic segmentation, respectively, Mask R-CNN is an enable framework for ***instance segmentation***. The output of each candidate object in Mask R-CNN includes:

- A class label (Faster R-CNN)
- A bounding- box offset (Faster R-CNN)
- additional mask output which is distinct from the class and box outputs, requiring extraction of much finer spatial layout of an object.(***Mask R-CNN***)

Mask R-CNN has two stages:
1. Region Proposal Network (RPN) which proposes candidate object bounding boxes
2. predicting the class and box offset in parallel (Mask R-CNN also outputs a binary mask for each ROI)

The multi-task loss on each sampled ROI during training is defined as:

$$L = L_{cls} + L_{box} + L_{mask}$$

IV. Implementation

Several tools of LV segmentation have already been presented and shared with other researchers for development and further exploration. Hereafter, we will frist explain implementation for deep learing which inspired this work and after breiefly explain the K-means.

Chen et al. [22] presented a source code for semi-automatic segmentation using active counter without edge. The open source implementation in MATLAB is accessible and can be downloaded from their GitHub page [23].Theory and mathematical detail behind their implementation is described in [24]. This job later was extended and integrated with another deep learning implementation. The complete source code is available in this link [25]. A part of this work has been used to test semi-automatic approach just to see the prose and crone of semi-automatic approach.

Julian Dewit et al [26] presented another python implementation for LV segmentation by deep learning, which works on GPU. As we are quite limited in terms of hardware this work was referred to as general guideline only and the code was not used. Further detail of the background and theory behind this implementation presented in [27]. Another open source code was presented by Abdelmaguide et al. [28], which developed based on deep learning in python. The code uses TensorFlow backbone for training and evaluation and the detail of the CNN model presented in [28]. This development is highly depending on code published by De wit et al [26].

Our development is not relaying on this code and no part of their code was used, but their handout is a good source to get an idea where to start the development and how to establish the training and testing framework. Woshi et al. [29] has published another source code developed in python. The detail of the development and code can be read in [30]. None of the above work has been completely integrated in our development nor partially used. Our development was more motivated by approach explained by Khush Patel [31]. Patel explains how to perform object detection using simple example, this helped us to understand how to setup TensorFlow and used it for new dataset training and evaluation. The code primarily developed to detect the ROI for the LV. Then this code extended to be trained to segment certain object using mask. More specifically, different masks were extracted from training data set and the CNN network was trained and tested for LV segmentation.

A. Deep Learning Segmentation Using Tensorflow

The second approach is based on MASK-RCNN framework, which was established and tested in [3]. The code was first developed in python and for simplicity, usability, easier development and better visualization was wrapped under MATLAB code. Three main items of the segmentation network will be detailed, i.e., CNN network, ROI definition to the LV localization, different masks used for LV segmentation, and training setting.

1) R-CNN Network Architecture

Feature Pyramid Network (FPN) is a backbone architecture which was proposed by Lin et al. [32]. FPN uses top-down architecture with lateral connections to build an in-network feature pyramid from a single-scale input. Faster R-CNN with an FPN backbone extracts ROI features from different levels of the feature pyramid according to their scale. Using a ResNet-FPN backbone for feature extraction with Mask RCNN gives excellent gains in both accuracy and speed.

Also, the network ***head*** will be followed to which a fully convolutional mask prediction branch will be added. Details are shown in Figure 2. The head on the ResNet-C4 backbone includes the 5-th stage of ResNet (namely, the 9-layer ‘res5’ [33]), which is compute-intensive. For FPN, the backbone already includes res5 and thus allows for a more efficient head that uses fewer filters. Mask branches have a straightforward structure. In the Mask R-CNN two existing Faster RCNN heads are extend. In Figure 2, Left/Right panels show the heads for the ResNet C4 and FPN backbones, from and, respectively, to which a mask branch is added. Numbers denote spatial resolution and channels. Arrows denote either conv, deconv, or fc layers as can be inferred from context (conv preserves spatial dimension while deconv increases it).

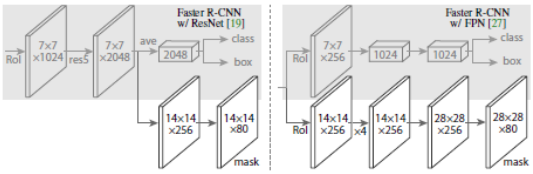

Fig. 2. Mask-RCNN Architecture for LV segmentation

Fig. 3. (a) elliptical mask generated manually (b, c) mask generated automatically by snake [34] (d) mask generated manually using polygon method

As of other CNN network presented above and in related literature, our network has two main part, convolution layers (feature extraction) and neural network for classification. Two networks were used consecutively for ROI localization and segmentation.

2) ROI Detection

The first RCNN network detects the ROI. In fact, the ROI is selected for each image and stored in an *.XML file when the training data is prepared. The ROI is selected as rectangular and defined by six parameters $[x_{\min}, y_{\min}, x_{\max}, y_{\max}, w, h]$.Coordinate $(x_{\min}, y_{\min})$ defines the left top corner of the rectangle which defines ROI and w,h defines width and height of the rectangle. These data are stored for each image in a file with same name as training image.

3) Segmentation Mask

Different masks were created and used during the training. There are several possibilities to select the masks, either manually (polygonal, elliptical or free hand) or selected semi-automatically using snake algorithm [34] or any other precise segmentation algorithm. Practical experiments reveal the fact that creating mask manually leads to a better and more precise results. 3 types of segmentation masks used in training phase as shown in Figure 3.

4) Training Setting

The training parameters were setup as of Table 1. After running the training procedure a *.h file which contains all training weights. This weight is necessary to test and validate the training result using CNN architecture for ROI detection and LV segmentation. As in Fast R-CNN, an RIO is considered positive if it has IoU with a ground-truth box of at least 0.5 and negative otherwise. The mask loss L_{mask} is defined only on positive ROIs. The mask target is the intersection between an ROI and its associated ground-truth mask.

Table 1:CCN network training parameters

BACKBONE	Resnet101	LEARNING_MOMENTUM	0.9
BACKBONE_STRIDES	[4,8,16,32,64]	WEIGHT_DECAY	0.0001
IMAGE_SHAPE	[1024 1024 3]	BATCH_SIZE	2
IMAGE_RESIZE_MODE	Square	COMPUTE_BACKBONE_SHAPE	None
IMAGE_MIN_SCALE	0	DETECTION_MAX_INSTANCES	100
IMAGE_MIN_DIM	800	DETECTION_MIN_CONFIDENCE	0.7
IMAGE_META_SIZE	14	DETECTION_NMS_THRESHOLD	0.3
IMAGE_MAX_DIM	1024	FPN_CLASSIF_FC_LAYERS_SIZE	1024
IMAGE_CHANNEL_COUNT	3	GPU_COUNT	1
GRADIENT_CLIP_NORM	5.0	IMAGES_PER_GPU	2
LEARNING_RATE	0.001	BBOX_STD_DEV	[0.1 0.1 0.2 0.2]
MASK_POOL_SIZE	14	MAX_GT_INSTANCES	100
MASK_SHAPE	[28, 28]	MEAN_PIXEL	[123.7,116.8,103.9]
MINI_MASK_SHAPE	(56, 56)	NUM_CLASSES	2
NAME	left_ventricle_cfg	POOL_SIZE	7
POST_NMS_ROIS_INFERENCE	1000	POST_NMS_ROIS_TRAINING	2000
PRE_NMS_LIMIT	6000	ROI_POSITIVE_RATIO	0.33
RPN_ANCHOR_RATIOS	[0.5, 1, 2]	RPN_ANCHOR_STRIDE	1
RPN_ANCHOR_SCALES	(32,64,128,256,512)	RPN_BBOX_STD_DEV	[0.1 0.1 0.2 0.2]
RPN_NMS_THRESHOLD	0.7	STEPS_PER_EPOCH	130
RPN_TRAIN_ANCHORS_PER_IMAGE	256	TRAIN_ROIS_PER_IMAGE	200
USE_RPN_ROIS	True	TOP_DOWN_PYRAMID_SIZE	256
VALIDATION_STEPS	50	TRAIN_BN/USE_MINI_MASK	False/True
LOSS_WEIGHTS	{rpn_class_loss:1.0 ,rpn_bbox_loss:1.0,	,mrcnn_class_loss: 1.0 ,mrcnn_bbox_loss': 1.0,	mrcnn_mask_loss:1.0 }

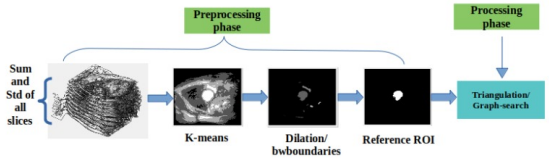

Fig. 4. K-means process

B. K-means

In order to implement the segmentation, we presented the k-means algorithm in 2 phases:

1) PreProcessing Phase

druign the first phase, the seed points and initial LV information such as center of gravity and intensity statistics are calculated. Then, the K-means clustering of pixels intensity values is computed. The LV is segmented by K-means selecting the correct value of K.Finally, an erosion operation is performed to extract the region of the LV.

2) Processing Phase

The triangulation to align each segment image on the slices to the center of the reference segmentation image is performed lastly in preprocessing phase. Then, segmented error is corrected by graph search. Here the idea is to visit every node exactly once. the diagram shown in Figure 4 to understand the process.

The K-means is an exploratory analysis technique and also implements a non hierarchical method of grouping of objects together which means it just takes datasets as they come in and then group them together and also uses the centroid to Euclidean method to calculate distance and groups objects based on the minimum distance. In this paper, we give the method to determine the initial centre in the previous sample based on streamline, the maximum minimum distance method is introduced into density as a measure, the initial clustering centre K choose sample K features of highest density sample, then, iteration to determine the initial clustering centre.

C. Volumetric Calculation

- With the segmented part, we can estimate the cavity volume and ejection Fraction EF. We took the slice thickness $S_t = 5$ and $S_g = 5$ to compute the volume cavity as:

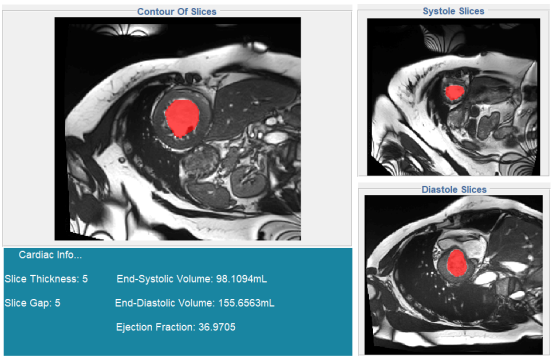

Fig. 5. K-means Clustering result sample

$$V = (S_g + S_t) * \sum_{i=1}^{n} A_i$$

- Left Ventricle Mass LVM and Ejection fraction EF are defined as in the following equations:

$$LVM = [VED_{epi} - VED_{endo}] * 1.05$$
$$EF = \frac{EDV - ESV}{EDV} * 100\%$$

Note: VED_{epi} and VED_{endo} represent epi and endo cordial volumes in the end Diastole while EDV and ESV represent endocardial volume in the end diastole and systole volume phase.

A complete guideline to use GUI interface for algorithm selection, training and testing is avaialbe in the detail report [35].

V. Result and Discussion

In this section, result of the above methodologies will be presented and detailed. K-means Clustering segmentation was done in the endocardium which produces a good result and the volume calculation of diastole and systole for patient2 as shown in the GUI image attached below: However, the same result might not be generated for all patients in this method because this method is prone to failure under more complex scenarios such as intensity variation, imaging artefacts, surrounding argons with intersection as LV in the same slices. The blockage of the LV due to cardio-vascular disease in the patient's image might create even more problem and leads to segmentation failure in this method. To overcome this problem we came up with other methods implemented using deep learning which has already been explained in more detail in the previous section. Here blow, first, we will have a furtive glance over the method results and then we will pass to the comparison of the result.

In Deep Learning methodology, by selecting a patient folder in ***Create Dataset*** panel, systole and diastole slices are read and converted to jpg files.

These images of the patient can be navigate back and forth slice by slice as shown in this figure and detail in section 7. The training and validation and testing procedure also were described in the same section.

Now, we will have a look at some results achieved using by this method. The result of the segmentation of the LV during diastole and systole is very similar to those have been generated by K-means algorithm. We keep the same display

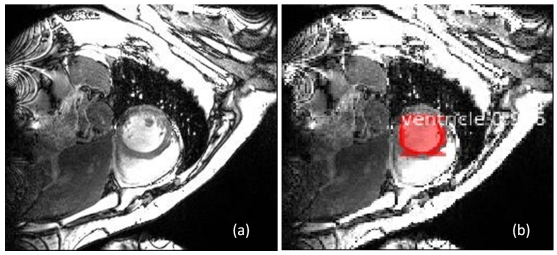

Fig. 6. Single image (a) its segmented result by Deep Learning (b)

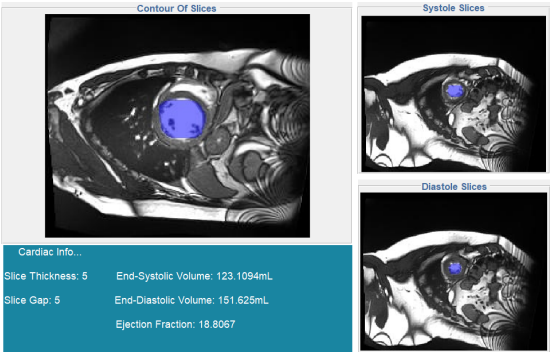

Fig. 7. Deep Learning result sample

and visualization interface meaning the result will appear in the same panel as of K-means. One sample of original image and its segmented result is shown in Figure 6 (a) and (b) respectively. In this method, the segmented region is highlighted in red in Figure 7. Sometimes, the algorithm might find several ROI by the first network. Naturally in this case, somehow, the algorithm shall determine which region is the most probable candidate for LV. For this purpose, the algorithm calculates a score for each segmented region. Definitely the highest score belongs to the most probable region of interest. For instance for a given image score for LV is 0.996. Practical experiment reveals the fact that in 95% of the time only one region is detected and segmented. However, in 5% reaming cases not more than two regions are detected and segmented and when the score is considered as an additional criteria the correct region can be selected which is 98% of time correct candidate for the LV.

Each NIFTI file contains several images for per patient. When we select the patient folder and start segmentation all images were selected and segmented for diastole and systole. Finally, the patient's LV is segmented by Deep Learning algorithm and volume for systolic and diastolic and ejection fraction metrics for slices are displayed as shown in Figure 7. The quality of the segmentation is highly depends on the selected training dataset and complexity of the scenarios involved in the images, the more complexity the more precision.

De wit et al. [27] presented some result achieved by deep learning using very similar approach as presented in this work. However the approach seems not work properly when intensity is changing. For that reason De wit teams uses a pre-processing step to improve the intensity mostly by applying intensity enhancement or histogram equalization and other methods which improves the segmentation results consequently. The

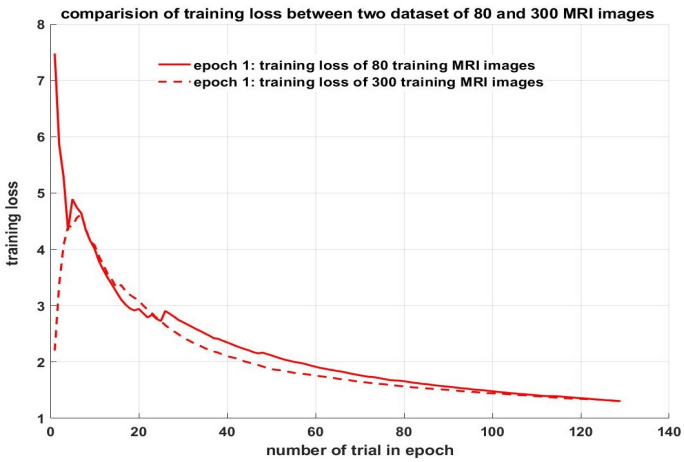

Fig. 8. Training loss of epoch 1 for dataset of 80 and 300 images

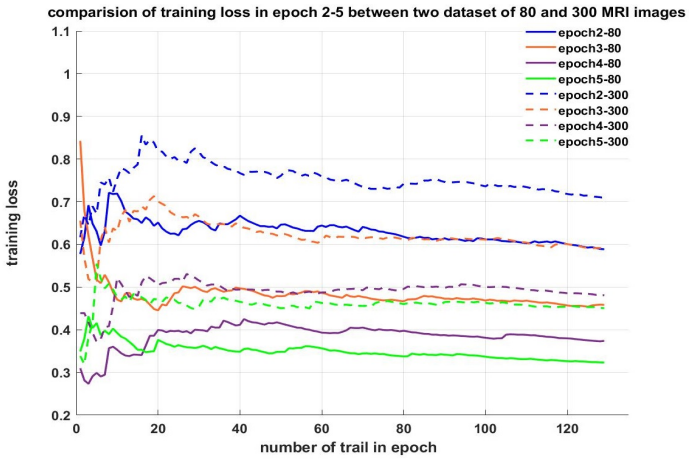

Fig. 9. Training loss of epoch 2-5 for both dataset with 80 and 300 images

processing has been advised by different researcher. However, we preceded different approach to resolve this issue by selecting proper training dataset and avoiding time-consuming pre-processing and image enhancement algorithm. In fact, we have selected training images from different patients with more verities of intensity and other image diversities when making the training dataset. It seems quite effective approached and leads to better result in practice although this approach necessarily will leads to longer training time and bigger training database. The practical experiment conducted on the unknown images shows the segmentation result is promising.

To investigate this approach more and to study the effect of the training dataset, image diversity and training time as well as accuracy of the segmentation, we have selected thee dataset, i.e., 20, 80, 300 MRI images. We setup training for each datasets and ran the training process as detail under section 7 with 5 epochs of training consecutively non-stop and in an automatic manner and at each epoch we perform 130 steps of trail for training. The number of trail for dataset with 20 images was 80 trials per epoch. The training weight was validated and tested but the results was very good because first of all several candidates were detected and segmented and the segmentation error was quite high. So this dataset was kept aside and only dataset with 80 and 300 images were considered. The training of both dataset was performed with 5 epochs and 130 trails per epoch. The training loss for both dataset for epoch 1 is shown in Figure 8. As seen, the training loss is quite significant at early trial and it decreases exponentially when the trail increases. The training loss for dataset with 300 images is most of the time lower than dataset with 80 images. This difference is consequently affecting the validation loss as shown in Figure 10. In each dataset, 20% of the images are selected for validation which was not involved in training process and remaining 80% is used for training only. The training loss of remaining epochs (epoch 2-5) is shown in Figure 9. As seen inversely, all training loss for the remaining epochs are higher for dataset with 80 images while we expect vice versa. However, this discrepancy in the training loss does not affect the validation loss because of the diversity involved in the training as mentioned above. We observed that in early trails the losses increase and reach to a local pick and after exponentially decrease which show almost correct training curve as we expect. Another observation comparing between training with 80 images versus 300 images is that during the validation and testing process the number of candidate detected by the weight extracted from 80 images for LV is more than one sometimes. Besides, the higher score might be allocated to the region which is not really true ROI as expected. This is not the case when weight extract from training which uses 300 images instead. In this case, always only one ROI for LV is detected and segmented. In a very rare case, especially at the beginning of diastole, because of small area of LV the detected ROI are not completely segmented. This is not a failure because even with the bare eye it is hard to recognize LV in this scenario and needs more experience. Sometimes the same issues are observed for systole. Looking into the detail provided in Figure 10, validation loss, it is clearly visible that losses for segmentation with resulted weight of 300 images are smaller than weight extracted from dataset of 80 images. This difference for all epochs is significant and the difference increases when the training process steps into the higher epoch. The difference of the losses for epoch 5 is almost 50% which consequently leads to very high precision for the weight calculated in epoch 5. This level of precision naturally needs more computation time as shown in Figure 11. In this experiment the training executed using general purpose CPU with multiple processing cores and the computation time when the number of the images increases is quite significant (half a day up to a day). For instance, as seen in Figure 11, it took almost half a day to complete one training epoch when 300 images were used as training dataset. The overall times spend on training using dataset with 300 image roughly is two times than the training of dataset with 80 images as shown in Figure 11 which leads to two times more precision as seen in Figure 10. It was recommended by the TensorFlow developer to use cluster GPU cards to accelerate the training process. We tested GPU Tensorflow and noticed always memory overflow just as a general rule to run this training on GPU to save the time, the target GPU should contain more than 12GB of internal RAM. Otherwise,

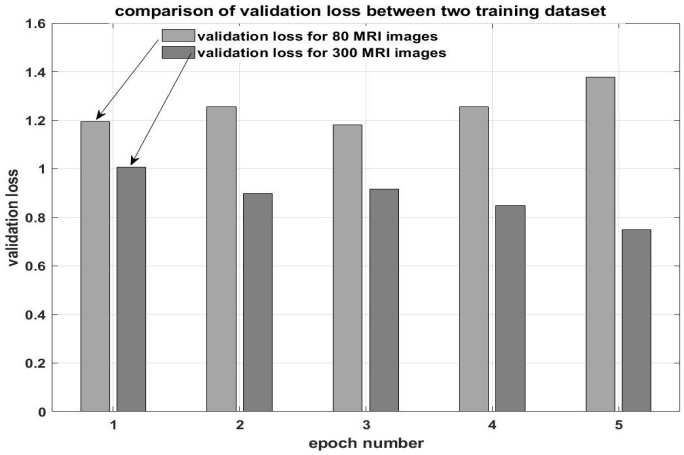

Fig. 10. Validation loss of all epoch for both dataset of 80 and 300 images

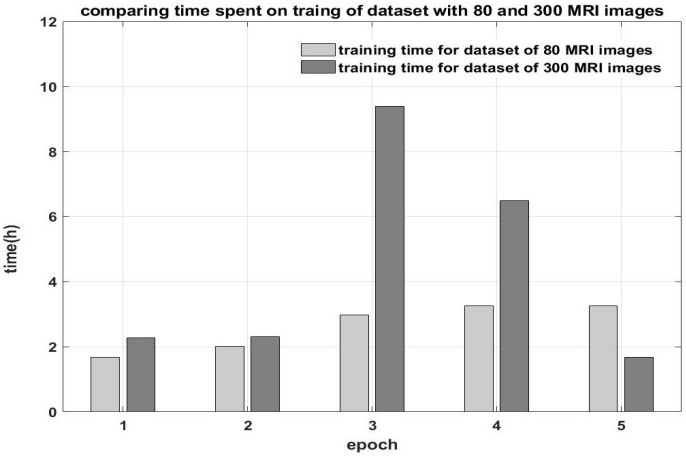

Fig. 11. Computation time difference of training for dataset of 80 and 300 images

the GPU implementation will fail which was the case for our experiment.

VI. Conclusion

Two algorithms, k-means and deep learning, were proposed for automatic segmentation of the LV. Each algorithm was implemented and practical experiment was setup to validate and test the algorithm. The result of the practical experiment for k-means reveals the fact that k-mean may prone to failure under complex scenarios. For that reason, alternative algorithm using deep learning based on TensorFlow was developed, trained and tested. Two datasets of 80 and 300 MRI images each were selected to training deep learning network. The experimental result of training and testing show that the result achieved by deep learning algorithm is promising even under significant image changes and more complex scenarios. Besides, dataset with 300 images takes more time (almost two times) more than dataset with 80 images to be trained which leads to 95% to 98% precision in terms of ROI detection and segmentation. The segmentation loss decreases 50% (consequently accuracy increases) when the number of the images of the dataset increases almost four folds.

References

[1] J. R. RJ van der Geest, "Quantification in cardiac mri," *JOURNAL OF MAGNETIC RESONANCE IMAGIN*, 1999.

[2] M. M. Hadhoud, M. I. Eladawy, A. Farag, F. M. Montevecchi, and U. Morbiducci, "Left ventricle segmentation in cardiac mri images," *American Journal of Biomedical Engineering*, vol. 2, no. 3, pp. 131–135, 2012.

[3] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[4] H. Abdeltawab, F. Khalifa, F. Taher, N. S. Alghamdi, M. Ghazal, G. Beache, T. Mohamed, R. Keynton, and A. El-Baz, "A deep learning-based approach for automatic segmentation and quantification of the left ventricle from cardiac cine mr images," *Computerized Medical Imaging and Graphics*, p. 101717, 2020.

[5] S. Dong, G. Luo, C. Tam, W. Wang, K. Wang, S. Cao, B. Chen, H. Zhang, and S. Li, "Deep atlas network for efficient 3d left ventricle segmentation on echocardiography," *Medical Image Analysis*, p. 101638, 2020.

[6] M. Al Noman, A. A. Hossain, and M. A. Rahman, "Initial point prediction based parametric active contour model for left ventricle segmentation of cmri images," in *Proceedings of International Joint Conference on Computational Intelligence*. Springer, 2012, p. 533.

[7] W. Kristanto and J.-P. Aben, "Left ventricle segmentation in contrast-enhanced cine mri datasets," May 2 2019, uS Patent App. 16/167,152.

[8] R. A. Moreno, M. F. de Sá Rebelo, T. Carvalho, A. N. Assunção, R. N. Dantas, R. do Val, A. S. Marin, A. Bordignom, C. H. Nomura, and M. A. Gutierrez, "A combined deep-learning approach to fully automatic left ventricle segmentation in cardiac magnetic resonance imaging," in *Medical Imaging 2019: Biomedical Applications in Molecular, Structural, and Functional Imaging*, vol. 10953. International Society for Optics and Photonics, 2019, p. 109531Y.

[9] H. Wei, W. Xue, and D. Ni, "Left ventricle segmentation and quantification with attention-enhanced segmentation and shape correction," in *Proceedings of the Third International Symposium on Image Computing and Digital Medicine*. ACM, 2019, pp. 226–230.

[10] J. Cong, Y. Zheng, W. Xue, B. Cao, and S. Li, "Ma-shape: Modality adaptation shape regression for left ventricle segmentation on mixed mr and ct images," *IEEE Access*, vol. 7, pp. 16 584–16 593, 2019.

[11] Y. Lan and R. Jin, "Automatic segmentation of the left ventricle from cardiac mri using deep learning and double snake model," *IEEE Access*, vol. 7, pp. 128 641–128 650, 2019.

[12] S. Moradi, M. G. Oghli, A. Alizadehasl, I. Shiri, N. Oveisi, M. Oveisi, M. Maleki, and J. Dhooge, "Mfp-unet: A novel deep learning based approach for left ventricle segmentation in echocardiography," *Physica Medica*, vol. 67, pp. 58–69, 2019.

[13] H.-Y. Lee, N. C. Codella, M. D. Cham, J. W. Weinsaft, and Y. Wang, "Automatic left ventricle segmentation using iterative thresholding and an active contour model with adaptation on short-axis cardiac mri," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 4, pp. 905–913, 2009.

[14] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[15] P. V. Tran, "A fully convolutional neural network for cardiac segmentation in short-axis mri," *arXiv preprint arXiv:1604.00494*, 2016.

[16] G. Luo, S. Dong, K. Wang, W. Zuo, S. Cao, and H. Zhang, "Multi-views fusion cnn for left ventricular volumes estimation on cardiac mr images," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 9, pp. 1924–1934, 2017.

[17] J. de Wit, "3rd place solution for the second national datascience bowl," 2016.

[18] e. a. Burms, Jeroen. Second annual data science bowl model documentation.https://github.com/317070/kaggle-heart/blob/master/documentation.pdf.

[19] H.-Y. Lee, "Automatic lv segmentation with k-means clustering and graph searching on cardiac mri," *International Journal of Computer and Information Engineering*, vol. 9, no. 3, pp. 292–295, 2015.

[20] R. Girshick, "Fast rcnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[21] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[22] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Transactions on image processing*, vol. 10, no. 2, pp. 266–277, 2001.

[23] Avendi. (2019) semi-automatic segmentation of lv in cardiac mr images. [Online]. Available: https://github.com/mravendi/LV-segmentation-in-cardiac-MRI

[24] C. Pluempitiwiriyawej, J. M. Moura, Y.-J. L. Wu, and C. Ho, "Stacs: New active contour scheme for cardiac mr image segmentation," *IEEE transactions on medical imaging*, vol. 24, no. 5, pp. 593–603, 2005.

[25] Avendi. (2019) Lv segmentation in cardiac mri. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/52090-mravendi-lv-segmentation-in-cardiac-mri

[26] J. de wit. (2015) Image segmentation with the u-net architecture, https://github.com/juliandewit/kaggle_ndsb2.

[27] J. de Wit. (2015) 3rd place solution for the second national datascience bowl,http://juliandewit.github.io/kaggle-ndsb/.

[28] E. Abdelmaguid, J. Huang, S. Kenchareddy, D. Singla, L. Wilke, M. H. Nguyen, and I. Altintas, "Left ventricle segmentation and volume estimation on cardiac mri using deep learning," *arXiv preprint arXiv:1809.06247*, 2018.

[29] A. W. Tencia Lee. (2016) left ventrile segmenation using deep learning. [Online]. Available: https://github.com/woshialex/diagnose-heart/blob/master/TenciaWoshialex_mode_documentation.pdf

[30] Q. L. Tencia Lee, "Automatic left ventricle volume calculation in cardiac mri using convolutional neural network," 2016.

[31] K. Patel. (2019) custom object detection using tensorflow from scratch. [Online]. Available: https://towardsdatascience.com/custom-object-detection-using-tensorflow-from-scratch-e61da2e10087

[32] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[34] S. C. Zhu and A. Yuille, "Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 18, no. 9, pp. 884–900, 1996.

[35] A. A. O. Azadeh Hadadi, *Left Ventricle Segmentation using K-means Clustering and Deep Learning*, UBFC, May 2020.