Deep learning algorithms for cardiovascular image analysis

Savioli, Nicolo

Awarding institution:
King's College London

Deep Learning Algorithms for Cardiovascular Image Analysis

DOCTORAL THESIS

Nicoló Savioli

Author:

Nicoló Savioli

Supervisors:

Dr. Pablo Lamata

Prof. Giovanni Montana

A thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy in the

Division of Imaging Sciences and Biomedical Engineering

School of Medicine

King's College London

April, 2019

Abstract

Deep Learning technologies are creating a revolution in the analysis of medical images. We are getting closer to the vision of a fully automated and reliable characterization of disease, both reproducing what the expertise of the radiologist is trained for, and proposing new metrics, revealing new patterns present in the data that are beyond the perceptual limitations of human beings. In this context, this Ph.D. thesis contributes towards this vision with specific solutions for both spatial and temporal analysis of two of the most prevalent modalities of medical imaging in cardiology, Ultrasound (US) and Magnetic Resonance Imaging (MRI).

The dissertation is started with a formal mathematical definition of the basic concepts of the most successful deep learning technology in the analysis of medical images: the Convolutional Neural Networks (CNN), specifically their back-propagation algorithm that is used for training. The Ph.D. thesis contributions are essentially based on the design and application of novel architectures of the CNNs. The first research Chapter and scientific publication propose a new deep learning model to extract the fetal aortic signal from an US video sequence. The architecture consists of three fundamental blocks: a convolutional layer for the extraction of imaging features, a Convolution Gated Recurrent Unit (C-GRU) for exploiting the temporal redundancy of a signal, and a novel regularized loss function, called CyclicLoss (CL). The method proposed achieves an accuracy far superior to the state of the art, providing an average reduction of the Mean Square Error (MSE) from $0.31mm^2$ to $0.09mm^2$, and execution speed of 289 frames per second.

The rest of the Ph.D. work is focused on the analysis of cardiac MRI. The second Ph.D. research Chapter, and second scientific contribution, proposes a solution for the segmentation of the left ventricle (LV) from CINE MRI images. The main idea is to learn from images

acquired through the entire cardiac cycle, instead of simply from keyframes. The workflow consists of three components: first, an automated localization and subsequent cropping of the bounding box containing the cardiac silhouette. Second, we identify the LV contours using a Temporal Fully Convolutional Neural Network (T-FCNN), which extends Fully Convolutional Neural Networks (FCNN) through a recurrent mechanism enforcing temporal coherence across consecutive frames. Finally, we further defined the boundaries using either one of two components: fully-connected Conditional Random Fields (CRFs) with Gaussian edge potentials and Semantic Flow. Our initial experiments suggest that significant improvement in performance (i.e. a 30% reduction in error metrics) can potentially be achieved by using a recurrent neural network component that explicitly learns cardiac motion patterns whilst performing LV segmentation.

The next Chapter and scientific publication propose an architecture called Volumetric Fully Convolution Neural Network (V-FCNN) with the aim of capturing the entire spatial anatomy of the atria in high-resolution MRI. V-FCNN is able to process eighty-eight slices in one-shot on available GPUs and consequently integrating the spatial redundancy through 3D-kernels. Learning outcomes are maximized with a loss function combining MSE and Dice Loss (DL) in order to both capture the bulk shapes and reduce over-segmentation, and training speed and convergence are also improved by removal of the skip-paths. The method achieves a Dice Index of 92.5 in the atrial segmentation task. Finally, the last contribution and publication propose a new network called Region Of Interest Generative Adversarial Network (ROI-GAN) that is tested in the problem of the Right Ventricle (RV) segmentation from MRI. In this context, the work first investigates the optimal combination of three concepts (C-GRU, the Generative Adversarial Networks (GAN), and the L1 loss function), achieving an improvement of 0.05 and 3.49 mm in DL and Hausdorff Distance respectively compared to the baseline FCNN. This improvement is then doubled by the ROI-GAN, that sets two GANs to cooperate working at two fields of view of the image; its full resolution and the region of interest (ROI). The rationale here is to better guide the FCNN learning by combining global (full resolution) and local Region Of Interest (ROI) features. The study is conducted in a large in-house dataset of 23,000 segmented MRI slices, and its generality is verified in a publicly available dataset.

Contents

List of Figures

List of Tables

1 Introduction

The objective of this thesis is mainly related to developing of new robust and accurate methods for cardiac images analysis. Especially, with the Magnetic Resonance Imaging (MRI) images make it possible to have a complete picture of cardiac function.

Whereas, fast and accurate segmentation of the Left Ventricle (LV), Right Ventricle (RV) and Atrium from MRI are considered gold standard biomarkers for the quantification of heart health, providing metrics such as Ejection Fractions (EF) and Stroke Volumes. The main challenges of MRI segmentation, are essentially due to tubercular and papillary muscle presence in the surrounding structures which present heterogeneity banding and motion artifacts.

Nevertheless, even if MRI has sufficient image quality for an optimal image processing it has also very costly equipment. On the other hand, Ultrasound (US) imaging has become more accessible due to lower costs but presents much noisier images. Therefore, is also desirable new ultrasound processing methods that break down noisy images problem.

Another fundamental aspect is to take into consideration the cyclic and temporal nature of the cardiovascular system. Then, including these temporal aspects, is accordingly possible to enhance the accuracy in which these methods are implemented.

A possible solution to embrace all those needs is Machine Learning (ML) a method being used in a variety of branches of applications.

Particularly, Deep Neural Network (DNN) is a Machine Learning (ML) technique that consists of different levels that have input, output and at least one hidden layer in between. Each hidden level performs specific operations of classification among congruent patterns allowing

the extraction of information from the structured or unstructured data.

The DNN has quickly become a methodology of choice for investigating medical images; where diverse acquisition methods have difficulties and benefits.

The advent of advanced supervised Deep learning (DL) algorithms has increased the speed and precision of cardiac analysis, although the RV shapes and myocardium segmentation are still very challenging [1].

Today, the Convolution Neural Network (CNN) is the most efficient mathematical model for extracting salience features from an input image. The CNN was inspired by a biological mechanism, where the neuron connectivity mimics the animal visual cortex organization [2], with the specific sensory neurons responding to visual stimuli in a specific region called the receptive field.

In the CNN the visual receptive field is modelled with filters, which are functions that progressively convolve the input image in order to extract salient characteristics. Every convolution layer of the networks downsizes the input image into smaller feature maps that could be restored to the original input size with a reverse convolution process.

In the problem of segmentation of medical images, CNN's have been widely used so that the features that contain segmentation information (encoding process), are restored to the original size through a set of up-sampling convolutions (decoding process). This architecture is referred to as a Fully Convolution Neural Network (FCNN) [3] or U-Net [4].

Unsupervised methods [5] can solve the tedious problem of creating enough labels, but at the same time the intrinsic anatomy (i.e spatial organ structure) and physiological properties (i.e temporal cyclic variation of the heart) of cardiovascular imaging data has a large level of redundancy across time and space.

While, the semi-supervised strategies are the most hopeful, as they take advantage of a training-set for manually labelled data in addition to searching for new structures led by the unsupervised techniques. Generative Adversarial Networks (GAN), display encouraging results in medical imaging, especially for segmentation problems [6, 7].

This thesis is organized into five Chapters with the target of solving three main tasks. The first task is the deep learning of methods for both spatial and temporal structure analysis. Secondly, better loss functions for also exploiting temporal and spatial variations. Thirdly, new generative methods to integrate different MRI resolutions in order to improve segmentation performances.

In Chapter two, a brief review of supervised feed-forward Neural Networks is given inside a formal mathematical definition of the back-propagation algorithm used for training the deep models.

Chapter three proposes a new deep learning model to extract fetal aortic signal variation from an US video sequence. The architecture consists of the following fundamental blocks: a convolutional layer for the extraction of imaging features, a Convolution Gated Recurrent Unit (C-GRU) for enforcing the temporal coherence across video frames and exploiting the temporal redundancy of a signal, and a regularized loss function, called *CyclicLoss* (CL). The strengths of this Chapter are predominately linked with the CL loss that imposes apriori knowledge about the periodicity of the observed signal; performed as an L2 norm between pairs of predictions at the same point of the heart cycle and from adjacent cycles. The T_{period} parameter is the period of the cardiac cycle determined through a peak detection algorithm, where the average of all peak-to-peak distances define its value. N_{cycles}, is the number of cycles present, calculated as the total length of the input signal divided by T_{period}. The CL is therefore combined with the Mean Square Error (MSE) as a regularization term.

As a result of this method an accuracy far superior to previously proposed methods has been achieved, providing an average reduction of the MSE from 0.31 mm^2 (state-of-art) to 0.09 mm^2, and a relative error reduction from 8.1% to 5.3% with a execution speed of 289 frames per second. A non-parametric test (KS-test) was also performed to verify the statistical significance of the proposed technique compared to other techniques.

Subsequently, the combination of C-GRU with recent state-of-the-art deeper networks ([8], [9]) and a shallow one (AlexNet [10]) was also performed. Furthermore, the CL loss proved to be performing better than all models studied here. Some weaknesses were noted, mostly relating to the small dataset available (2-5 cycles and 125 frames per patient). As time progresses, deep

networks will be required to handle the availability of new data. The current dataset will be made public for ensuring the reproducibility of results.

In Chapter four, the learning of the LV heart physiology (i.e cyclic heart movement over time) is tackled in the same fashion as the previous US video analysis problem with appropriate DL segmentation work-flow performed in three components. Firstly, the automated localization of LV. Secondly, the identification of LV counters using a Temporal Fully Convolution Neural Network (T-FCNN) which extends the FCNN through a GRU mechanism. Ultimately, additional boundaries with a post-processing technique such as fully connected Conditional Random Fields (CRFs) including Gaussian edge potentials [11] and Semantic Flow [12]. The use of the recurring unit led to a 30% reduction in the Average Perpendicular Distance (APD) [13] compared to the FCNN baseline.

The use of recurrent units is not a perfect solution since they are limited by vanishing gradients which can reduce the performance of the back-propagation over time. A common solution is the use of a gradient clipping strategy or adding a regularization term that improves or reduces the magnitude of the gradient [14].

In Chapter five, an innovative architecture called Volumetric Fully Convolution Neural Network (V-FCNN) is proposed; with the aim of capturing the entire spatial anatomy of the atria and mitigate the vanishing gradient problem of a recurrent unit. The V-FCNN is able to process 88 slices in one-shot on the available GPU, and consequently integrating the spatial redundancy through 3D-kernels with a novel loss function. The introduced loss function is based on a combination of both Mean Square Error (MSE) and Dice Loss (DL) in order to capture bulk shapes and reduce, at the same time, the errors produced by over-segmentation. However, the V-FCNN is a simplification of V-Net [15] by removal of the skip-path with the objective of achieving a faster training convergence.

The elimination of the skip-connections (i.e used for recovering spatial resolution lost by the pooling layer) and the pooling layers results in the processing of the entire volume slices in fast way through the initial down-sampling, though there are large losses of spatial information. This initial down-sampling is needed because the available GPU memory is insufficient to contain 88 high-resolution images in one input shot. The solution of this problem, that invalidates

the final accuracy. The V-FCNN achieved a Dice Index [13] of 92.5.

Though, the 3D-convolutional approach allows acquiring the three-dimensional structure without gradient problems, any other way is a very expensive approach for the Graphics Processing Unit (GPU) memory that necessitates a reduction in the input size, and consequently losing spatial resolution.

In the final Chapter, a new network called Region Of Interest Generative Adversarial Network (ROI-GAN) is proposed. The ROI-GAN works in two different fields of view: full MRI ROI resolution among two GAN's, where the GAN generators can also operate with GRU units.

This semi-supervised approach mitigates the absence of a large training set and concurrently improves the segmentation performance thus achieving an increase 0.05 Dice and 3.49 mm Index and Hausdorff Distance [16] each with regard to the FCNN baseline.

1.1 Impact

In the vision of a fully automated, robust and accurate analysis of medical images, this Ph.D. has contributed with i) loss functions for getting the temporal and spatial biological variation; ii) comparison of different CNN/FCNN architectures for ultrasound prediction and MRI segmentation; iii) an optimized FCNN for large 3D datasets that requires less training to achieve a reasonable performance; iv) new adversarial training strategy capable of cross-integrating full MRI resolution with the local one.

The thesis is correlated with the following list of publications:

- Nicoló Savioli, Silvia Visentin, Erich Cosmi, Enrico Grisan, Pablo Lamata, and Giovanni Montana. Temporal convolution networks for real-time abdominal fetal aorta analysis with ultrasound. *Springer International Publishing*, pages 148–157, 2018

- Nicoló Savioli, Miguel Silva Vieira, Pablo Lamata, and Giovanni Montana. Automated segmentation on the entire cardiac cycle using a deep learning work - flow. IEEE

Xplore:153–158, Oct 2018

- Nicoló Savioli, Giovanni Montana, and Pablo Lamata. V-fcnn: Volumetric fully convolution neural network for automatic atrial segmentation. In *Statistical Atlases and Computational Models of the Heart. Atrial Segmentation and LV Quantification Challenges*, pages 273–281, Cham, 2019. Springer International Publishing

- Nicolo Savioli, Miguel Silva Vieira, Pablo Lamata, and Giovanni Montana. A generative adversarial model for right ventricle segmentation. *arXiv:1810.03969*, Sep 2018

2 The Neural Networks

This Chapter describes the theory of Neural Networks and in particular the backpropagation algorithm. Convolutional networks are further explained along with the recurring networks. The problem of the vanishing gradient when training with the aid of the previous observations (i.e. recurrence) is also introduced. This will provide an overview of the key concepts within Deep Learning to act as a basis for further Chapters, where particular attention is paid to the mathematical explanation of the theory.

2.1 Feed-forward neural network overview

In the supervised learning problem, two training examples are given: $(x(i), y(i))$ where $x(i)$ is the input image and $y(i)$ the output target. The Neural Networks (NN's) provide the possibility to define a complex function $h_{W,b}(x)$ that transforms the input data into the output target depending on parameters w_i, b.

Particularly, the parameters w_i are called weights while b is the slope, referred to as the bias and they change during the training process. The NN is composed of different stacked layers l whit a specific hidden function $h^l_{w,b}$ defined as:

$$h^l_{w,b}(x) = \sigma(\sum_{i=1}^{N} w^l_i x_i + b) = \sigma((W^l)^T * x + b) \tag{2.1}$$

Where $\sigma : R \to R$ is the activation function; usually a sigmoid function is chosen, with N being the number of connections between weights. Equation 2.2 is an example of a sigmoid function

with input **z**.

$$\sigma(z) = \frac{1}{1+exp(-z)} \tag{2.2}$$

The choice of a sigmoid function has some benefits during the derivation of the NN, as the derivative can be expressed as a product of the sigmoid function itself.

$$\sigma'(z) = \sigma(z)[1-\sigma(z)] \tag{2.3}$$

A NN with three neurons is shown in Fig 5.1 serve as to a base example which can be generalised for more connections.

The left hand layer, denoted layer L1, is referred to as the input layer while the right hand layer, denoted as layer L3, is referred to as the output layer. The central layer, denoted as layer L2, is referred to as the hidden layer. This is due to the internal values not being fully observable during the training process.

The step of generating the output from an input, governed by the following set of equations 2.4, is called Forward Propagation (FP).

$$a_1^2 = \sigma(W_{11}^1 x_1 + W_{12}^1 x_2 + W_{13}^1 x_3 + b_1^1) \tag{2.4}$$

$$a_2^2 = \sigma(W_{21}^1 x_1 + W_{22}^1 x_2 + W_{23}^1 x_3 + b_2^1) \tag{2.5}$$

$$a_3^2 = \sigma(W_{31}^1 x_1 + W_{32}^1 x_2 + W_{33}^1 x_3 + b_3^1) \tag{2.6}$$

$$h_{w,b}(x) = \sigma(W_{11}^2 a_1^2 + W_{12}^2 a_2^2 + W_{13}^2 a_3^2 + b_1^2) \tag{2.7}$$

Where W_{ij}^l denotes the weights parameters between the neurons i in precessing layer l and the corresponding neurons j in layer $l+1$; while b^l is the bias associated with the specific layer. The a_i^l denotes the activation function of neurons i in layer $l+1$.

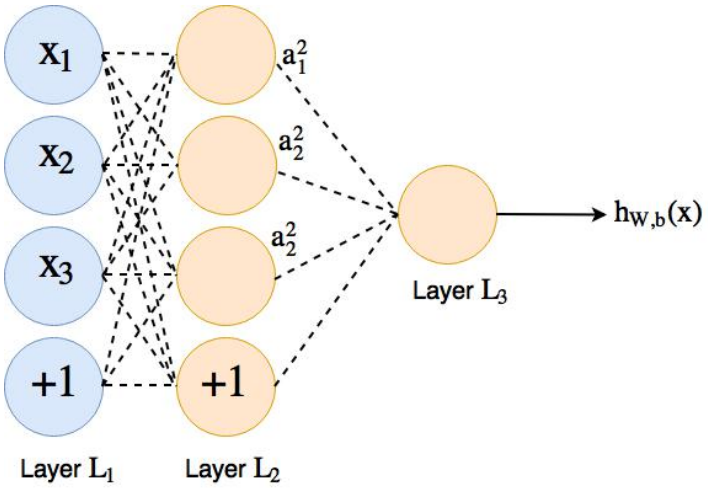

Figure 2.1 The figure shows a feed-forward neural network structure. In particular, we have three layers: the input layer l_1, the hidden l_2 and the output l_3. The circles labelled "+1" are referred to as bias units b that for simplicity is taken constant.

2.2 Back-propagation Algorithms

The challenge is to optimize the set of coefficients of the network for a given $(x^1, y^1), ..., (x^m, y^m)$ m random training examples. The NN can be trained with the Stochastic Gradient Descent (SGD) algorithm [21] technique. Then, for a single training observation (x, y) the loss function is defined as:

$$J(W, b; x, y) = \frac{1}{2}||h_{W,b} - y||^2 \tag{2.8}$$

The cost function $J(W, b; x, y)$ returns as an output a value which is used for updating all W for the parameters on each NN layer. This is formulated as a loss function minimization problem with respect to W^l and b^l for all l specific layers.

Then, for a given layer l the update rule is:

$$W^l = W^l - \alpha \frac{\partial J}{\partial W^l} \tag{2.9}$$

$$b^l = b^l - \alpha \frac{\partial J}{\partial b^l} \tag{2.10}$$

The α constant is called the Learning Rate (LR) that controls how much the weight parameters are adjusted with respect to the loss.

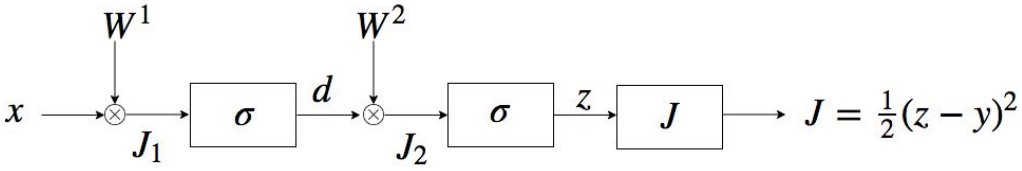

Figure 2.2 The figure shows a block diagram of a NN where x is the NN input, W^1 and W^2 the weights of the two specific layers l. J_1 and J_2 the multiplication between each weights at every input block. Finally, J the loss function between the output z and the target y.

The back-propagation algorithm is described here. A schematic NN of two layer within parameters W^1 and W^2 is chosen(see Fig. 2.2) for solving the following minimization problem:

$$\begin{cases} W^1 = W^1 - \alpha \frac{\partial J}{\partial W^1} \\ W^2 = W^2 - \alpha \frac{\partial J}{\partial W^2} \end{cases} \tag{2.11}$$

For the simplification of notation:

$$z = h_{W.b} \tag{2.12}$$

Then, looking at Fig. 2.2 we want to find the analytical description of how the loss function depends on the parameters W.

From the start of the output of the network, z, going step by step backwards from the loss function to W_1, the first variable z (i.e output of the network) is outputted. Here a partial derivative of J with respect to W^2 is required utilising the chain rule, defined as:

$$F^{'}(x) = f^{'}(g(x))g^{'}(x) \tag{2.13}$$

The chain rule can be also written in Leibniz notation:

$$\frac{dz}{dx} = \frac{dz}{dy}\frac{dy}{dx} \tag{2.14}$$

In this notation $z = f(y)$ and $y = g(x)$ then:

$$\frac{dz}{dx} = \frac{dz}{dy}\frac{dy}{dx} = f^{'}(y)g^{'}(x) = f^{'}(g(x))g^{'}(x) \tag{2.15}$$

That applied to Eq 2.11

$$\frac{\partial J}{\partial W^2} = \frac{\partial J}{\partial z}\frac{\partial z}{\partial W^2} \tag{2.16}$$

Where $\frac{\partial J}{\partial z}$ is the derivative of the loss function with respect to z:

$$\frac{\partial J}{\partial z} = (z - y) \tag{2.17}$$

The chain rule is also applied to $\frac{\partial z}{\partial W^2}$, as the J_2 variable is found first.

$$\frac{\partial J_2}{\partial W^2} = \frac{\partial z}{\partial J_2}\frac{\partial J_2}{\partial W^2} \tag{2.18}$$

Where since $J_2 = W^2 * d$ then:

Since $J_2 = W^2 * d$, it follows on to Eq 2.19

$$\frac{\partial J_2}{\partial W^2} = d \tag{2.19}$$

Therefore, for the second term of 2.11 the $\frac{\partial J}{\partial W^2}$:

$$\frac{\partial J}{\partial W^2} = (z-y)\frac{\partial z}{\partial J_2}d \tag{2.20}$$

While, for the first term of 2.11 is the same here for $\frac{\partial J}{\partial W^2}$:

$$\frac{\partial J}{\partial W^1} = \frac{\partial J}{\partial z}\frac{\partial z}{\partial W^1} = (z-y)\frac{\partial z}{\partial J_2}\frac{\partial J_2}{\partial W^1} \tag{2.21}$$

The $\frac{\partial J_2}{\partial W^1}$ must be calculated, but since d is the first variable that the propagation signal meets; for the chain rule:

$$\frac{\partial J_2}{\partial W^1} = \frac{\partial J_2}{\partial d}\frac{\partial d}{\partial W^1} \tag{2.22}$$

However, $J_2 = W^2 d$ as well as:

$$\frac{\partial J_2}{\partial d} = W^2 \tag{2.23}$$

Where it is possible to decompose $\frac{\partial d}{\partial W^1}$ in:

$$\frac{\partial d}{\partial W^1} = \frac{\partial d}{\partial J_1}\frac{\partial J_1}{\partial W^1} \tag{2.24}$$

Thus $J_1 = W^1 x$ and it leads on from this that:

$$\frac{\partial J_1}{\partial W^1} = x \tag{2.25}$$

Finally, the system from 2.11 becomes:

$$\begin{cases} W^1 = W^1 - \alpha(z-y)\frac{\partial z}{\partial J_2}W^2\frac{\partial d}{\partial J_1}x \\ W^2 = W^2 - \alpha(z-y)\frac{\partial z}{\partial J_2}d \end{cases} \tag{2.26}$$

Then, the calculation of the derivatives of $\frac{\partial z}{\partial J_2}$ and $\frac{\partial d}{\partial J_1}$ are still required for the convenient sigmoid function given in Eq 2.2).

$$\frac{\partial d}{\partial J_1} = \frac{\partial}{\partial J_1}\frac{1}{1+e^{-J_1}} = \frac{\partial}{\partial J_1}\sigma(J_1) = \sigma(J_1)[1-\sigma(J_1)] = d(1-d) \tag{2.27}$$

$$\frac{\partial z}{\partial J_2} = \frac{\partial}{\partial J_2}\frac{1}{1+e^{-J_2}} = \frac{\partial}{\partial J_2}\sigma(J_2) = \sigma(J_2)[1-\sigma(J_2)] = z(1-z) \tag{2.28}$$

This means that the derivative of the output, with respect to the input, is expressed in terms of the output itself due to the sigmoid function.

$$\begin{cases} W^1 = W^1 - \alpha[(z-y)z(1-z)W^2 d(1-d)]x \\ W^2 = W^2 - \alpha[(z-y)z(1-z)]d \end{cases} \tag{2.29}$$

The bias equation is obtained with the same procedure. The general back-propagation algorithm is divided into six steps:

- **Input** x set the activation a^1 for the specific input layer.
- **Feedforward**: For each $l = 2,3,...,L$ compute $z^l = w^l a^{l-1} + b^l$ and $a^l = \sigma(z^l)$;
- **Delta error δ^L for the last layer** L: calculate as the derivative vector $\delta^L = J_a^{'} \cdot \sigma^{'}(z^L)$;
- **Delta error back-propagation**: For each layer $l = L-1, L-2, ..., 2$ must be determined $\delta^l = ((W^{l+1})^T \delta^{l+1} \cdot \sigma^{'}(z^l))$;
- **Compute the layer gradient**: $\frac{\partial J}{\partial w^l_{jk}} = a^{l-1}_k \delta^l_j$ and $\frac{\partial J}{\partial b^l_j} = \delta^l_j$;
- **Update each layer**: $W^l = W^l - \alpha\frac{\partial J}{\partial w^l_{jk}}$ and $b^l = b^l - \alpha\frac{\partial J}{\partial b^l_j} = \delta^l_j$

Indeed, an α value too small can lead to a slow convergence, while large values can cause the local minimum to be missed by over-stepping.

New update layer algorithms are then applied to bypass old-fashioned learning rate selection.

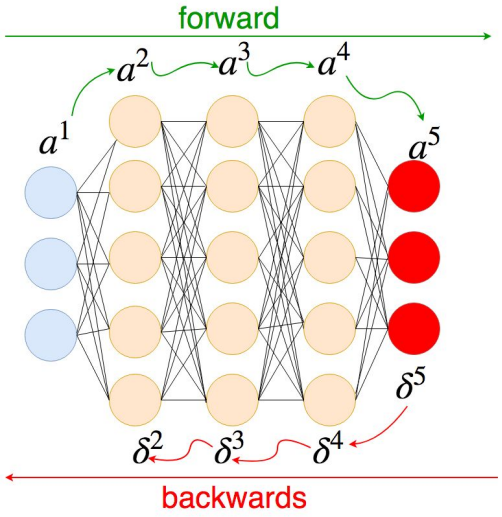

Figure 2.3 The figure shows the steps of the back-propagation algorithm. First the forward a^l activations are calculated for each layer l, then after the loss computation the error term δ^{l+1} is computed.

The most famous method is the Adagrad algorithm [22], and will be explained in the next section.

2.3 Computing the gradients: the Adagrad algorithm

The key is to compute the gradients, the partial derivatives. The Adagrad algorithm is the solution, and also avoids the need to fix an a-priory LR

Then, at every training time t, in the (x^t, y^t) dataset iteration with an initial LR, the Adagrad algorithm computes the g_t value.

The g_t denotes the gradient at time t with $g_{t,l}$ the partial derivative of the loss function with respect to the layer $J(W^{l,t}, b^{l,t}; x^t, y^t)$. Where $W_{l,t}$ are specific parameters of NN layer l at iteration t.

$$g_{t,l} = \nabla_W J(W_{t,l}, b_{t,l}; x, y) \tag{2.30}$$

Then, for the update step:

$$W_{t+1}^{l} = W_{t}^{l} - \alpha g_{t,l} \tag{2.31}$$

Adagrad automatically tunes the general LR as:

$$W_{t+1}^{l} = W_{t}^{l} - \frac{\alpha}{\sqrt{G_{t,ii} + \varepsilon}} g_{t,l} \tag{2.32}$$

$G \in R^{d \times d}$ is a diagonal matrix of dimension $d \times d$ in which, each diagonal element, (i,i) is the sum of the square gradients $g_{t,l}$ at time t plus ε, a smoothing term.

In recent years, different modifications of Adagrad have been proposed, such as Adadelta [23] that impose a size window of w in order to restrict the quantity of previous gradient $g_{t,l}$ to accumulate in order to overcome the monotonic decreasing of learning rate.

Adaptive Moment Estimation (Adam) [24] has also been proposed to not just include previous gradients but to also include their exponentially decaying averages of past gradients.

2.4 Convolution Neural Network (CNN)

The CNN is a type of NN has taken inspiration from neuro-biology, derived specifically from the work of Hubel and Wiesel in the Visual Cortex (VC) of cats [25]. In the VC, it is possible to find a complex system of small sensitive cells organised in sub-regions positioned to cover the entire visual input space.

Those cells can be characterised as the local space filters, they are used for exploiting the surrounding spatial correlation present in natural input images. The CNN mathematically emulates this space filtering operation of the human VC that represents a powerful vision system.

There have been numerous examples of neuro-biology inspired vision systems that have been presented: NeoCognitron [26], Lenet-5 [27] and HMAX [28]. New studies have shown that CNN should be considered the best candidate for most visual recognition tasks [29]. The CNN

is composed of a sequence of hidden layers. More generally, every hidden layer forms a block of three main elements.

2.4.1 The Convolution Layer

The Convolution Layer (CL) is formed by a set of spatial filters. Each of these filters convolves across the spatial image location to produce a 3D tensor of activation maps called Feature Maps (FM). Each FM contains the location response of the input image. The 2D discrete convolution operation is given by the following equation:

$$\omega[x,y]^{l-1} * w[x,y]^{l} = \sum_{u=-k}^{k} \sum_{v=-k}^{k} \omega[u,v]^{l-1} W[x-u, y-v]^{l} \tag{2.33}$$

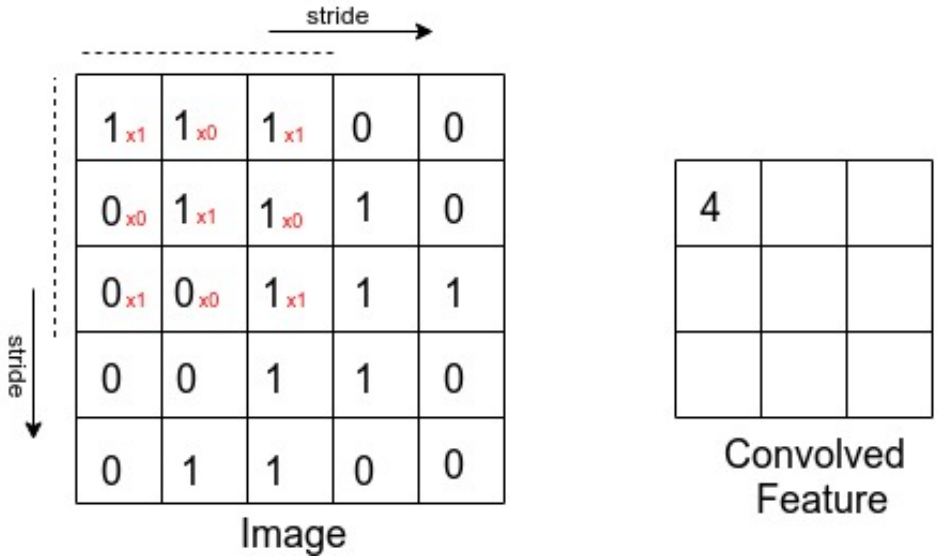

Figure 2.4 Example of a convolved operation. Every 3×3 kernel (red indices) is convolved with a 5×5 image with a dot product repetition. The stride indicates how many units the weight filters are moving each step. Finally, the kernel (weights) will be updated through the back-propagation algorithm.

The 2D discrete convolution can be generalized as a dot operator repeated several times between the $l-1$ layer input $\omega[x,y]^{l-1}$ and the learn-able weight matrix (kernels) $W[x,y]^{l}$ of destination layer l. Where $*$ is the convolution operator and x, y is the corresponding index of the pixels for both input features and kernels to be trained. Please note that the notation ω represents the greek letter omega and is different from W; while k the filter size. Other parameters

to be considered are stride and padding. While the stride controls how many units the weights w shifts on the image input, the padding adds a number of zeros around the features border for restoring the lost size after the convolution process.

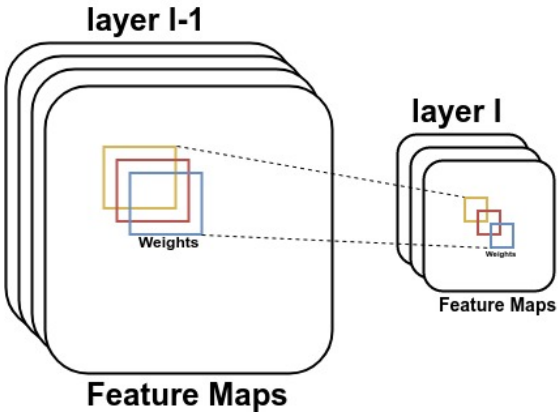

Figure 2.5 Example of a convolutional layer

The formula for determining the output feature size, for each given convolution layer, is:

$$O = \frac{W - K + 2P}{S} + 1 \tag{2.34}$$

Where O is the feature output, (height/length), W is the input size (height/length), K is the filter size, P is the padding, and S is the stride. Further, the padding P, can be also used for restoring the size of down-sampling features with an up-sampling path such as in U-Net [30] and Fully Convolution Neural Network [31]. For example, if the final CNN layer has a set of feature maps with a size of 22 pixels, it is then possible to progressively restore their size with a set of deconvolutions by an appropriate padding (i.e for doubling the 22 pixels to 44, a P value of 12 zeros with a kernel size of 3×3 and a stride of 1 is needed).

2.4.2 Activation functions

During the training of a neural network the magnitude of each the gradients at the corresponding layer becomes exponentially small part due to the minimization process. This is translates into very slow learning process. For example the magnitude of the sigmoid derivative is below 1.0 in the whole range of the function and this creates the gradient vanishing.

2.4.2.1 Rectified Linear Unit (ReLU)

The Rectified Linear Units (ReLU) layer is a common non-linear activation function for convolutional hidden units, which is used to improve the training of deeper CNN networks. In particular, it allows overcoming the problem of gradient disappearance by a sparse representation with true zeros that seems to be more suitable for neural networks [32].

The mathematical description of ReLU is given by the following equation:

$$ReLU(x) = \begin{cases} x, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases} \tag{2.35}$$

Where x is the output of the convolution layer and α is the constant slope.

2.4.2.2 Leaky ReLU (LReLU)

Differently, from ReLU, Leaky ReLU presents a small slope from negative values. Having zero-slope parts allows it to make the training faster.

$$LeakyReLU(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.01x, & \text{otherwise} \end{cases} \tag{2.36}$$

2.4.2.3 Parametric ReLU (PReLU)

The PReLU is a type of Leaky ReLU that instead of having a fix slop (i.e 0.01) makes it a parameter of the neural network (i.e α).

$$PReLU(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha x, & \text{otherwise} \end{cases} \tag{2.37}$$

2.4.3 Pooling

The Pooling (P) layer is a down-sampling operation across spatial dimensions after the convolution layer operation. The aim of pooling is to decrement the number of parameters and the total network computation time. The P operator does not require any parameters and can be interpreted as an operator that divides each Feature Map (FM) into subregions of size s_w and s_h.

For each subregion s_w and s_h of the input FM, f_k, the max value is taken and run for all features K_s. The 2D max pooling operators $H(x,y)$ are expressed as:

$$H(x,y) = \sum_{k \in K_s} \sum_{m \in s_w} \sum_{n \in s_h} max(f_{s_m,s_n})_k \tag{2.38}$$

Where $max(\cdot)$ is a max function.

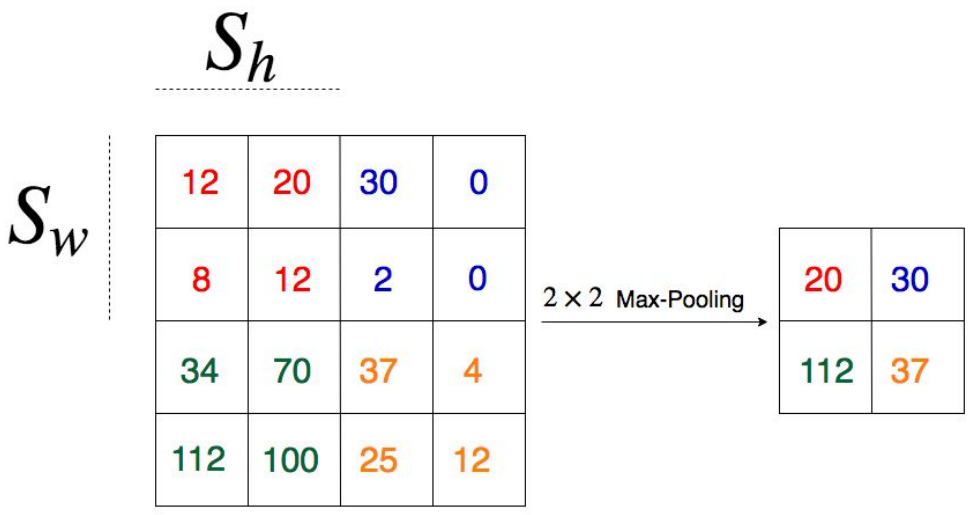

Figure 2.6 The figure shows the max pooling layer mechanism in practice.

2.4.4 Fully Connected Layer

After the last convolution layer, all 2D feature maps are flattened in a 1D vector where every neuron in the layer $l-1$ is connected with every other neuron of the layer l. This network can

be seen as a regular neural network where all nodes are connected to all previous activations. Frequently, it is possible to express Fully Connected (FC) Layers as a 1×1 convolution.

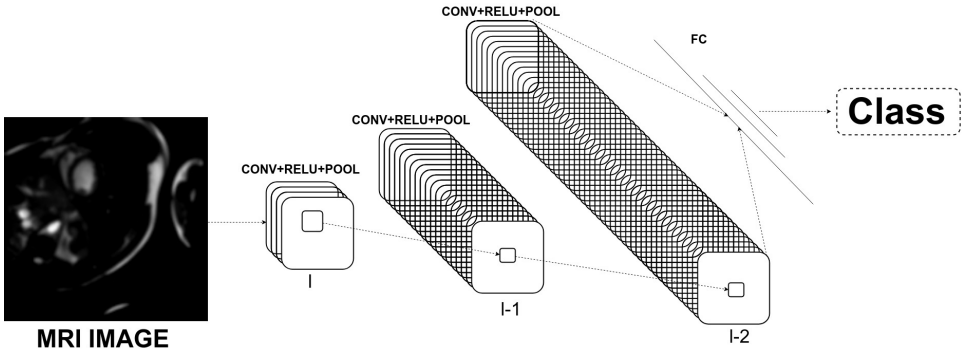

Figure 2.7 Example of a Convolution Neural Network

2.5 Backpropagation with CNN

This section provides a description of the back-propagation through a CNN. As a standard NN, this is divided in two-steps: forward and backward (see Fig 2.8). In the forward pass a set of i input features $\omega_i[x,y]$ at layer $l-1$ are convolved with $W_i[x,y]$ weight kernel in order to obtain $\omega^l_{features}[x,y]$ through the convolution equation 2.33, with x,y the corresponding spatial pixels. Then all FM enter into the sigmoid function σ (eq 2.2) and pooling layer $H(x,y)$ (Eq 2.38) with x,y being the corresponding spatial pixel indices; obtaining respectively for ω^l_σ and $\omega^l_{pooling}$:

$$\omega^l_{features} = \sum_{i \in K_s} \sum_{u=-k}^{k} \sum_{v=-k}^{k} \omega_i[u,v]^{l-1} W[x-u, y-v]^l_i \tag{2.39}$$

$$\omega^l_{sigma} = \sigma(\omega^l_{features}) \tag{2.40}$$

$$\omega^l_{pooling} = H(\omega^l_{sigma}) \tag{2.41}$$

The backward process is still similar to the standard NN, where each node of the matrix W_{ji}

is fully described by a set of 2D kernels (size $k \times k$). Moreover, the pooling error needs to be up-sampled with the convolution size of $convSize \times convSize$. Thus, given $\delta^{l+1}_{pooling}$ a matrix of derivatives among pooling pixel position x_i, y_i of size $\frac{convSize}{S_w} \times \frac{convSize}{S_h}$; the max index position (at sub feature region $S_w \times S_h$) is defined as:

$$\delta^{l+1}_{pooling} = \frac{\partial \omega^{l+1}_{pooling}}{\partial x_i \partial y_i} = \begin{cases} 1, & \text{if } max(x_{S_w}) = x_i \\ 1, & \text{if } max(y_{S_h}) = y_i \\ 0, & \text{otherwise} \end{cases} \tag{2.42}$$

We defined $J \in R^{S_w \times S_h}$ as a matrix of ones, the the Kronecker product between $\delta^{l+1}_{pooling}$ and $J_{S_w \times S_h}$ produce a $\delta^{l+1}_{Upsampling}$ of size $\frac{convSize}{S_w} S_w \times \frac{convSize}{S_h} S_h$ or rather $convSize \times convSize$ then:

$$\delta^{l+1}_{Upsampling} = \delta^{l+1}_{\frac{convDim}{S_w} \times \frac{convDim}{S_h}} \otimes J_{S_w \times S_h} \tag{2.43}$$

Finally, the δ^{l}_{Conv} is obtained from:

$$\delta^{l}_{Conv} = (W^{l+1})^T * \delta^{l+1}_{Upsampling} * \sigma^{'}(\omega^{l}_{features}) \tag{2.44}$$

Finally, the update functions are computed:

$$W^l = W^l - \alpha(\omega^{l-1}_{features} * \delta^{l}_{Conv}) \tag{2.45}$$

$$b^l = b^l - \alpha \delta^{l}_{Conv} \tag{2.46}$$

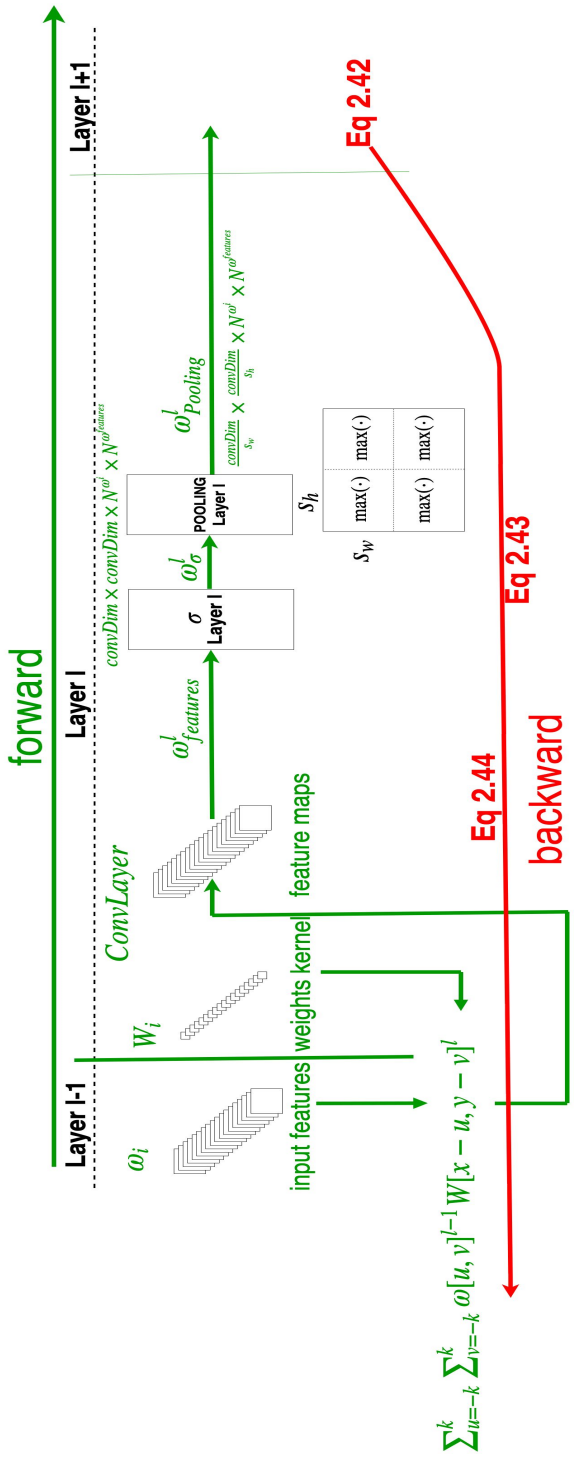

Figure 2.8 The figure shows the entire forward and back-propagation process through the convolution, activation function, and pooling layers.

2.6 Recent developments of CNN's

Deep-learning has historically developed many different types of CNN's, however only five of them have become very popular.

The first CNN taken into consideration is AlexNet [10]. The AlexNet is composed of five CL's with 11×11 Receptive Field (RF) size in the first layer, 5×5 in the second and finally 3×3 in the last three layers. Each CL is repetitively by ReLU and Pooling.

On the contrary, the VGG CNN model [33] is formed by a sequence of multiple convolution blocks with RF size of 3×3. This has two advantages: First, 3×3 filters are able to capture smaller details in the image. The ability to emulate the effect of a larger receptive field (i.e 11×11).

The VGG model is very expensive at runtime and doesn't appear scalable. Then, to overcome this problem, Google® proposed the Inception Module (IM) (Fig. 2.9 a); hence the name Inception CNN. In the IM, multiple CL of different size (1×1,3×3,5×5) are processed in parallel and combined together at the end of this. Where before each parallel CL has used a small 1×1 convolution filter in order to reduce the FM number with a consequent increase in network execution speed. Particularly, in this thesis, we use the new Inception V4 version [9].

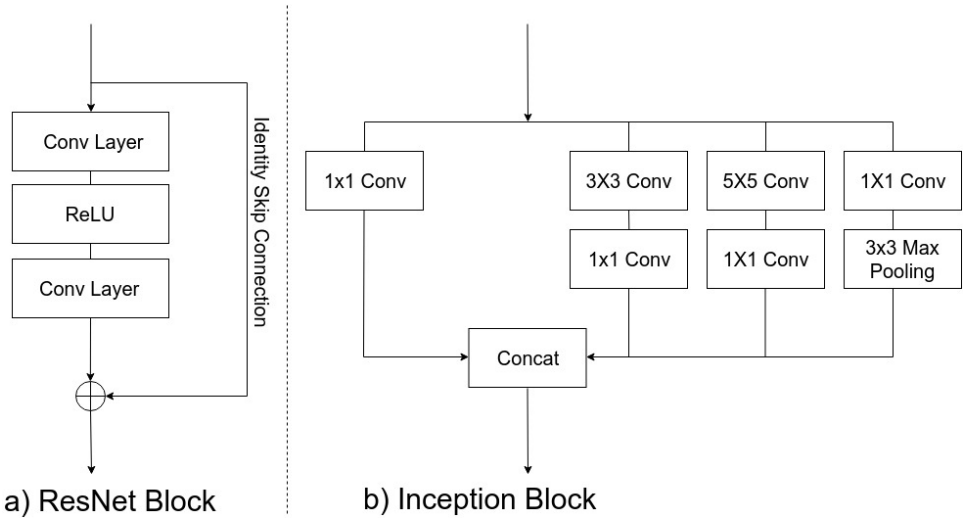

Figure 2.9 The figure shows two different blocks: a) ResNet and b) Inception.

The difference between the V4 version with the previous one is due to it's combination with another module called ResNet Module (RNM), from which is derived from the ResNet CNN by Microsoft® (Fig. 2.9 b). The simple idea that lies behind ResNet [34] is to pass the input between two successive cascade CL with an Identity Skip Connection (ISC), where both outputs of the last convolution and the ISC are added together. This allows the gradient to run through many layers without disappearing. Using the ISC it is therefore possible to train neural networks with thousands of layers.

Finally, an extension of the ResNet architecture is DenseNet [8]. In DenseNet each layer receives a signal from all preceding layers and, at the end, the input is combined by a concatenation. Similar to ResNet, the ISC is used to transfer all previous signals on the dense deepest levels. There are three main advantages to DenseNet: Firstly, a strong gradient flow, so the early layers can have more direct supervision from the final classification layer. Secondly, a huge number of parameters but with excellent computation efficiency. Thirdly, the DenseNet maintains features of high complexity because of a combination of different convolution levels.

2.7 Recurrent Neural Networks

Sequential information needs a specific neural network framework for learning temporal dynamics [35] where consecutive patterns are stored in a linear recurrent unit h_{t-1} in three steps. Firstly, the input x is carried into the recurrent unit by its parameters W_{xh}. Secondly, the hidden unit h_{t-1} is in a dot product within the recurrent parameters W_{xh}. Then, the summation of these two operation updates the new hidden state h_t with the below equation.

$$h_t = \sigma(W_{hh}h_{t-1} + W_{xh}x_t) \tag{2.47}$$

Once the actual state h_t is calculated the final output is given with the following equation:

$$y = W_{hy}h_t \tag{2.48}$$

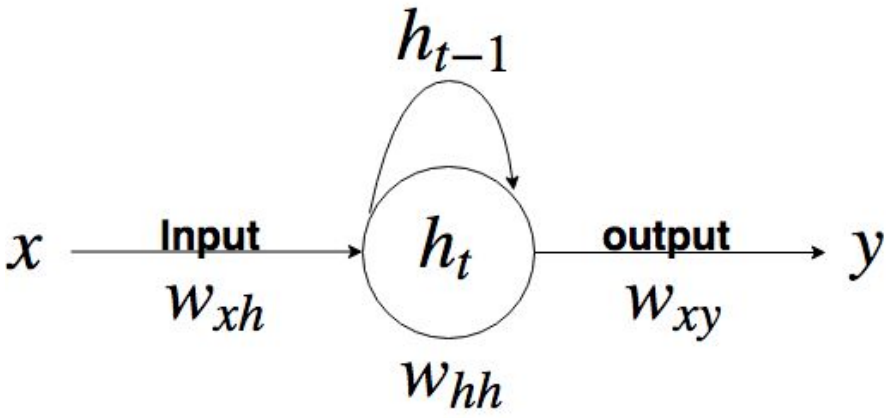

Figure 2.10 Schematic example of a Recurrent Neural Network

In order to train the RNN model the back-propagation algorithm is applied, the derivative of each recurrent node h_t dependent on all the previous steps (i.e h_{t-1}, h_{t-2}). The RNN network is unrolled in the total number of time steps, S, and loss, J.

$$\frac{\partial J}{\partial_{W_{hh}}} = \sum_{t=1}^{S} \frac{\partial J}{\partial_{W_{hh}}} \tag{2.49}$$

If the chain rule is applied:

$$\frac{\partial J_t}{\partial_{W_{hh}}} = \sum_{t=1}^{S} \sum_{k=1}^{t-1} \frac{\partial J}{\partial y_t} \frac{\partial y_t}{\partial h_t} \frac{\partial h_t}{\partial h_k} \frac{\partial h_k}{\partial W_{hh}} \tag{2.50}$$

Where the index k indicates the specific RNN hidden layer from time $k = 1$ to $k = t - 1$. The term $\frac{\partial h_t}{\partial h_k}$ denotes the derivative of the hidden layer at time t with respect to the hidden layer at time k, expressing this as the Jacobian product.

$$\frac{\partial h_t}{\partial h_k} = \prod_{i=k+1}^{t} \frac{\partial h_i}{\partial_{h_{i-1}}} = \prod_{i=k+1}^{t} W_{hh}^T diag[\sigma^{'}(h_{i-1})] \tag{2.51}$$

An issue arises from the multiplication of this Jacobian term many times during back-propagation. To demonstrate this we take the norm of both parameters W_{hh}^T and $diag[\sigma^{'}(h_{i-1})]$ corresponding to the $\gamma_{W_{hh}}$ and γ_{σ}, that are their maximum eigenvalues.

$$||\frac{\partial h_i}{\partial h_{i-1}}|| \leqslant ||W_{hh}^T||||diag[\sigma^{'}(h_{i-1})]|| \leqslant \gamma_{W_{hh}}\gamma_{\sigma} \tag{2.52}$$

Then, $\frac{\partial h_t}{\partial h_k}$ is bounded by the eigenvalues product of $\gamma_{W_{hh}}\gamma_{\sigma}$ and consequently the result is the following function:

$$||\frac{\partial h_i}{\partial h_{i-1}}|| \leqslant (\gamma_{W_{hh}}\gamma_{\sigma})^{t-k} \tag{2.53}$$

Accordingly, if $\frac{\partial h_t}{\partial h_k} \geqslant 1$, we have gradient explosion as the exponent becomes a very large number.

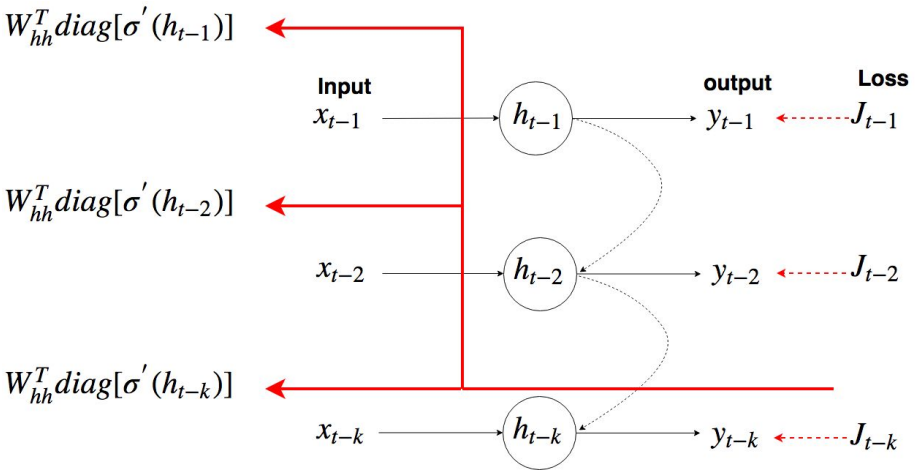

Figure 2.11 Example of RNN back-propagation through time, the red line represents the loss signal.

2.8 Solutions for the exploding gradient

Several methods have been used in the past to attack the gradient explosion (or vanishing) for long input sequences.

Initially, an L1 or L2 loss were added for penalizing the recurrent weights in order to contain the spectral radius of Eq 2.52, where the model can exhibit a loss of long-term memory traces when a long temporal sequence is given. Better learning algorithms or clipping gradients have proved useful to solve this problem[36].

Other solutions are linked to the structural change of the RNN model such as the Long Short-Term Memory Unit (LSTM) [37] or Gated Recurrent Unit (GRU) [38]. In both models, a special set of RNN units called gates behave as a binary (0 or 1) switch that directs the information to the memory, deletes it or addresses the new state. The opening and closing of these binary gates are learned directly from the data. In this section, we will deal with them in detail.

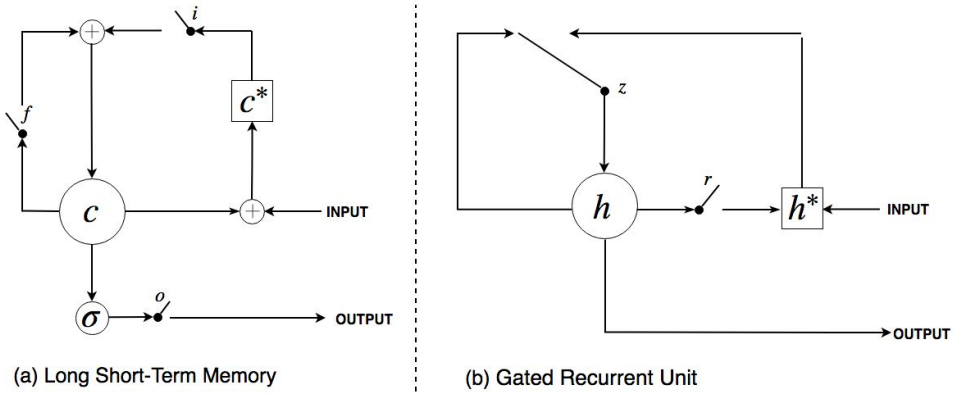

Figure 2.12 The figure shows a schematic diagram of (a) LSTM and (b) GRU. (a) i, f, o are the input, forget and output gates. c and c^* indicate the memory cell and the new activation cell. (b) r, z are the reset and update gates, and h and h^* are the previous recurrent state and the candidate state.

2.8.1 Long Short-Term Memory Unit

The Long Short-Term Memory (LSTM) (see Fig 2.12 (a)) maintains a memory, c, at time t. Again, for a given input x_t and recurrent hidden layer, h_t, the LSTM equations are computed:

$$h_t = o_t tanh(c_t) \tag{2.54}$$

Where o_t is the *output gate* that regulates the volume information in output from the c cell memory. The output gate is computed by:

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \tag{2.55}$$

The memory cell c is updated by deleting the existing memory while the new memory information c^* is added in to.

$$c_t = f_t c_{t-1} + i_t c_t^* \tag{2.56}$$

This new information c^* is modelled as:

$$c_t^* = tanh(W_{c^*x} x_t + W_{c^*h} h_{t-1} + b_{c^*}) \tag{2.57}$$

Where memory cell c is controlled by the interaction of two gates: the forget gate and the input gates. While the forget gate tries to forget the information no longer needed by the cell; the input gates decided how much of it must be added. Both are computed with the following equations:

$$f_t = \sigma(W_{fx} x_t + W_{fh} h_{t-1} + b_f) \tag{2.58}$$

$$i_t = \sigma(W_{ix} x_t + W_{ih} h_{t-1} + b_i) \tag{2.59}$$

Note σ is the activation function, while W_{ox}, W_{oh}, W_{c^*x}, W_{c^*h}, W_{fx}, W_{fh}, W_{ix} and W_{ih} are all recurrent weight matrices of *o output gate*, c^* memory, *f forget gate*, *i input gate* respectively. The corresponding bias outputs are b_o, b_c and b_g.

2.8.2 Gated Recurrent Unit

The Gated Recurrent Unit (GRU) (see Fig 2.12 (b)) is a simplification of LSTM for catching those dependencies faster. Likewise to the LSTM unit, the GRU has a number of gates that change the flow of information inside each unit but without a specific cell.

The activation unit h_t at time t is given as the interpolation between the previous activation h_{t-1}

and the candidate activation h_t^*:

$$h_t = (1 - z_t)h_{t-1} + z_t h_t^* \tag{2.60}$$

Where z_t is the *update gate* that decides how much information flows through the unit h_t and is computed as:

$$z_t = \sigma(W_{zx}x_t + W_{zh}h_{t-1} + b_z) \tag{2.61}$$

While the candidate activation h_t^* is measured the same as in the RNN unit:

$$h_t^* = tanh(W_{h^*x}x_t + W_{h^*h}(r_t \odot h_{t-1}) + b_{h^*}) \tag{2.62}$$

Where r_t is the *reset gate* and $\odot$ is the element-wise multiplication. When r_t is off the gate in Fig 2.12 (b) is open to 0. This mechanism acts like the LSTM *forged gate* as it allows you to decouple h_{t-1} from the h^* that is updated only by the input sequence x_t. The *reset gate* is computed in the same fashion as the *update gate*.

$$r_t = \sigma(W_{rx}x_t + W_{rh}h_{t-1} + b_r) \tag{2.63}$$

Where W_{zx}, W_{zh}, W_{h^*x}, W_{h^*h}, W_{rx} and W_{rh} are all recurrent matrices. While b_z, b_{h^*} and b_r the bias values.

2.8.3 Generative Adversarial Networks (GAN)

The GAN can be seen as variational auto-encoder [39] where given a set of i.i.d data samples $(x_1, x_2, ..., x_n)$, p_g generator distribution and p_d a data distribution.

The optimal discriminator $d(\cdot)$ for k different data samples and for a fixed generator $g(\cdot)$ is

given by:

$$d(x_i) = \sum_{i=1}^{k} \frac{p_d(x_i)}{p_d(x_i) + p_g(x_i)} \tag{2.64}$$

Demonstrated by the following Shannon entropy function $L(g,d)$:

$$\begin{aligned} L(g,d) &= \sum_{i=1}^{k} \int_{x_i} p_d(x_i) \log(d(x_i)) dx_i + \sum_{i=1}^{k} \int_{z_i} p_{z_i} \log(1 - d(g(z_i))) dz_i \\ &= \sum_{i=1}^{k} \int_{x} p_d(x_i) \log(d(x_i)) + p_g(x_i) \log(1 - d(x_i)) dx_i \end{aligned} \tag{2.65}$$

Where z_i, are generative samples (or also called fake samples) come from generator function $g(\cdot)$ in which $p_{z_i} = p_g(x_i)$ (distribution of z_i equal to generative distribution of x_i) and $x_i = g(z_i)$.

Thus, if $p_d(x_i) = a$, $p_g(x_i) = b$ and $d(x_i) = y$ the $L(g,d)$ function become:

$$L(b,a) = \int_y (a \log(y) + b \log(1-y)) dy \tag{2.66}$$

$$\frac{\partial}{\partial y} L(b,a) = \frac{\partial}{\partial y} \int_y (a \log(y) + b \log(1-y)) dy = 0 \tag{2.67}$$

$$\frac{\partial}{\partial y} L(b,a) = y(a+b) - a = 0 \tag{2.68}$$

$$y = \frac{a}{a+b} \tag{2.69}$$

The $L(g,d)$ reaches a maximum of $\frac{a}{a+b}$ for any $(a,b) \in R^2$. For this reason the discriminator is not define outside $Supp(p_d) \cup Supp(p_g)$.

However, during the loss GAN minimisation the global minimum is achieved in Nash Equilib-

rium (NE) [40] $p_g = p_d$ with a value of $-log(4)$. Where if the equation 2.66 is substitute to y with maximum value $y = \frac{a}{a+b}$ is obtained:

$$max(L(g,d)) = \int_y (a\log(\frac{a}{a+b}) + b\log(1-(\frac{a}{a+b})))dy \tag{2.70}$$

So if the global minima is reached then $p_g(x_i) = p_d(x_i)$; also expressed as $a = b$. Then, when $a = b$ is replace in the above equation:

$$\begin{aligned} max(L(b,a)) = \int_y (\log(\frac{1}{2}) + \log(1-(\frac{1}{2})))dy = \\ \int_y (\log(\frac{1}{2}) + \log(\frac{1}{2}))dy = -\log(4) \end{aligned} \tag{2.71}$$

In the original GAN work [41] is proves the possibility to estimate it by the minimization of the symmetric Jensen-Shannon (JS):

$$max(L(p_g,p_d)) = \frac{1}{2}D_{KL}\left(p_d \left\| \frac{1}{2}(p_d+p_g)\right.\right) + \frac{1}{2}D_{KL}\left(p_d \left\| \frac{1}{2}(p_d+p_g)\right.\right) - log(4) \tag{2.72}$$

where D_{KL} represent the Kullback-Leibler divergence that is defined as:

$$D_{KL}(p_d \,\|p_g) = \int_x p_d \log \frac{p_d}{p_g} dx \tag{2.73}$$

The main role of the discriminator $d(\cdot)$ is to minimize this JS divergence between two different distributions p_d (real data distribution) and p_g (fake data distribution).

2.9 Metrics

This section gives a global view of the metrics used in the following Chapters. In particular, to evaluate segmentation, Dice Index (DI) and Hausdorff Distance (HD) are more commonly

used

Given the two binary masks A and B, the DI is defined as the following ratio:

$$DI(A,B) = \frac{2|A \cdot B|}{|A| + |B|} \tag{2.74}$$

The HD is the greatest of all the distances from a point in one set to the closest point in the other set.

$$HD(A,B) = min_{a \in A} min_{b \in B} d(a,b) \tag{2.75}$$

It is defined as the $max(d_a, d_b)$, where d_a is the distance from the automatic contour points to the closest point of the manual contour, and d_b is the opposite. HD is measured in *mm* in this work.

While in regression problems, where a series of real values must be compared, Mean Squared Error (MSE) is more used:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 \tag{2.76}$$

where n are the predictions generated from the data samples with Y the ground truth values vector and $\hat{Y}$ the predicted one.

3 Real-Time Abdominal Fetal Aorta Analysis with Ultrasound

The automatic analysis of ultrasound sequences can substantially improve the efficiency of clinical diagnosis. In this Chapter we present our attempt to automate the challenging task of measuring the vascular diameter of the fetal abdominal aorta from ultrasound images.

We propose a neural network architecture consisting of three blocks: a convolutional layer for the extraction of imaging features, a Convolution Gated Recurrent Unit (C-GRU) for enforcing the temporal coherence across video frames and exploiting the temporal redundancy of a signal, and a regularized loss function, called *CyclicLoss*, to impose our prior knowledge about the periodicity of the observed signal. However, an extension of this architecture will be presented in the following Chapters for performing segmentation tasks of left-ventricle, right-ventricle and atrium.

Here we present experimental evidence suggesting that the proposed architecture can reach an accuracy substantially superior to previously proposed methods, providing an average reduction of the mean squared error from $0.31mm^2$ (state-of-art) to $0.09mm^2$, and a relative error reduction from 8.1% to 5.3%. The mean execution speed of the proposed approach of 289 frames per second makes it suitable for real time clinical use.

3.1 Introduction

Fetal ultrasound (US) imaging plays a fundamental role in the monitoring of fetal growth during pregnancy and in the measurement of the fetus well-being. Growth monitoring is becoming increasingly important since there is an epidemiological evidence that abnormal birth weight is associated with an increased predisposition to diseases related to cardiovascular risk (such as diabetes, obesity, hypertension) in young and adults [42].

Among the possible biomarkers of adverse cardiovascular remodelling in fetuses and newborns, the most promising ones are the Intima-Media Thickness (IMT) and the stiffness of the abdominal aorta by means of ultrasound examination. Obtaining reliable measurements is critically based on the accurate estimation of the diameter of the aorta over time. However, the poor signal to noise ratio of US data and the fetal movement makes the acquisition of a clear and stable US video challenging. Moreover, the measurements rely either on visual assessment at bed-side during patient examination, or on tedious, error-prone and operator-dependent review of the data and manual tracing at later time. Very few attempts towards automated assessment have been presented [43, 44], all of which have computational requirements that prevent them to be used in real-time. As such, they have reduced appeal for the clinical use. In this Chapter we describe a method for automated measurement of the abdominal aortic diameter directly from fetal US videos. We propose a neural network architecture that is able to process US videos in real-time and leverage both the temporal redundancy of US videos and the quasi-periodicity of the aorta diameter.

The main contributions of the proposed method are as follows. First we show that a shallow CNN is able to learn imaging features and outperforms classical methods as level-set for fetal abdominal aorta diameter prediction. Second we add to the CNN a Convolution Gated Recurrent Unit (C-GRU) [45] for exploiting the temporal redundancy of the features extracted by CNN from the US video sequence. Finally, we add a new penalty term to the loss function used to train the CNN to exploit periodic variations.

3.2 Related work

The interest for measuring the diameter and intima-media thickness (IMT) of major vessels has stemmed from its importance as biomarker of hypertension damage and atherosclerosis in adults. Typically, the IMT is assessed on the carotid artery by identifying its lumen and the different layers of its wall on high resolution US images. The improvements provided by the design of semi-automatic and automatic methods based mainly on the image intensity profile, distribution and gradients analysis, and more recently on active contours. For a comprehensive review of these classical methods we refer the reader to [46] and [47]. In the prenatal setting, the lower image quality, due to the need of imaging deeper in the mother's womb and by the movement of the fetus, makes the measurement of the IMT biomarker, although measured on the abdominal aorta, challenging.

Methods that proved successful for adult carotid image analysis do not perform well on such data, for which only a handful of methods (semi-automatic or automatic) have been proposed, making use of classical tracing methods and mixture of Gaussian modelling of blood-lumen and media-adventitia interfaces [43], or on level sets segmentation with additional regularizing terms linked to the specific task [44]. However, their sensitivity to the image quality and lengthy computation prevented an easy use in the clinical routine.

Deep learning approaches have outperformed classical methods in many medical tasks [48]. The first attempt in using a CNN, for the measurement of carotid IMT has been made only recently [49]. In this work, two separate CNNs are used to localize a region of interest and then segment it to obtain the lumen-intima and media-adventitia regions. Further classical post-processing steps are then used to extract the boundaries from the CNN based segmentation. The method assumes the presence of strong and stable gradients across the vessel walls, and extract from the US sequence only the frames related to the same cardiac phase, obtained by a concomitant ECG signal.

However, the exploitation of temporal redundancy on US sequences was shown to be a solution for improving overall detection results of the fetal heart [50], where the use of a CNN coupled with a recurrent neural network (RNN) is strategic. Other works, propose similar approach in

order to detect the presence of standard planes from prenatal US data using CNN with Long-Short Term Memory (LSTM) [51].

3.3 Datasets

This study makes use of a dataset consisting of 25 ultrasound video sequences acquired during routine third-trimester pregnancy check-up from the Department of Woman and Child Health of the University Hospital of Padova (Italy). The local ethics committee approved the study and all patients gave written informed consent. The gestational age for the scans we used is 32 weeks and 4 $days \pm 4$ $weeks$ $(mean \pm stdev)$.

Fetal US data were acquired using an US machine (Voluson E8, GE) equipped with a 5 MHz linear array transducer, according to the guidelines in [52, 53], using a 70^o FOV, image dimension 720x960 pixels, a variable resolution between 0.03 and 0.1 *mm* and a mean frame rate of 47 fps. Gain settings were tuned to enhance the visual quality and contrast during the examination. The length of the video is between 2s and 15s, ensuring that at least one full cardiac cycle is imaged.

After the examination, the video of each patient was reviewed and a relevant video segment was selected for semi-automatic annotation considering its visual quality and length: all frames of the segment were processed with the algorithm described in [43] and then the diameters of all frames in the segments were manually reviewed and corrected. The length of the selected segments varied between 21 frames 0.5s and 126 frames 2.5s. The 25 annotated segments in the dataset were then randomly divided into training (60% of the segments), validation (20%) and testing (20%) sets. In order to keep the computational and memory requirements low, each frame was cropped to have a square aspect ratio and then resized to 128×128 pixels. We also make this dataset public to allow for the results to be reproduced.

3.4 Network architecture

Our output is the predicted value $\hat{y}[t]$ of the diameter of the abdominal aorta at each time point. Our proposed deep learning solution consists of three main components (see Figure 1): a Convolutional Neural Network (CNN) that captures the salient characteristics from ultrasound input images; a Convolution Gated Recurrent Unit (C-GRU) [45] exploits the temporal coherence through the sequence; and a regularized loss function, called *CyclicLoss*, that exploits the redundancy between adjacent cardiac cycles.

Our input consists of a set of sequences whereby each sequence $S = [s[1], ..., s[K]]$ has dimension $N \times M$ pixels at time t, with $t \in \{1, \ldots, K\}$. At each time point t, the CNN extracts the feature maps $x[t]$ of dimensions $D \times N_x \times M_x$, where D is the number of maps, and N_x and M_x are their in-plane pixel dimensions, that depend on the extent of dimensionality reduction obtained by the CNN through its pooling operators.

The feature maps are then processed by a C-GRU layer [45]. The C-GRU combines the current feature maps $x[t]$ with an encoded representation $h[t-1]$ of the feature maps $\{x[1], \ldots, x[t-1]\}$ extracted at previous time points of the sequence to obtain an updated encoded representation $h[t]$, the *current state*, at time t: this allows the exploitation of the temporal coherence in the data. The $h[t]$ of the C-GRU layer is obtained by two specific gates designed to control the information inside the unit: a reset gate, $r[t]$, and an update gate, $z[t]$, defined as follows:

$$r[t] = \sigma(W_{hr} * h[t-1] + W_{xr} * x[t] + b_r) \tag{3.1}$$

$$z[t] = \sigma(W_{hz} * h[t-1] + W_{xz} * x[t] + b_z) \tag{3.2}$$

Where, $\sigma()$ is the sigmoid function, $W_{\cdot}$ are recurrent weights matrices whose first subscript letter refers to the input of the convolution operator (either the feature maps $x[t]$ or the state $h[t-1]$), and whose second subscript letter refers to the gate (reset r or update z). All these matrices, have a dimension of $D \times 3 \times 3$ and $b_{\cdot}$ is a bias vector. In this notation, $*$ defines the

convolution operation. The current state is then obtained as:

$$h[t] = (1 - z[t]) \odot h[t-1] + z[t] \odot \tanh(W_h * (r[t] \odot h_{t-1}) + W_x * x[t] + b). \tag{3.3}$$

Where $\odot$ denotes the dot product and W_h and W_x are recurrent weight matrices for $h[t-1]$ and $x[t]$, used to balance the new information represented by the feature maps $x[t]$ derived by the current input data $s[t]$ with the information obtained observing previous data $s[1], \ldots, s[t-1]$. On the one hand, $h[t]$ is then passed on for updating the state $h[t+1]$ at the next time point, and on the other is flattened and fed into the last part of the network, built by Fully Connected (FC) layers progressively reducing the input vector to a scalar output that represent the current diameter estimate $\hat{y}[t]$.

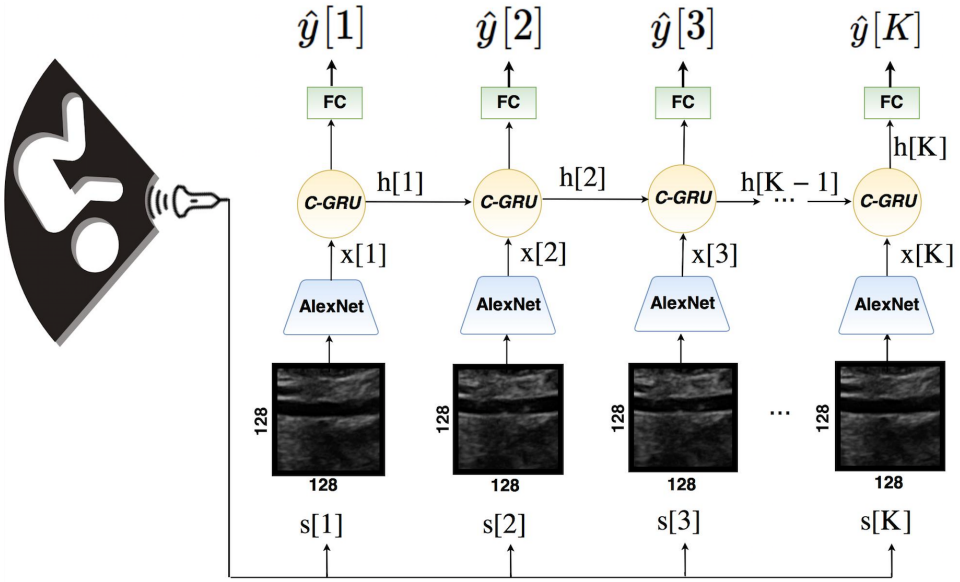

Figure 3.1 The deep-learning architecture proposed for abdominal diameter aorta prediction. The blue blocks represent the features extraction through a CNN (AlexNet) which takes in input a US sequence S, and provides for each frame $s[t]$ a features map $x[t]$ that is passed to Convolution Gated Recurrent Units (C-GRU) (yellow circle) that encodes and combines the information from different time points to exploit the temporal coherence. The fully connected block (FC, in green), takes as input the current encoded state $h[t]$ as features to estimate the aorta diameter $\hat{y}[t]$.

3.4.1 CyclicLoss

Under the assumption that the pulsatility of the aorta follows a periodic pattern with the cardiac cycle, the diameter of the vessel at corresponding instants of the cardiac cycle should ideally be equal. Assuming a known cardiac period T_{period}, we propose to add a regularization term to the loss function used to train the network as to penalize large differences of the diameter values that are estimated at time points that are one cardiac period apart.

We call this regularization term *CyclicLoss* (*CL*), computed as L_2 norm between pairs of predictions at the same point of the heart cycle and from adjacent cycles:

$$CL = \sqrt{\sum_{n=1}^{N_{cycles}} \sum_{t=0}^{T_{period}} \| \hat{y}[t+(n-1)T_{period}] - \hat{y}[t+nT_{period}] \|_2} \tag{3.4}$$

The T_{period} is the period of the cardiac cycle, while N_{cycles} is the number of integer cycles present in the sequence and $\hat{y}[t]$ is the estimated diameter at time t. Notably, the T_{period} is determined through a peak detection algorithm on $y[t]$, and the average of all peak-to-peak detection distances define its value. While the N_{cycles} is the number of cycles present, calculated as the total length of the $y[t]$ signal divided by T_{period}.

The loss to be minimized is therefore a combination of the classical MSE (eq 2.76) with the *CL*, and the balance between the two is controlled by a constant λ:

$$Loss = MSE + \lambda \cdot CL = \frac{1}{K} \sum_{t=1}^{K} (y[t] - \hat{y}[t])^2 + \lambda \cdot CL \tag{3.5}$$

where $y[t]$ is the target diameter at time point t. It is worth noting that the knowledge of the period of the cardiac cycle is needed only during training phase. Whereas, during the test phase, on an unknown image sequence, the trained network provides its estimate blind of the periodicity of the specific sequence under analysis.

3.4.2 Implementation details

For our experiments, we chose AlexNet [10] as a feature extractor for its simplicity. It has five hidden layers with 11×11 kernel size in the first layer, 5×5 in the second and 3×3 in the last three layers; it is well suited to the low image contrast and diffuse edges characteristic of US sequences. Each network input for the training is a sequence of $K = 125$ ultrasound frames with $N = M = 128$ pixels, AlexNet provides feature maps of dimension $D \times N \times M = 256 \times 13 \times 13$, and the final output $\hat{y}[t]$ is the estimated abdominal aorta diameter value at each frame.

The loss function is optimised with the Adam algorithm [54] that is a first-order gradient-based technique. The learning rate used is $1e^{-4}$ with 2125 iterations (calculated as number of patients $\times$ number of ultrasound sequences) for 100 epochs. In order to improve generalization, data augmentation of the input with a vertical and horizontal random flip is used at each iteration. The λ constant used during training with *CyclicLoss* takes the value of $1e^{-6}$.

3.5 Experiments

3.5.1 Methods

The proposed architecture is compared with the currently adopted approach in section 4. This method provides fully-automated measurements in lumen identification on prenatal US images of the abdominal aorta [44] based on edge-based level set. In order to understand the behaviour of different features extraction methods, we have also explored the performance of new deeper network architectures whereby AlexNet was replaced it by InceptionV4 [9] and DenseNets 121 [8].

3.5.2 Results

The performance of each method was evaluated both with respect to the MSE and to the mean absolute relative error (RE); all values are reported in Tab.3.1 in terms of average and standard deviation across the test set.

Methods	**MSE [mm^2]**	**RE [%]**	**p-value**
AlexNet	0.29(0.09)	8.67(10)	1.01e-12
AlexNet+C-GRU	0.093(0.191)	6.11(5.22)	1.21e-05
AlexNet+C-GRU+CL	**0.085(0.17)**	**5.23(4.91)**	"-"
DenseNet121	0.31(0.56)	9.55(8.52)	6.00e-13
DenseNet121+C-GRU	0.13(0.21)	7.72(5.46)	7.78e-12
InceptionV4	6.81(14)	50.4(39.5)	6.81e-12
InceptionV4+C-GRU	0.76(1.08)	16.3(9.83)	2.89e-48
Level-set	0.31(0.80)	8.13(9.39)	1.9e-04

Table 3.1 The table shows the mean (standard deviation) of MSE and RE error for all the comparison models. The combination of C-GRU and the *CyclicLoss* with AlexNet yields the best performance. Adding recurrent units to any CNN architecture improves its performance; however deeper networks such as InceptionV4 and DenseNets do not show any particular benefits with respect to the simpler AlexNet. Notably, we also consider the p-value for multiple models compared with the propose network AlexNet+C-GRU+CL, in this case the significance level should be 0.05/7 using the Bonferroni correction.

In order to provide a visual assessment of the performance, representative estimations on four sequences of the test set are shown in Fig.5.1. The naive architecture relying on a standard loss and its C-GRU version are incapable to capture the periodicity of the diameter estimation. The problem is mitigated by adding the *CyclicLoss* regularization on MSE. This is quantitatively shown in Tab.3.1, where the use of this loss further decreases the MSE from $0.093mm^2$ to $0.085mm^2$, and the relative error of from 6.11% to 5.23%.

Strikingly, we observed that deeper networks are not able to outperform AlexNet on this dataset. Their limitation may be due to over-fitting. Nevertheless, the use of C-GRU greatly improve the performance of both networks both in terms of MSE and of RE. Further, we also performed a non-parametric test (Kolmogorov-Smirnov test) to check if the best model was statistically different compared to the others.

The results obtained with the complete model AlexNet+C-GRU+CL are indeed significantly different from all others ($p < 0.05$) also, when the significant level is adjusted for multiple comparison applying the Bonferroni correction [55, 56].

3.6 Discussion and conclusion

The deep learning (DL) architecture proposed shows excellent performance compared to traditional image analysis methods, both in accuracy and efficiency. This improvement is achieved through a combination of a shallow CNN and the exploitation of the temporal and cyclic coherence. Our results seem to indicate that a shallow CNNs perform better than deeper CNNs such as DenseNet 121 and InceptionV4; this might be due to the small dimension of the data set, a common issue in the medical settings when requiring manual annotations of the data.

3.6.1 The *CyclicLoss* benefits

The exploitation of temporal coherence is what pushes the performance of the DL solution beyond current image analysis methods, reducing the MSE from $0.29mm^2$ (naive architecture) to $0.09mm^2$ with the addition of the C-GRU. The *CyclicLoss* is an efficient way to guide the training of the DL solution in case of data showing some periodicity, as in cardiovascular imaging. Please note that the knowledge of the signal period is only required by the network during training, and as such it does not bring additional requirements on the input data for real clinical application. We argue that the *CyclicLoss* is making the network learn to expect a periodic input and provide some periodicity in the output sequence.

3.6.2 Limitations and future works

A drawback of this work is that it assumes the presence of the vessel in the current field of view. Further research is thus required to evaluate how well the solution adapts to the scenario of lack of cyclic consistency, when the vessel of interest can move in and out of the field of view during the acquisition, and to investigate the possibility of a concurrent estimation of the cardiac cycle and vessel diameter. Finally, the C-GRU used in our architecture, has two particular advantages compared to previous approaches [50, 51]: first, it is not subject to the vanishing gradient problem like the RNN, allowing the training from long sequences of data. Second, it has less computational cost compared to the LSTM, and that makes it suitable for

real time video application.

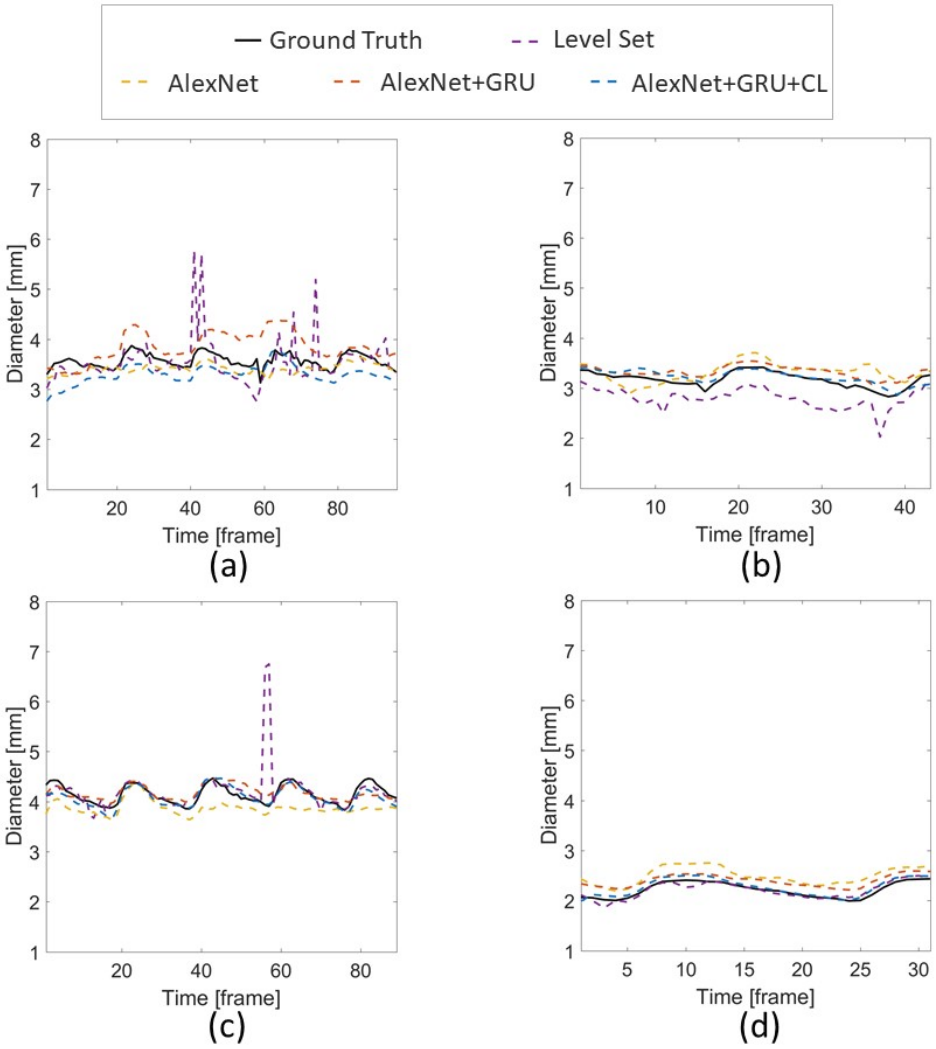

Figure 3.2 Each panel (a-c) shows the estimation of the aortic diameter at each frame of fetal ultrasound videos in the test set, using the level set method (dashed purple line), the naive architecture using AlexNet (dashed orange line), the AlexNet+C-GRU (dashed red line), and AlexNet+C-GRU trained with the *CyclicLoss* (dashed blue line). The ground truth (solid black line) is reported for comparison. Panels (a,c) show the results on long sequences where more than 3 cardiac cycles are imaged, whereas panels (b,d) show the results on short sequences where only 1 or two cycles are available.

4 Automated segmentation on the entire cardiac cycle

The segmentation of the left ventricle (LV) from CINE MRI images is essential to infer important clinical parameters. Typically, machine learning algorithms for automated LV segmentation use annotated contours from only two cardiac phases, diastole, and systole.

In this Chapter, we present an analysis work-flow for fully-automated LV segmentation that learns from images acquired through the cardiac cycle. The workflow consists of three components: first, for each image in the sequence, we perform an automated localization and subsequent cropping of the bounding box containing the cardiac silhouette; Second, we identify the LV contours using a Temporal Fully Convolutional Neural Network (T-FCNN), which extends the Recurrent Convolutional Neural Networks seen in the previous Chapter.

Finally, we further defined the boundaries using either one of two components: fully-connected Conditional Random Fields (CRFs) with Gaussian edge potentials and Semantic Flow. Our initial experiments suggest that significant improvement in performance can potentially be achieved by using a recurrent neural network component that explicitly learns cardiac motion patterns whilst performing LV segmentation.

4.1 Introduction

The LV segmentation is often carried out using Short-Axis (SA) section. The SA is a plane perpendicular to the Long-Axis (LA) of the heart that aligns the apex of the left ventricle

with its base [57]. The segmentation of the heart requires the identification of two anatomical regions: the inner part called the endocardium, and the outer part, the epicardium. Identifying and segmenting those two regions in CMR images presents different levels of difficulty: while the endocardium has sufficient contrast, the epicardial surface presents profiles of intensity with little contrast [58]. Notwithstanding, a clear delineation of the epicardium contours is a challenging task due to non-homogeneity in the blood flow. Moreover, the presence of strong unevenness on the wall in the interior of the heart chambers due to the sporadic presence of papillary muscles does not allow a clear delimitation of the endocardial wall [59]. The level of difficulty also depends on the particular ventricular section; the apical and basal sections are much more difficult to segment because the resolution of the image is lower and does not allow the detection of ventricular structure [60].

To speed up segmentation, in the last decade, fully automatic segmentation algorithms have been used. An automatic segmentation approach would typically consist of two components: an object detector, which attempts to locate the Region Of Interest (ROI), i.e. the region that is most likely to contain the heart, and then a segmentation algorithm that extracts the cardiac silhouette. Existing approaches for fully-automated segmentation of the LV can be divided into three groups: image-based methods, deformable models and pixel-based classification methods. Image-based methods consist of finding the endocardium border using a simple threshold [61] or Dynamic Programming (DP) [62]. Under the umbrella of deformable models, Level-Set (LS) segmentation approaches such as Chan-Vase (CV) [63] have been commonly used.

Pixel-based classification is one of the most commonly used approaches to cardiac segmentation. Current state-of-the-art methods are based on supervised Deep Neural Networks [64, 65, 66, 67, 68], which are used for both object detection and segmentation. Recently, proposed architectures include Fully Convolution Neural Network (FCNN) [67], Stacked Autoencoders (SA) [66] and Deep Belief Networks [64, 65]. Occasionally, a post-processing method is also applied, following the initial segmentation, to obtain more realistic contours. In some cases, a single architecture trained end-to-end has been shown to achieve satisfactory performance without the need for further pre- and post-processing, e.g. by modeling the spatial dependence amongst SA sections [68]. There are also architectures for segmentation of

three-dimensional images, such as [69] which have not been used for cardiac MRI.

To the best of our knowledge, none of the existing neural network architectures for cardiac segmentation takes into explicit consideration the movement of the LV. As the heart is in a constant motion, taking into account the temporal dimension and learning motion patterns across the entire cardiac cycle is expected to produce improved and temporally coherent segmentations.

Prior to deep learning, a number of segmentation methodologies have been proposed in the literature to leverage this dynamic component such as Deformable Surface with Time-dependent Constraints [70, 71], 4D Markov Random Fields [72], and Principal Component Analysis (PCA) of the myocardium movement field [73]. Other approach combine 3D deformable surfaces with a statistical model of four-dimensional heart movement [74]. The optical flow equations have also been combined with the level-set to enforce visual constancy [75].

The Automated Cardiac Diagnosis Challenge (ACDC) challenge recorded a state-of-the-art of deep-learning architecture for RV, LV and myocardium segmentation. Where, different U-Net style architectures, in several hyper-parameters configuration for 2D or 3D convolutions with or without pre-training on downsampling layer [76, 77, 78] was tested. However, the ACDC challenge winner uses an ensemble network between 2D and 3D FCNN built without pooling operation (i.e due to the large slice gap)[79], where final prediction of 2D U-Net are joined whit the 3D model.

Innovative networks with 2D convolution kernels was also presented, as M-Net architecture [80], in which the features of each up-sampling layer was concatenated with a previous allowing greater generalization of information from other layers. While a FCNN with dense convolutions and a pre-inception layer was trained after the pre-localisation of LV and RV trough Fourier transforms followed by a Canny edge detector [81]. Dense convolution has different advantages: they strengthen feature propagation, encourage the functionality to reduce and reuse the number of parameters, alleviate the vanishing gradient problem. Eventually, Zotti et al. presented a 2D U-Net with convolution layer long-drawn the skip path connections called Grid Net [82]; this strategy did not show decisive results.

In this Chapter, we exploit a complementary source of information, the coherence across consecutive frames, and propose a Temporal Convolution Neural Network (T-FCNN) architecture with Semantic Flow post-processing. The performance of the proposed architecture is compared against an established FCNN model, which treats each cardiac phase independently and achieves good segmentation performance [67]. An alternative post-processing choice, the use of Conditional Random Fields (CRFs), is also investigated.

4.2 TWINS-UK dataset

The TWINS-UK is a voluntary registry that includes >12,000 twins [83]. For this study, 68 consecutive female subjects (mean age 62 ± 9 years) were recruited from the TWINS-UK cohort.

The CMR scans were performed on a 1.5-T clinical scanner (Achieva, Philips Healthcare, Best, The Netherlands). Each dataset included 12 to 14 equidistant and contiguous short-axis CINE from the atrioventricular (AV) ring to the apex, completely covering both ventricles (slice thickness 8 mm; no gap mm; field of view was 360×480 mm and matrix size 156×144). The ECG-gated steady-state free-precession (SSFP) end-expiratory breath-hold 2D CINES were acquired. Images were acquired with 30 phases/cardiac cycle corresponding to a temporal resolution of 25-35 milliseconds at a heart rate of 60-80 beats per minute.

The dataset was randomly divided into training, validation and testing sets of sizes 70%, 15% and 15%, respectively. Each pair of twins was allocated to a specific subset and not separated to avoid any genetic similarities affecting our results.

4.3 Proposed analysis work-flow

The work-flow is divided into three stages: LV position detection, LV segmentation, and LV contour refinement. Our input $x_{t,i}$ consists of the entire temporal sequence obtained in all cardiac phases, while the label output $y_{t,i}$ is the corresponding sequence of binary masks, one per time point. Whereas, the i index indicates the specific label pixel at the temporal image

phase t. Each input image was downsized to 236×236. The output masks have the same size after a padding operation of 44×44 pixels.

4.3.1 Single-frame LV position detection

For the automated detection of the LV in each time frame, we used the Over-Feat algorithm [84]. In order to train the object detection layers in Over-Feat, we pre-trained the GoogLeNet architecture [9] within the ImageNet database and later carried out fine-tuning using our LV images in a sliding window fashion. The prediction of the bounding box coordinates is obtained through regression layers minimizing an L2 loss.

4.3.2 Sequence-based LV segmentation

Upon detecting the bounding box containing the heart, the cardiac contours are inferred with an architecture that extends U-Net [30], originally proposed for the segmentation of biomedical images. More specifically, our solution is an improvement of a Fully Convolutional Neural Network (FCNN) [31], which becomes our baseline for comparisons.

A standard U-Net takes an individual frame as input and estimates the corresponding binary mask as output. An encoding (descending) path is composed of a sequence of hidden layers that are used to learn a representation of the input image. This is then followed by another sequence of hidden layers forming a decoding (ascending) path through which all the feature maps are gradually restored to the original image size and are used to infer the final binary masks.

The hidden layers in the encoding path consist of 2D convolutional filters followed by rectified linear units (ReLU) and max-pooling layers whereas the decoding path consists of deconvolutional (or up-sampling) filters also followed by ReLU mappings. One of the key features of this architecture is the use of skip paths connections between convolution and deconvolution layers for the purpose of fusing semantic and local information. Every skip path concatenates feature maps from the encoding to the decoding path. For the purpose of segmenting the entire cardiac

motion, we modify the U-Net architecture by adding a recurrent layer immediately after the descending path.

The purpose of this layer is to leverage the information flowing from preceding image frames and enforce temporal coherence. Then, the feature maps learned at the encoding stage are used as inputs for a recurrent element coded with a Convolutional Gated Recurrent Unit (Conv-GRU) [85]. The resulting architecture, a Temporal Fully-Convolutional Neural Network (T-FCNN), is illustrated in Fig. 4.1. The encoding path consists of four blocks of hidden layers, which are arranged as follows: two 3×3 convolutional layers (with stride set to 1), a ReLU layer, a Batch Normalization (BN) layer, that scaling and adjusting the features activations and, finally, a 2×2 max pooling layer (with stride set to 2).

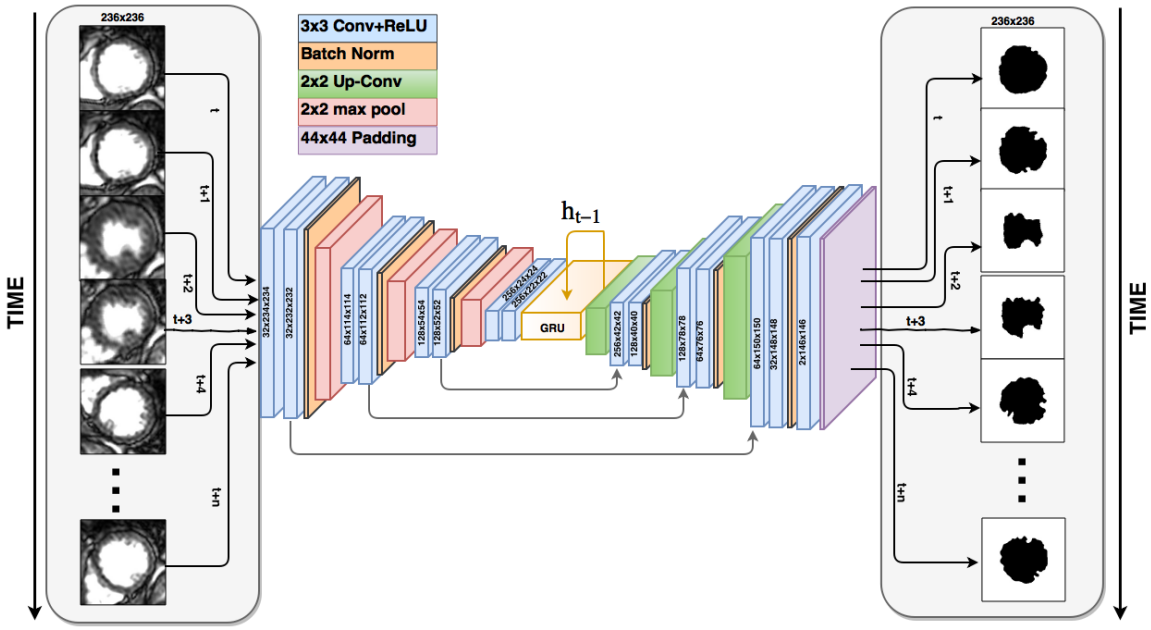

Figure 4.1 T-FCNN architecture. Blue blocks represent a convolutional layer followed by ReLU operations; orange blocks represent batch normalization operations; green blocks correspond to up-convolution operations; pink blocks max-pooling operations and purple blocks identify padding operations. Black arrows represent the input and output for each CINE MRI frame and its temporal segmentation. Gray arrows denote copy operations and the yellow arrow indicates the recurrent connection implemented through a Conv-GRU layer.

4.3.3 Post-processing

The final component of the segmentation work-flow is a post-processing algorithm that has the potential to further improve upon the binary masks predicted by the T-FCNN. In this study, we

compare two methods that have been particularly successful for semantic segmentation tasks, i.e. Fully Connected CRFs with Gaussian edge potentials [11] and Semantic-Flow [86]. The fully connected CRFs minimise an energy function:

$$\mathrm{E}(x) = \sum_{t=0}^{T-1} (\sum_{i} \theta_i(y_{t,i}) + \sum_{ij} \theta_{ij}(y_{t,i}, y_{t,j})) \tag{4.1}$$

Where $y_{t,i}$ is the segmentation mask from frame t, i is the index of each label pixel, θ_i is the unary potential, defined as $\theta_i(y_{t,i}) = -\log(\mathrm{P}(y_{t,i}))$, and θ_{ij} is the pairwise potential, defined by:

$$\theta_{ij}(y_{t,i}, y_{t,j}) = \mu(y_{t,i}, y_{t,j}) k(f_{t,i}, f_{t,j}). \tag{4.2}$$

The function $\mu(y_{t,i}, y_{t,j})$ equals one when $y_{t,i} \neq y_{t,j}$, zero otherwise, and the function $k(f_{t,i}, f_{t,j})$ is a Gaussian kernel, evaluated using features $f_{t,i}$ and $f_{t,j}$ corresponding to pixels i and j respectively. Notably, the first kernel is driven by both pixel position (p) and color intensities (I); where the second kernel only uses pixel position. Kernel coefficients were kept fixed, and the energy function was minimized using the L-BFGS algorithm for nonlinear optimization using multi-thread CPUs.

The second post-processing algorithm uses a Semantic-Flow (SF) approach, which is another alternative to exploit the temporal coherence in a sequence. Our implementation follows the formulation presented in [86], and the reader is referred to that paper for a detailed explanation of the concepts that are summarised next. Given a CINE MRI sequence of 2D frames $x_{t,k}$, our aim is to estimate simultaneously the flow vector field $(u_{t,k}, v_{t,k})$ that maps every pixel between consecutive 2D images (Fig. 4.2). The task is divided in two regions defined by $g_{t,k}$ for $k \in \{1,2\}$, corresponding to LV mask and background, and vector fields are parametrised with a set of parameters $\theta_{t,k}$. Input is $y_{t,k}$, the initial LV segmentation comes from T-FCNN. We then wish to minimize the following energy function $E_{LV_{Seg}}(u, v, g, \theta, x, y)$, that consists of five terms: data, motion, time, space and coupling term.

$$E_{LV_{Seg}}(u,v,g,\theta,x,y) = \sum_{k=0}^{2}\left\{\sum_{t=0}^{T-1}\left\{E_d(u_{t,k},v_{t,k},g_{t,k},x_{t,k},x_{t+1,k})\right.\right.$$
$$\left.\left.+\lambda_m E_m(u_{t,k},v_{t,k},g_{t,k},\theta)+\lambda_t E_t(u_{t,k},v_{t,k},g_{t,k},g_{t+1,k})\right\}+\sum_{t=0}^{T}\left\{\lambda_c E_c(g_{t,k},y_{t,k})+\lambda_s E_s(g)\right\}\right\} \tag{4.3}$$

The data term E_d measures the similarity between the gray scale intensities of adjacent frames, the motion term E_m enforces some regularity of the vector field in pixels that belong to the same region or that are close to each other, the time term E_t constrains the regularity of pixels belonging to a LV region or background along the sequence, E_s enforces the connectivity of pixels within the same region, and the coupling term E_c emphasizes the affinity between background segmentation and LV segmentation. The different λ are constants that weight the contribution of the energy terms, and that are left constant in our study.

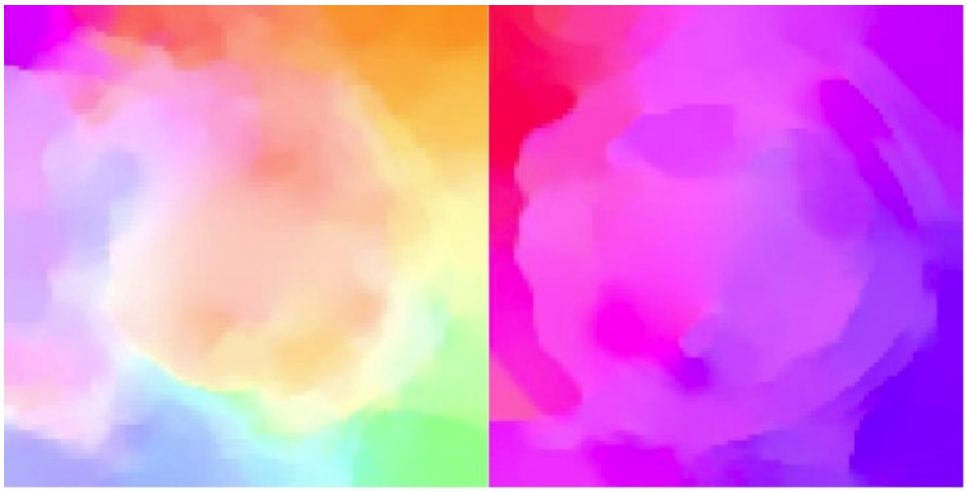

Figure 4.2 Temporal LV flow field after the Optical Flow application between two CINE-MRI frames. As we can see the Optical flow highlight the contour of the LV.

4.4 Experimental Results

The performance of the detection of the position of the LV was assessed with the Intersection Over Union, reaching a score of 98%. The accuracy of the final segmentation was measured using three different metrics: the Dice Index (DI), the Average Perpendicular Distance (APD)

and the Conformity (C) index [66].

A set of architectures from the baseline FCNN to the proposed T-FCNN with CRFs was implemented and analyzed. Parameters of both T-FCNN and FCNN were tuned within Ada-delta optimizer, and both architectures are trained end-to-end with spatial cross entropy criterion with a learning rate of $1e-4$.

Results are reported in [Table 4.1]. T-FCNN improves FCNN, reducing an error of 30.5% for the APD metric (from 10.3 to 7.1 mm), and the addition of CRFs further reduces another 12% this error metric (from 7.1 to 6.3 mm). The CRF did improve the performance in all cases, but SF only improved the metric of APD when added to the FCNN. Some illustrative examples of the final segmentation result are provided in [Fig. 4.3].

Algorithms	DICE (%)	APD (mm)	C(%)
FCNN	0.9745(0.0163)	10.2734(12.7622)	0.9472(0.0352)
T-FCNN	0.9803(0.0263)	7.1427(11.0284)	0.9583(0.0592)
FCNN+CRFs	0.9774(0.0161)	8.2084 (8.3545)	0.9532(0.0345)
T-FCNN+CRFs	**0.9815(0.0245)**	**6.2903(8.3814)**	**0.9610 (0.0551)**
FCNN+SF	0.9735(0.0506)	9.3289(13.4287)	0.8872(1.2804)
T-FCNN+SF	0.9717(0.0743)	8.1635(12.8057)	0.8213(1.7954)
FCNN+CRFs+SF	0.9762(0.0462)	8.4325(11.1094)	0.8939(1.2843)
T-FCNN+CRFs+SF	0.9737(0.0735)	7.5983(11.5803)	0.8097(1.9614)

Table 4.1 Segmentation performance results obtained on the TwinsUK dataset using different combinations of segmentation (FCNN and T-FCNN) and post-processing (CRF and SF) algorithms.

4.5 Discussion and Conclusions

Exploitation of the temporal coherence across frames is a useful resource for the fully automated, robust and accurate segmentation of CINE MRI sequences, as required for clinical practice.

A CINE MRI study has a large level of coherence both in space and time. Driven by clinical interest (i.e. the generation of metrics such as the blood pool volume at a given time) and by the lower burden required for the generation of the ground truth (i.e. only need to segment 10-12 slices at a given time), most of the algorithms proposed to date exploit the coherence in space [87]. The use of a recurrent unit in space (i.e. across slices) did reduce the APD in a 4.2% or a

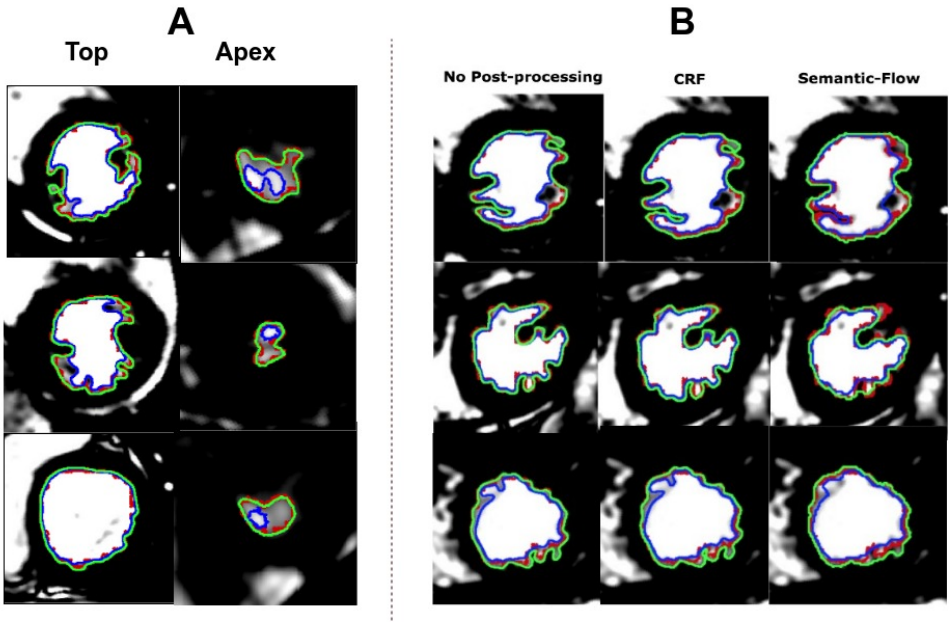

Figure 4.3 a) Comparison of the segmentations obtained from FCNN (blue line) vs T-FCNN (green line) compared within clinical ground truth (red line). The left column shows the top slices LV segmentation. While the right column shows the apex LV segmentation cases. Both, show that T-FCNN has good segmentation performance in comparison with FCNN. Especially FCNN, it tends to segment only high-intensity regions; as we can see in the apical cases. b) Comparison of the segmentation obtained with no post-processing methods vs post-processing. As we can see the Semantic-Flow (Semantic flow column) tends to blunt the LV prediction, while the CRF (CRF column) remains fairly consistent with T-FCNN (no post-processing column).

13% at the MICCAI and PRETERM cohorts as reported in [87], whereas the use of a recurrent unit in time (i.e. across frames) improved the APD of 30% more in the TWINS-UK cohort (see Table 4.1). Temporal coherence seems to be a more useful resource, probably due to the fact that there is indeed a larger correlation between adjacent frames in time than between adjacent slices in a CINE MRI study. And this has been achieved by a simple extension of the U-Net architecture through a recurrent neural network component (T-FCNN).

In the search of the best strategy to exploit temporal coherence, our results suggest that T-FCNN is a better solution than an FCNN+SF. We can't claim that we fully exploited the potential of both strategies (i.e. exploration of all the parametric space in SF is a tedious exercise), but our results with a reasonable set of parameter and architecture combinations consistently suggested that SF did at most only marginally improve the LV segmentation performance. It is important to note that the clinical protocol for the CMR segmentation in our TWINS-UK cohort included the papillary muscles as part of the myocardium, in contrast with the majority of previous

studies [1, 87]. This is a more challenging task for the human observer and the algorithm, since the smoothness constraints of contouring a circular shape cannot be used, and is the reason why previous APD values were smaller (i.e. around 2mm [68]).

Nevertheless, the use of a recurrent unit is not a perfect solution, since it is going to be limited by the problem of the vanishing gradients that reduces the performance of the backpropagation through time. This is going to limit the number of cardiac frames that can be inputted into the architecture. In practice, this should not be a major problem, since acquisitions with more than 30 frames, as used in this study, are not common in CMR. The translation of this concept to echocardiography will nevertheless bring some challenges, where you can deal with sequences of up to thousands of frames using ultra-fast acquisition protocols. An easy first solution will then be the use of an upper bound value to the gradient (i.e. gradient clipping) or adding a regularisation term able to increase or decrease the gradient magnitude [36]. Finally, the use of 3D convolution kernels could be useful for progressively decreasing the number of input sequences; avoiding excessively long temporal or spatial sequences.

The main practical bottleneck is the heavier burden of segmentation needed to transfer the learning to another study or experimental setting. Instead of learning from single frames, the proposed architecture will need to learn from complete sequences. The availability of semi-automatic segmentation solutions in commercial products should make this task affordable. In any case, our TWINS-UK dataset with its manual ground truth segmentation is made available to the community with this publication as a reference for future solutions to exploit the temporal coherence in CMR sequences.

5 V-FCNN: Volumetric Fully Convolution Neural Network For Automatic Atrial Segmentation

Atrial Fibrillation (AF) is a common electro-physiological cardiac disorder that causes changes in the anatomy of the atria. A better characterization of these changes is desirable for the definition of clinical biomarkers. There is thus a need for its fully automatic segmentation from clinical images. This work presents an architecture based on 3D-convolution kernels, a Volumetric Fully Convolution Neural Network (V-FCNN), able to segment the entire atrial anatomy in a one-shot from high-resolution images (640×640 pixels). A loss function based on the mixture of both MSE and Dice Loss (DL) is used, in an attempt to combine the ability to capture the bulk shape as well as the reduction of local errors caused by over-segmentation.

Results demonstrate a good performance in the middle region of the atria along with the challenges impact of capturing the pulmonary veins variability or valve plane identification that separates the atria to the ventricle. Despite the need to reduce the original image resolution to fit into Graphics Processing Unit (GPU) hardware constraints, 92.5% and 85.1% were obtained respectively in the 2D and 3D Dice metric in 54 test patients (4752 atria test slices in total), making the V-FCNN a reasonable model to be used in clinical practice.

5.1 Introduction

Atrial Fibrillation (AF) is a common electro-physiological cardiac disorder with a large world-wide prevalence [88] that causes changes in the anatomy of the atria. A better characterization of these changes, which are known to further promote and sustain fibrillation, is desirable for the definition of clinical biomarkers. These biomarkers can be directly started from the image (i.e. the shape of the atria [89], or the fibrotic burden by late gadolinium enhanced (LGE) magnetic resonance imaging (MRI) [90]) rather than mechanistic simulations of the function (i.e. computation of the risk of arrhythmia perpetuation [91]).

There is thus a need for a full atrium automatic segmentation from clinical images, especially in LGE studies. The current state of the art is based on tedious, along with error-prone manual procedures, and the main difficulty is the lack of atrial tissue contrast. Fully automated solutions are desirable to speed up the process and remove inter- and intra-observer variability. In this direction, a combination of multi-atlas registration within 3D level-set has been proposed, reporting a reasonable performance in the main atrial body and pulmonary vein regions [92]. The large computational burden of this multi-atlas approach can be alleviated by the use of Convolutional Neural Networks (CNNs), as has been illustrated in [93] for the analysis of 2D MRI slices.

The 3D atrial segmentation challenge highlighted new state of art algorithms whereas the majority of submitted networks have been an extension of 3D U-net with cropping or downsampling input where dice loss or Mean Square Error was used for training[19, 94, 95, 96]. Though, with the Dilated Convolutions (DC), able to explore complex scale portions of the output down-sampling the features with 3D convolution and residual connections in both down-sampling and up-sampling. Surely, the residual transports feature before each convolution block to those after it for a final concatenation [97].

A dual 3D U-Net approach has also been examined, wherever that first U-Net localizes the atrium's ROI, while the second gives the segmentation of it; a succession of dice loss within contour loss (i.e multiplication between dice map and prediction map) was also used [98]. The winner of the challenging used a 3D U-net approach (as in ACDC) with an input volumetric

pre-cropping for not losing spatial resolution [94].

The standard 2D U-Net was implemented [95, 99, 100, 101] with a pyramid module for producing semantic signals at multiple scales [102]. While a multi-task CNN that, by sharing features at the end of the down-sampling path for pre or post ablation classification, was able to boost the final segmentation performance [103]. Transfer learning pre-trained on ImageNet dataset [104] for the down-sampling and fine-tuned with the training dataset was also explored [105]. Xin Yang et. al suggested a similar road but including a novel loss called focal positive to promote the learning of voxel-specific thresholds and at the same time to control the foreground sensitivity of the classification [106].

Despite, the idea to include local and global MRI information have been explored in two works. Tim Sodergren et. al propose a maximum-a-posteriori formulation that incorporates regional intensity and global shape prior where an autoencoder is used for capturing the complex intensity distribution for modeling it within a mixture of Gaussians [107]. While Zhaohan Xiong et. al uses a multi-scaled, dual pathway architecture that captures both the local and global tissue geometry extracting 15×15 patch respectively and classifying them as positive or negative pixel classification [108].

While Caizi Li et. al adds an attention mechanism inside 3D convolutions. Where after the atrium ROI cropping, respective blocks of the down-sampling path are placed in parallel among two units: Hierarchical Aggregation Module (HAAM) and Components of the Attention Unit (AU). While the HAAM mitigate the huge computational cost of the 3D kernel by the concatenation of five convolutional operations in a hierarchical way. The AU, implement a 3D convolution unit, followed by a sigmoid, in order to force the deep network to produce an attention mask [109].

Two classic methods were also proposed. In the first, the original MRI was converted in probability map then an atlas method within registration and labeling fusion was jointly combined with the level set for the final atria segmentation [110]. The second method is always based in multi-atlas, wherein the training phase the whole heart segmentation was constructed with the atlases unsupervised clustering according to the shape, then all clusters were registered to the target image and conditional entropy ranked them [111] or after a multiple image registration,

the final softmax pixel-wise layer combines individual U-Net outputs [112].

The first idea followed in this work is the use of 3D CNNs as the effective solution for the 3D LGE datasets segmentation. The starting point is the V-Net [15], which takes into consideration the spatial redundancy naturally present on the entire volumetric stack within 3D-kernels, showing good benefits in different cardiac segmentation problems [113], [15]. This architecture is modified to reduce the memory burden and speed-up its training. The last idea explored in this work is the sensible choice of a loss function, the joint combination of both the MSE and Dice Loss, which has been reported to be beneficial as the MSE minimizes global image details while the Dice Loss reduces local over-segmentation errors [15].

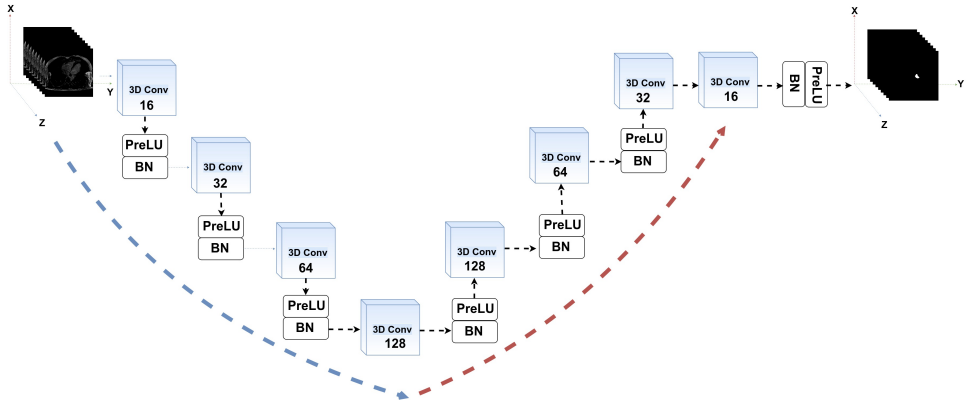

Figure 5.1 V-FCNN architecture. Input is the (XYZ) 3D MRI volume of size ($127 \times 127 \times 88$), also passed through the down-sampling path (blue arrow), represented by a 3D kernels Convolution Neural Network (CNN) able to progressively reduce the input volume slices. Then, the hidden features, at the end of it, are restored within 3D up-sampling kernels (red-arrow), ending in an output being a 3D mask of size ($127 \times 127 \times 88$). Both down-sampling and up-sampling paths consist of four 3D-convolutions blocks (blue boxes) followed by PreLU plus 3D-Batch Normalisation (BN). The number of feature maps for each convolution layers are $16, 32, 64, 128$ both in down and up-sampling.

5.2 Atrial Datasets

A population of 100 3D LGE-MRIs, and masks, were made available through the [2018 Atrial Segmentation Challenge] and used in this work. Images had a acquisition resolution of 0.625 x 0.625 x 0.625 mm^3.

5.3 Method

A volumetric fully convolutional network, V-FCNN, is designed with two main paths (see Fig.1): the volumetric down-sampling path as well as the volumetric up-sampling path. The volumetric down-sampling path has four 3-D convolutions blocks, each following by PreLU along with 3-D Batch Normalisation (BN) layers. This takes as input the entire (XYZ) volume and progressively reduces the size of each slice (XY) together with the number of spatial slices stack (Z). In this phase, the volume is compressed and presents both (XY) and (Z) reduction. In a complementary fashion, the up-sampling volumetric restores the compressed volume to its initial size, with every 3-D up-sampling convolution blocks being followed by PreLU and 3-D Batch Normalisation (BN). Each sampling direction of V-FCNN contains four blocks within 16,32,64,128 -3D kernels respectively (with the size fix to 3×3).

Image down-sampling, during a segmentation task, presents the problem of feature map reduction, followed by a strong spatial information loss. This problem has been addressed in V-Net [15], [113] by adding skip layers between down-sampling and up-sampling layers in order to fuse low-level features within high level features.

Our work explores an alternative approach to the skip path connections. In particular, we boost our model to capture fine details in two ways. First, the use of max-pooling operations is avoided in order to prevent the loss of spatial resolution (i.e if pools do not overlap well, pooling operation loses appreciable information where the objects are located in the image). The second idea is to use a loss layer that combines the MSE_{loss} (eq 2.76) and $DICE_{loss}$ metrics as an attempt to recover details in the image. The rationale is that the $DICE_{loss}$ term searches for local details in the volume data, as a consequence, the MSE_{loss} can be seen as a regularization that instead focuses on global features on the MRI volume. This bimodal loss also prevents the V-FCNN to fall in a local minimum; especially within small atrial regions. Specifically, the loss is defined as:

$$Loss = MSE_{loss} + \lambda \cdot DICE_{loss} = \frac{1}{Z}\sum_{s=0}^{Z}(y[s] - \hat{y}[s])^2 + \lambda \cdot \frac{1}{Z}\sum_{s=0}^{Z}\frac{2\sum_i^N y[s]_i \hat{y}[s]_i}{\sum_i^N y[s]_i + \sum_i^N \hat{y}[s]_i}, \tag{5.1}$$

where $y[s]$ are the Ground Truth (GT) binary slices, $\hat{y}[s]$ are the correspondent's prediction masks. s is an index through the spatial slices in the Z (z-axis). Then, the sums for i in $DICE_{loss}$ runs over all N pixels of the prediction masks $\hat{y}[s]$ and the $y[s]$ GT masks. Finally, λ controls the amount of $DICE_{loss}$ during the optimisation training process (set to $1e-3$).

The size of each MRI slice is reduced to 127×127 pixels using down-sampling bi-cubic interpolation. Please note this is a necessary operation linked to the vRAM (6-12GB) of current conventional GPUs, which not enough for storing high-resolution volumetric images. Finally, an up-sampling bi-cubic interpolation is finally used for restoring the mask to the size of 640×640 pixels.

5.3.1 Implementation details

For our experiments, we train the network with a number of epochs of 1000 up to convergence. The Stochastic Gradient Descent (SDG) was used with a learning rate of $1e-4$, while the momentum and weight decay are 0.9, $1e-5$ respectively. Furthermore, to increase the generalization of the network, data augmentation was used, finding particularly effective the random vertical and horizontal flip in close combination with plane translation. Input sequences were equalized in grayscale intensity with CLAHE (Contrast Limited Adaptive Histogram Equalization) [114], and the noise was minimized with a combination of High-Pass Filters and Gaussian blurring filters.

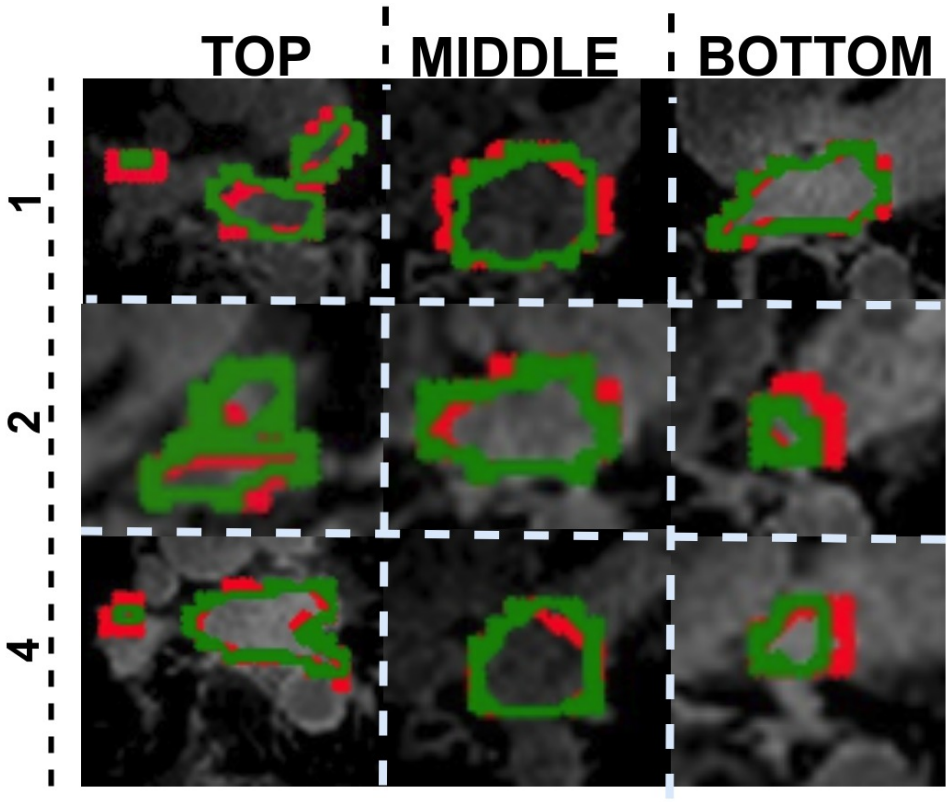

Figure 5.2 Visual comparison of the segmentations obtained from V-FCNN (green line) vs clinical ground truth (red line) in three different test patients (number 1, 2 or 4). The comparison is made at three different sections of the atrium: top, middle and bottom. Note how the V-FCNN is able to segment not only visually simpler slices (middle section) but also more complex cases (top and bottom sections).

5.4 Experiments

The experimentation phase are divided into two phases. In the first phase (preparation phase) 5 patients are used for validation (440 slices in total), and the rest (90) is used for training. In the second phase (competition), our algorithm was evaluated in the test-set with 54 patients (4752 atria slices in total).

Segmentation accuracy was measured with the 2D Dice Metric (DM) [66], which was subdivided into 3 regions of the atria: top (including the pulmonary veins), middle and bottom (including the valve plane that divides the atria and ventricle). Besides, 2D DM and the surface Hausdorff Distance (HD) [115] are computed for the entire atrial anatomy. The 3D Dice of the full volume is finally also measured at the second phase.

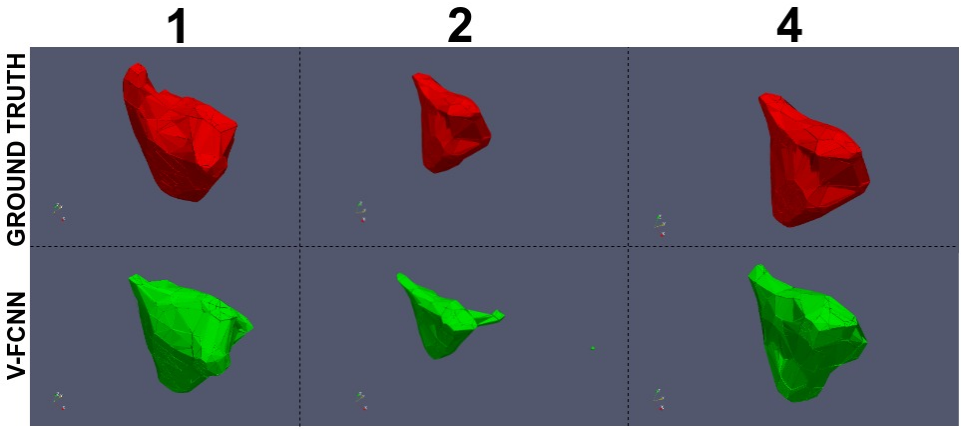

Figure 5.3 Visual comparison between ground truth (red) compared with those obtained by proposed V-FCNN (green). Note that, the mesh coarse resolution is related to the low number of triangles used.

Particularly, given a point p and a surface S, the distance $\varepsilon(p,S)$ is defined as:

$$\varepsilon(p,S) = min_{p^{'} \in S} d(p,p^{'}) \tag{5.2}$$

whereas the $d(\cdot)$ is the Euclidean distance between two points in a Euclidean space. Then, the HD between mesh surfaces $S_{GroundTruth}$ and S_{vfcnn} is given as:

$$HD(S_{GroundTruth}, S_{vfcnn}) = max_{p \in S_{GroundTruth}} \varepsilon(p, S_{vfcnn}) \tag{5.3}$$

The proposed V-FCNN achieved in the first phase an average 2D DM of 69.6 ± 16.1, 82.1 ± 1.4 and 78.0 ± 6.0 in the top, middle and bottom regions, respectively (see Table 1 and an illustrative example in Fig. 2). The HD ranged from 0.31 to 0.86 mm. The average 2D DM and HD were $76.58(7.87)$ and 0.59 respectively. The 2D and 3D DM in the competition phase were 92.5% and 85.1% respectively.

Visual inspection of the contours reveals a loss of accuracy on the top slices for two of the cases, creating an artificial shape flattening (please see patients 1 and 2 of Fig. 3), and on the top of the atria with the more variable anatomy of the pulmonary veins.

	TOP	***MIDDLE***	***BOTTOM***	***3D-MESH***
Patient number	**DM (%)**	**DM (%)**	**DM (%)**	**HD**
1	77.74 (8.46)	84.62 (2.40)	**76.99 (10.89)**	0.86
2	72.26 (9.50)	81.25 (1.57)	54.18 (33.70)	0.66
3	74.10 (8.92)	77.75 (1.36)	68.27 (14.75)	**0.31**
4	81.60 (0.74)	81.54 (1.20)	72.32 (14.32)	0.55
5	**84.48 (2.58)**	**85.36 (0.66)**	76.33 (7.05)	0.58

Table 5.1 Automatic segmentation results (2D Dice Metric and Hausdorff Distance) for all five test patients. Results report the mean and standard deviation, and are divided into three different atrium sections: top, middle and bottom.

5.5 Discussion and Conclusion

The exploitation of spatial coherence across different volumetric slices is an important resource for improving the accuracy of fully automated segmentation. Proposed V-FCNN achieves a good segmentation performance, mostly in the middle atrial section. Limitations occur in the top and bottom sections, caused by the presence of the pulmonary veins and the difficulty to identify where the atria and ventricle split. In the preparation phase the V-FCNN achieved a 2D DM between $82.05(3.43)$ (top case) and $69.23(14.92)$ (worst case), along with a 2D DM of 92.5% and 3D DM of 85.1% at the competition phase, satisfactory for using V-FCNN model in clinical practice.

A cardiac imaging study has a huge space coherence that many segmentation algorithms have exploited within different deep-learning techniques [15], [113], [15], [68], [116]. The use of 3D convolutional kernels [15], [113], [15] has the advantage of directly capturing the spatial information in each convolutional layer without adding extra parameters (i.e. the addition of a recurring network). This is the main reason that has driven us to use them in our optimized V-FCNN.

Proposed V-FCNN simplifies the V-Net [15] by removing the skip paths, in an effort to achieve a much quicker training convergence. The constitutive advantage of the skip paths is to increase the localization, but at the cost of a considerable slow down of the network speed (i.e GPU out of memory error) while propagating the gradient, at every iteration, forward and back through those paths [117]. We not directly compare with the default V-Net architecture, with the skip

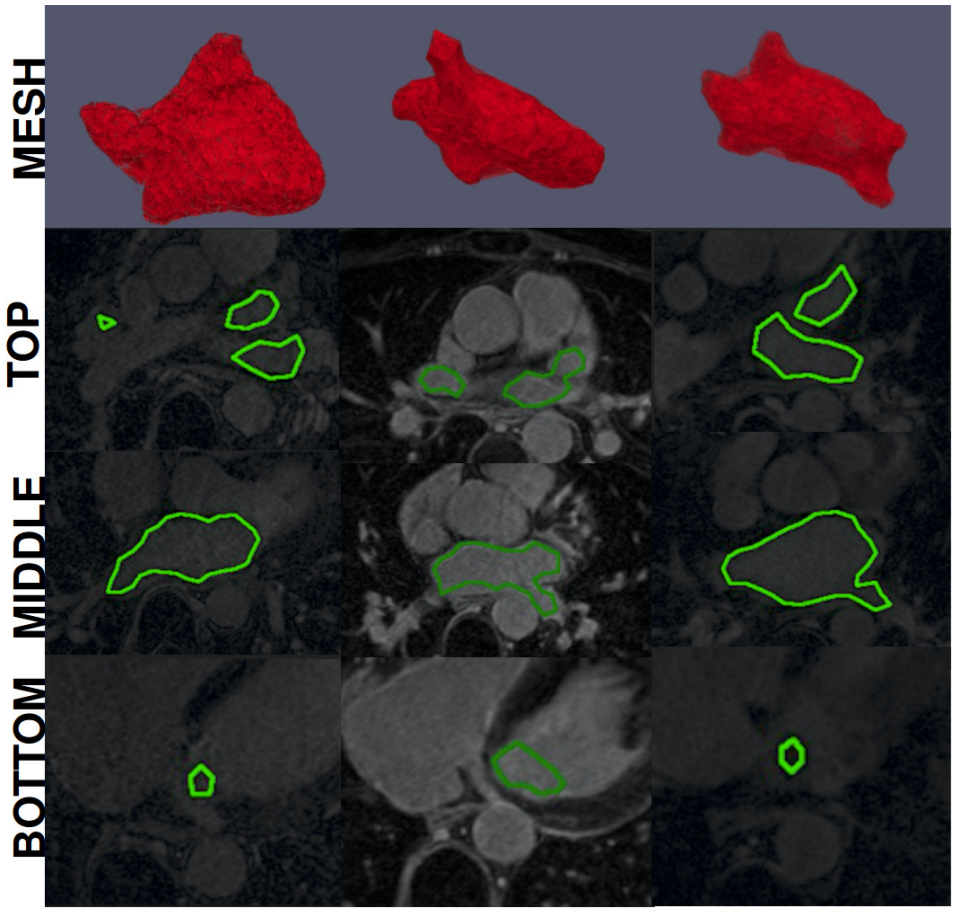

Figure 5.4 Exemplary result in 3 of the competition cases, illustrated with the 3D reconstruction (red, top) and 3 different sections of atrium segmentation: top, middle, and bottom.

paths, but our results suggest that the speed-up in training did not cause a significant loss in segmentation accuracy.

The choice of the loss function is an important consideration in any CNN, and the joint minimization of MSE and Dice Loss has been adopted in our solution. The interpretation is that the MSE looks at global volumetric features, while the Dice Loss (DL) regularize it trying to fit local details. This choice allowed the learning to avoid local minima and the corresponding slow convergence of using only MSE. The optimal weight between them, and the inclusion of further criteria such as an L1/L2 loss or a statistical distance to a library of existing cases, should be addressed in future extensions of this work.

The limitation of the 3D convolutional kernels is the large memory burden, and this is the reason why the original resolutions of the images were reduced (from 640 to 127 pixels). This image reduction is required due to the current hardware limitations (i.e. memory of the GPU) and

obviously caused a loss in performance. The concatenation of two CNNs, the second working at the full resolution cropping the region of interests containing the atria, is also a good strategy that will minimize the impact of this choice. An alternative is to work with the information at two levels of detail, which has shown to achieve good performance in this problem of atrial segmentation [118]. Recent evidence also suggests that not simply the concatenation, but the combination of two fields of view is a useful strategy to maximize the segmentation performance of MRIs [20].

An alternative strategy to 3D convolution is recurrent units [68],[116],[18], where recurrence is used to capture the redundancy between adjacent slices. The use of this approach is more memory efficient (i.e fewer parameters in the GPU global memory to capture recurring partners), also reporting improvements at the challenging apical slices of the left ventricle [68]. These solutions are limited by the vanishing gradient that unfortunately occurs within very long sequences, which can be partially avoided by imposing upper bound constraints on the backward gradient (i.e gradient clipping) or with a regularisation term [36]. Moreover, the creation of Long short-term memory (LSTM) co-processor, for future MRI scanner inference, requires several parallel MultiplyAccumulate units (MAC) (i.e due to multiple linear LSTM gates) which require huge amounts of memory bandwidth [119].

6 A Generative Model In Right Ventricle Segmentation

The clinical management of several cardiovascular conditions, such as pulmonary hypertension, need the assessment of the right ventricular (RV) function. This work addresses the fully automatic and robust access to one of the key RV biomarkers (i.e the ejection fraction) from the gold MRI standard imaging modality; that requires an accurate segmentation of the RV blood pool from CINE MRI sequences. Recent studies providing a snapshot of performance increasing through new deep learning models. Here we report a solution based on Fully Convolutional Neural Networks (FCNN) a well used deep learning model, where our first contribution is the optimal combination of three concepts: The convolution Gated Recurrent Units (GRU), the Generative Adversarial Networks (GAN), and the L1 loss function.

We show an improvement of 0.05 and 3.49 mm in Dice Index and Hausdorff Distance respectively with respect to the baseline FCNN. This improvement is then doubled by our second contribution, the ROI-GAN, that sets two GANs to cooperate working at two fields of view of the image, its full resolution and the region of interest (ROI). Our rationale is to better guide the segmentation networks learning by combining global (full resolution) and local Region Of Interest (ROI) features. The study is conducted in a large dataset acquired at our institution of $\sim$ 23,000 segmented MRI slices, and its generality is verified in a publicly available dataset.

This multi generative network explores the latent segmentation space through the implementation of cooperative models that independently integrate two different resolutions and consequently provides a generative strategy for searching global/local RV intensity distributions.

Moreover, such a training strategy promotes the final segmentation network to boost the overall segmentation performances.

6.1 Introduction

Cardiovascular diseases (CV) remain the leading cause of death worldwide [120], accounting for 17.3 million total deaths worldwide. In the management of these conditions, cardiac magnetic resonance (CMR) is considered the gold standard for the assessment of key biomarkers such as the blood pool volume or ejection fraction (EF) of the ventricular chambers [121]. The fully automatic and robust access to this important diagnostic and prognostic information is nevertheless missing in the clinical armamentarium.

The Left Ventricle (LV), has traditionally focused the clinical interest for the characterization of the disease progression, but in recent years a strong attention shift to the Right Ventricle (RV) has led to important findings for the conditions management as pulmonary hypertension, coronary heart disease, dysplasia and cardiomyopathies [122, 123].

Compared to the LV, the RV is a challenging anatomical structure to be characterized, mainly due to larger morphological variability and the thinner myocardial walls [124]. The RV biomarkers of volume, EF or cardiac output is conventionally assessed through the acquisition of a stack of short-axis (SA) slices. The interesting problem becomes the automatic RV segmentation in CMR SA slices, where the goals are the removal of the (intro- and inter-) observer variability and the immediate access to this information right after acquisition.

The attention on RV segmentation was initiated with a-priori probabilistic atlases [125, 126], using both shape and appearance information. The strength and weakness of this approach lay on the suitability of the cohort used to build the atlas. This solution will render a low performance in new anatomical configurations not accounted for in the training dataset.

In an attempt to alleviate this limitation, manifold learning techniques have been applied to better capture the variability of shape models, for example using Markov Random Field (MRF) [127]. Image gradient algorithms [128], region-merging [129] and graph-cut [130] based meth-

ods have been shown to be more compelling. The implementation of these ideas led to popular methods such as active contours able to reach reasonable performance, although with a dependence on the actual choice of weighting factors and the optimal initialization point.

In the last few years, deep-learning (DL) methods are being developed for extracting automatic spatial features. Particularly, Fully Convolutional Neural Networks (FCNN) can be considered the state of the art or the automatic segmentation of the RV [131, 132, 133].

In this work we have extended the capability of FCNNs, exploring two main ideas: first, modelling and exploiting the spatial redundancy (i.e spatial repetition of ventricular information) between adjacent SA slices; and second, guiding the FCNN to the useful RV features without the need of an automatically pre-localisation of the RV region of interest (ROI). The approach to exploit spatial redundancy is the incorporation of a recurrent unit in the middle of up-sampling and down-sampling path of the FCNN, an R-FCNN, a strategy that has been shown to improve the LV segmentation, especially at the apex [87].

Guiding an FCNN, to the correct image features, is a complex goal. The explicit ROI extraction is an approach followed by RV segmentation [134] and many medical applications [135, 136, 137] to facilitate the segmentation task. Mask-RCNN [138], is an example where the segmentation obtained from an FCNN is in close combination with an ROI-pooling mechanism capable to locally identify the bounding box of each object.

Our hypothesis is that there are still useful image features outside the ROI that can guide the FCNN, and that approaches that jointly learn detection and segmentation are desirable, avoiding focussing only on the ROI features. Some works explore this idea, where ROI prelocalisation becomes an additional sequential task in an end-to-end training chain [139, 140], but without an explicit use for guiding the segmentation.

However, a dual FCNN approach within local and global downsampling pathways, for two different MRI resolutions, was also used in atrial segmentation problem [118]. In this work, the local path only helps to scan every single patch of the image in order to classify it as negative or positive. In truth, this method differs from the principle of FCNN (i.e uses down-sampling filters to examine the whole pixel image) and approaches more to prior old segmentation tech-

niques, where the adding of the global path works as a multi-scale integration of global contexts.

The strategy to guide the FCNN to features within the ROI, without losing the features outside it, is inspired in the concept of the Coupled Generative Adversarial Networks (CoGAN) [141, 142], where a pair of corresponding images in different domains can be mapped in the same representation within a shared parameters strategy between two Generative Adversarial Networks (GAN). We adopt this concept by taking two versions of the same image, one at full resolution and another at the exact ROI around the segmentation mask, and we call it ROI-GAN. Note that the second image will only be needed at the training stage.

In this work, we thus explore the use of three existing models, a recurrent unit (R-FCNN) to exploit the spatial redundancy of a stack of SA slices, the concept of adversarial training (FCNN-GAN) to better guide the selection of features, and the use of the L1 loss [143, 144], and we propose the ROI-GAN as a solution to maximize the performance of FCNNs for the task of RV segmentation.

6.2 Material and methods

In this section, we present the datasets used in this work, and we review the concepts of the FCNN, the R-FCNN, the L1 loss, and the GAN training strategy using either FCNN or an R-FCNN. Finally, our ROI-GAN architecture is explained, with 3 possible variants that will be analyzed.

6.2.1 Datasets

Two datasets are used, the Twins-UK (i.e already described in 4.2 section) and a small public RV MICCAI dataset[124] to test the generality of the findings. On the other hand, the public RV MICCAI dataset was used to refine weights (16 subjects with two-time points segmented, 250 slices) and the performance was evaluated on its Test2Set, blind cohort used for benchmarking (i.e another set of 16 subjects with a similar number of slices). This public dataset is composed

of subjects with a variety of disease conditions, with an average age of 55.5 ± 17.5 and where 70% of them were male.

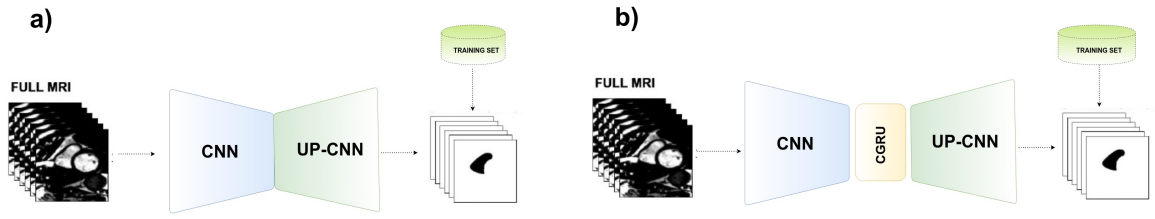

Figure 6.1 FCNN and RFCNN architectures. The FCNN (panel a) is a combination of a decoder and encoder paths. The decoder path (blue trapezoid) consists of six convolution layers (stride of 2) followed by ReLU and Batch Normalization (BN). The up-convolution path (green trapezoid) made by deconvolution layers in combination with LeakyReLU (set to 0.2) and Batch Normalization (BN). The R-FCNN (panel b) is similar to FCNN but a convolution-GRU (yellow rectangle) is used between the decoder and encoder in order to model and exploit the spatial MRI redundancy.

6.2.2 The FCNN/R-FCNN

An FCNN is the core of the DL solution, taking a stack of SA slices as input and returning the segmentation mask. Then, the FCNN extracts features from the image in a first decoding path, and these are gradually restored to the original image size through a decoding block used to infer the final binary mask. One of the key features of an FCNN are the skip paths connections between convolution and deconvolution layers for fusing mid-height level features together. Each connection concatenates feature maps from the encoding to the decoding blocks.

The FCNN (Fig. 6.1 panel (a)) considers each slice of the SA input independently. The extension in an R-FCNN (Fig. 6.1 panel (b)) is to take the full SA stack as an input, as a sequence of slices from base to apex where the current slice depends on the previous observed ones [87].

The R-FCNN has the same encoding and decoding structure of an FCNN (i.e six encoding and decoding blocks followed by ReLU and Batch Normalization (BN) operators) but, in the middle of both, a Convolution Gated Recurrent Unit (C-GRU) [145, 146] is used. The C-GRU unit presents two specific gates designed to control the spatial information inside: a rest gate r_s and an update gate z_s defined as follow:

$$
\begin{aligned}
r_s &= \sigma(W_{hr} * h_{s-1} + W_{xr} * x_s + b_r) \\
z_s &= \sigma(W_{hz} * h_{s-1} + W_{xz} * x_s + b_z)
\end{aligned}
\tag{6.1}
$$

Here, $\sigma(\cdot)$ is the sigmoid function and the h_{s-1} represents the hidden activation learned at the previous SA slice $s-1$ with s the current index of volumetric space slices.

The W_{hr} and W_{hz} are the weight matrices of dimension $D \times 8 \times 8$, with D the number of features maps in the down-sample layer and b_r, b_z bias vectors.

In this notation, $*$ defines the convolution operation. The reset gate switch (on or off) the signal coming in input to $\hat{h_s}$; where $\hat{h_s}$ is called the candidate activation, defined as:

$$
\hat{h}_s = \tanh(W_h * (r_s \odot h_{s-1}) + W_x * x_s + b) \tag{6.2}
$$

The $\odot$ denotes the dot product and W_h, W_x the recurrent weight matrices for h_{s-1} and x_s, while b still to be a bias vector.

Then, the final C-GRU activation is:

$$
h_s = (1 - z_s) \odot h_{s-1} + z_s \odot \hat{h_s}. \tag{6.3}
$$

6.2.3 The L1 loss

The training of a network is guided by the metric used to define the error, and the L1 loss [143, 144] has shown to be a good addition to the total loss. The L1 distance used is given as:

$$
L_{L1} = \frac{\beta}{n} \mid x_i - y_i \mid \tag{6.4}
$$

This metric measures the mean absolute value of element-wise difference among the network

output x^i and ground truth y^i, where, β (set to $5e^{-6}$) is a regularization constant parameter for controlling the quantity of L1 loss used and n is the set of training spatial sequences cases (i.e slices).

6.2.4 The GAN

Two neural net architectures, the generative and the discriminative, compete in a GAN 2.8.3 to perform a task.

We embrace this concept to our problem so that the FCNN or R-FCNN become generators of binary masks (note that their input here is not a distribution of random numbers, but the distribution of MRI images), and we add a new CNN to serve as discriminator that will try to identify if the binary mask is "fake" (i.e output of the generator) or "real" (i.e the ground truth mask). See the top of Fig. 6.2 for an illustration of a GAN architecture.

The CNN discriminator network is thus trained to distinguish how much the ground truth deviates from that produced by the FCNN generator. This information, generated by the discriminator, is back-propagated towards the generator, which later uses the previous knowledge for producing undistinguishable masks from the corresponding ground truth. Besides, in order to avoid deterministic generators, Gaussian Noise (GN) is added by dropping the first three up-sampling layers of the generator.

The adversarial process is summarized by the maximization of the following loss:

$$\begin{aligned} min_G\, max_D L_{GAN}(D,G) = E_{x\sim MRI_{real}(x)}[log(D(x))] + \\ + E_{x\sim MRI_{fake}(x)}[log(1-D(G(x)))] \end{aligned} \tag{6.5}$$

where D represent the discriminator, G the discriminator and x is the set of binary masks sampling from $MRI_{real}(x_k)$ and $MRI_{fake}(x)$.

This adversarial loss is also combined within the minimum squared error MSE loss function between the generator output and the ground truth:

$$L_{MSE}(G) = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2 \tag{6.6}$$

The two loss functions are combined through a λ regularization parameter (set to $5e^{-3}$) that control the amount of GAN loss taken into account:

$$L_{TOTAL}(G,D) = (L_{MSE}(G) + \lambda L_{GAN}(D,G)) \tag{6.7}$$

6.2.5 The ROI-GAN

The ROI-GAN model (Fig. 6.2), takes inspiration from the idea of CoGAN [141, 142], which is adapted for working with the same image but at two different fields of view: one on a global level (i.e original full resolution MRI image), and another at the local ROI level. We will thus refer to the global (working with the full resolution) or local (working with the cropped image) generators and discriminators in each of the two collaborative GANs.

Remark that the cropped images needed for the second set of images, the ROIs, are simply the bounding boxes enclosing the ground truth segmentation, and are only required for the training phase of the architecture.

The local GAN inform and better guide the global GAN. This is articulated through a mechanism of parameter sharing between the generators and discriminators of the two GANs in an attempt to intensify the attention on the correct subset of mid-level features.

The training process is sequential: first, the cropped MRI images are segmented (i.e forward pass on the local generator) with a corresponding backward propagation of the loss by comparison to the segmentation ground truth. Second, the updated parameters of the local generator are passed to the global generator by using the weights-sharing connections that are enabled on the first three up-sampling layers. Then, the global generator repeats the forward and backward process. This third step is the training of the discriminator and the backpropagation of the total network gradient from the discriminator to both generators.

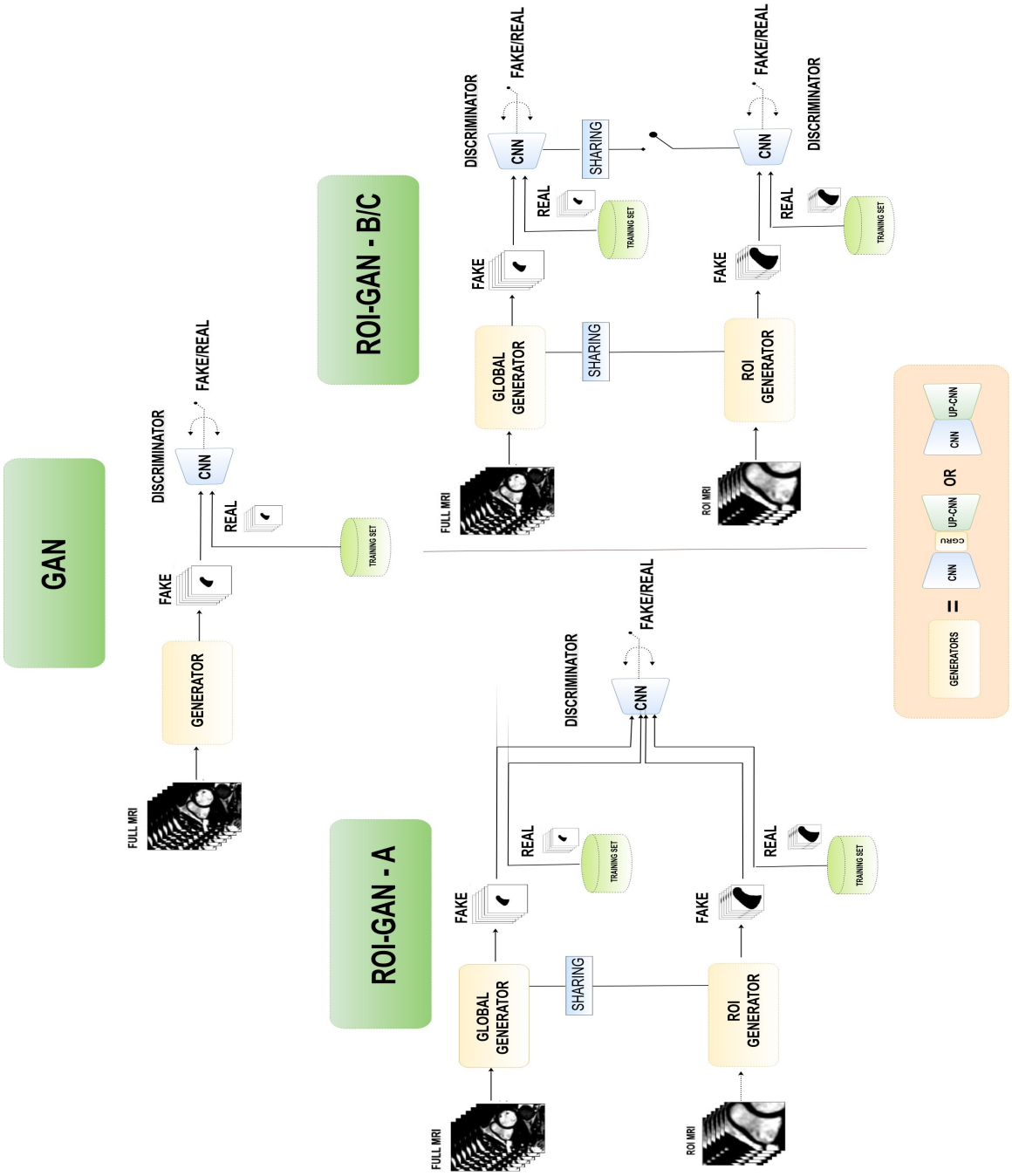

Figure 6.2 The three ROI-GAN architectures studied in this work. TOP: basic GAN architecture. BOTTOM LEFT: the ROI-GAN-A, where masks at two different sizes are feeding the same discriminator CNN. BOTTOM RIGHT: the ROI-GAN-B/C architectures, where two different CNN are used as discriminators, one for each image size, either in coordination (i.e sharing parameters) in B configuration, or independently in C configuration.

Three different strategies (identified by the letters A, B or C) for the discriminator are designed and compared in this work. ROI-GAN-A (Fig. 6.2 bottom left) uses one single discriminator fed by the images at both fields of view (full and ROI), motivated by the idea of maximizing the interplay between the two generators through the sharing of the same discriminator. The alternative is to use two discriminators (Fig. 6.2 right), one per generator, in the ROI-GAN-B. The third option is an intermediate solution, where the two discriminators are allowed to share weights between them, as set in the ROI-GAN-C.

6.3 Results

The baseline for this study is the performance of the FCNN to fully automatically segment the RV. This section presents the gradual improvement from this baseline by applying the idea of recurrence, L1 loss, GAN and finally the proposed ROI-GAN. Results will show how these concepts do not always complement each other. Illustrative examples of the segmentation performance are provided in Fig. 6.4 and Fig. 6.5.

6.3.1 Metrics

Segmentation performance is evaluated for endocardial contours with two different metrics: DI 2.74 and HD 2.75. These metrics are provided for three anatomical regions of the heart (basal, mid-ventricular and apical regions) for the different challenges that they face. The middle of the right ventricle is the most stable and consistent part, but the basal (base of the ventricle) suffers from quite a bit of variability in the shape of contours, and the bottom (apex of the ventricle) usually renders the worst performance due to the small size of the contour and poor contrast in the image.

6.3.2 The added value of a recurrent unit, GAN and L1 loss

Fig. 6.6 shows how the R-FCNN introduces a significant improvement over the FCNN in the RV apical (low) region, with an increase of DI of a 52% (from 0.38 to 0.58), and a reduction of

the HD in a 72% (from 13.60 to 3.82). The other two regions, basal and mid-ventricular, show a similar performance being only slightly worse at the DI of the basal region.

The addition of the L1 loss, to the baseline FCNN, also improves the performance at the apical region of the RV, both in DI and HD, with a small gain in both metrics at the mid-region, but with a drop of 0.03 in DI at the basal region.

Finally, the GAN-FCNN improves the performance with respect to the FCNN in all regions and metrics, but with a small impact (DI increase of 0.01-0.02, HD reduced in 0.5-1 mm) except for the large reduction of the HD in the apical region.

The combination of L1 and R-FCNN leads to worse results than using any of these two approaches in isolation at the apical region but matches the best performance of the other two notions in the additional two regions.

On the contrary, the combination of L1 and GAN leads to better results than using any of these two theories in isolation, in all regions and using both metrics. The best performance is provided by the FCNN+GAN+L1 and will be used as a baseline for the next experiments.

Fig. 6.8 shows how the GAN and R-FCNN combination does not have any benefit, and that the performance of adding the L1 loss is even worse.

6.3.3 ROI-GAN with an R-FCNN provides the best performance

The three versions of the ROI-GAN architecture are first evaluated with an FCNN as the generator, showing a drop in performance with respect to the FCNN+GAN+L1, see Fig. 6.8. On the contrary, the ROI-GAN using R-FCNNs as generators increase the performance, see Fig. 6.7, being the ROI-GAN-A the best in average in all regions.

In more detail, ROI-GAN-A addresses the best in all scores but in the DI of the mid-region, where ROI-GAN-B produces the best results, and the HD in the low apical region, whereas the FCNN+GAN+L1 gives the best score.

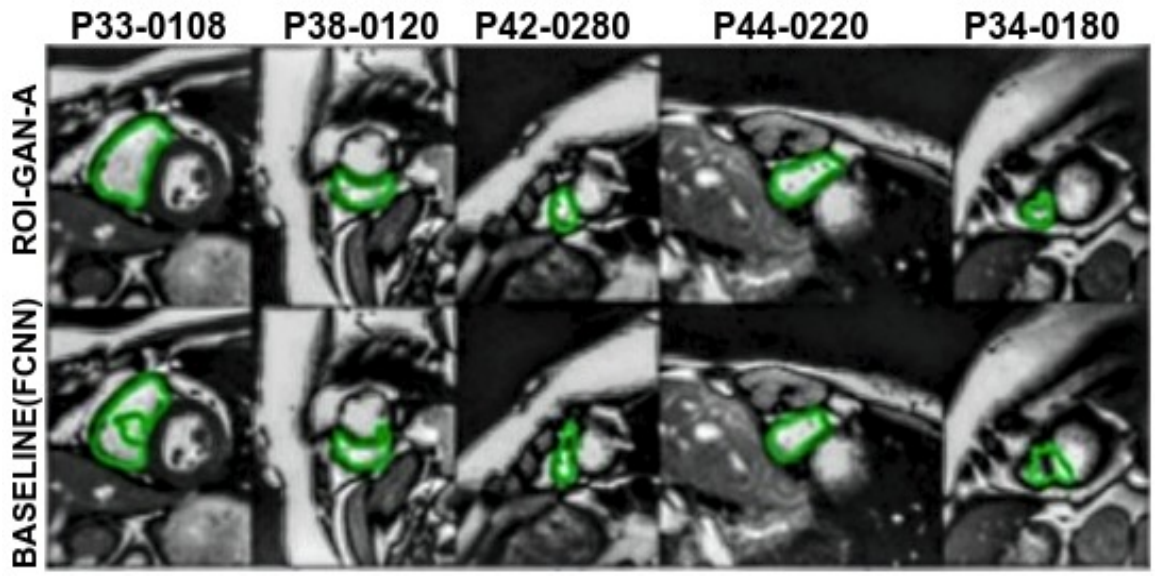

Figure 6.3 Examples of automatic segmentation results in images from the RV MICCAI dataset. These are instances where ROI-GAN-A showed superior segmentation results in comparison with the baseline FCNN.

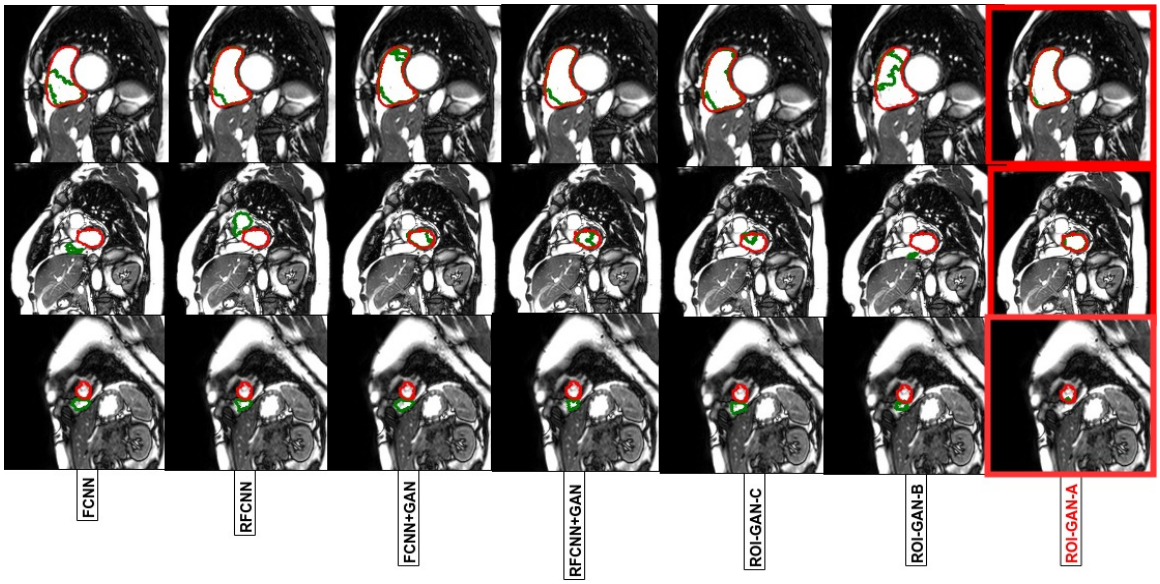

Figure 6.4 Illustrative segmentation results on our in-house Twins-UK dataset, comparing neural networks predictions (green line) to the ground truth (red line). The ROI-GAN-A (last column) shows a good match both in an easy case (first row, a slice from the basal of the RV) and in a difficult case (third row from the apical low region or the RV).

6.3.4 Generalization of results

To confirm the results obtained in our Twins-UK dataset, the performance of the baseline FCNN and of proposed ROI-GAN-A architecture are tested in the RV MICCAI Challenge 2012. Illustrative examples are provided in Fig. 6.3.

Table 6.2 shows how the ROI-GAN-A clearly raises over the FCNN in DI and HD, with gains of 0.05 and of 5.09 mm respectively. This gain in performance is of similar magnitude than the one observed in our dataset, where an average of 0.05 and 3.49 mm was observed.

The performance of the ROI-GAN-A, a fully automatic method, is close to being the best method in published literature, some of them semi-automatic (see Table 6.2), which are able to further improve the DI in 0.03 and reduce the HD of 0.75mm.

One last test is conducted to evaluate the robustness in the extraction of clinical indexes such as volume or EF: a linear regression analysis between manual and automated endocardiac areas, for both ROI-GAN-A and FCNN, reveal a R correlation coefficient of 0.9642 and 0.8899 each.

		BASAL	***BASAL***	***MID***	***MID***	***APICAL***	***APICAL***
METHODS	**FA/SA****	**DM**	**HD**	**DM**	**HD**	**DM**	**HD**
FCNN	FA	0.87(0.20)	3.19(6.20)	0.73(0.28)	5.01(12.98)	0.38(0.37)	13.60(22.79)
R-FCNN	FA	0.86(0.21)	2.79(3.03)	0.73(0.28)	4.50(10.76)	**0.58(0.30)**	**3.82(10.83)**
FCNN+L1	FA	0.84(0.25)	3.33(6.12)	0.74(0.26)	3.33(8.78)	0.47(0.36)	4.12(10.30)
R-FCNN+L1	FA	0.86(0.24)	2.90(2.79)	0.74(0.27)	5.00(12.66)	0.45(0.36)	8.38(17.44)
FCNN+GAN	FA	0.88(0.17)	2.87(4.65)	0.75(0.27)	3.92(10.03)	0.40(0.36)	4.63(9.06)
R-FCNN+GAN	FA	0.87(0.20)	2.86(5.57)	0.72(0.28)	4.61(12.33)	0.43(0.36)	9.17(18.50)
FCNN+GAN+L1	FA	0.88(0.18)	2.68(3.99)	0.76(0.25)	3.81(10.44)	0.42(0.38)	2.68(3.99)
R-FCNN+GAN+L1	FA	0.87(0.20)	3.31(7.63)	0.72(0.27)	5.10(13.55)	0.41(0.33)	14.46(25.01)
ROI-GAN-A-FCNN	FA	0.85(0.22)	3.30(6.61)	0.75(0.27)	3.35(8.64)	0.46(0.35)	3.30(6.61)
ROI-GAN-A-R-FCNN	FA	**0.89(0.18)**	**2.43(2.21)**	0.77(0.22)	2.67(6.67)	0.49(0.33)	6.03(14.49)
ROI-GAN-B-FCNN	FA	0.87(0.21)	2.72(3.41)	0.75(0.27)	3.77(9.76)	0.37(0.37)	9.07(17.23)
ROI-GAN-B-R-FCNN	FA	0.87(0.22)	2.84(4.60)	**0.78(0.22)**	**2.64(6.42)**	0.47(0.35)	4.29(9.89)
ROI-GAN-C-FCNN	FA	0.86(0.23)	2.75(2.75)	0.71(0.29)	5.70(14.49)	0.37(0.37)	11.56(20.51)
ROI-GAN-C-R-FCNN	FA	0.85(0.25)	3.33(7.24)	0.76(0.25)	3.88(10.93)	0.47(0.364)	7.29(16.56)

Table 6.1 Segmentation performance results on the Twins-UK dataset. DI: Dice Index; HD: Hausdorff Distance (mm); FA: fully automatic; SA: semi automatic.

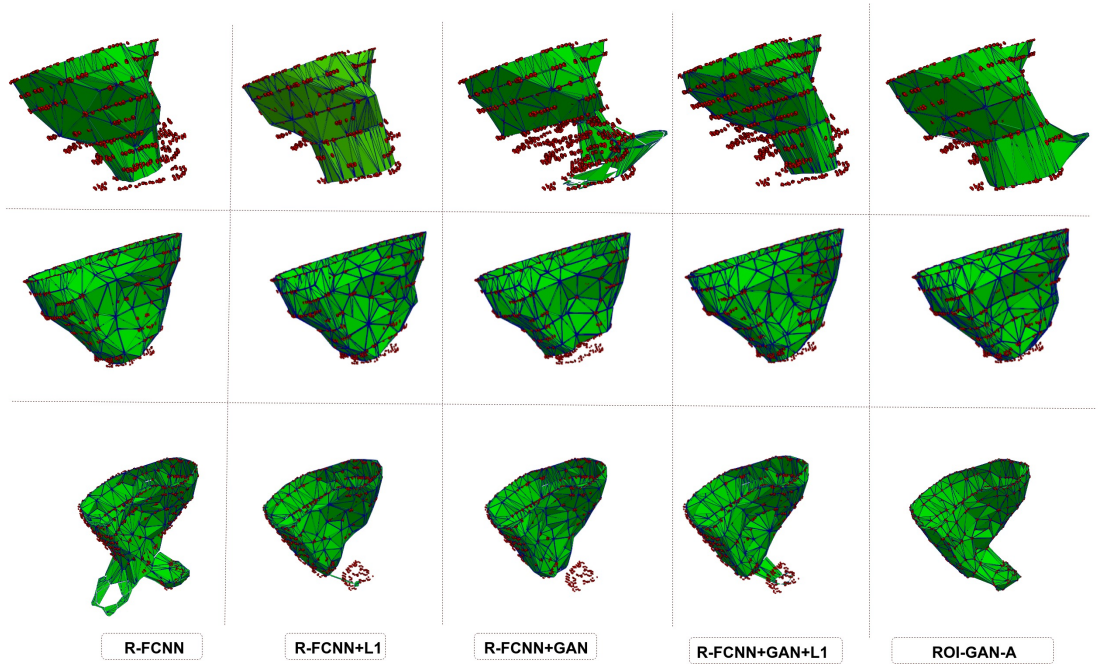

Figure 6.5 Examples of the reconstructed 3D anatomies of the RV, where the model prediction (green surface) is compared to the ground truth contours (red points).

Methods	FA/SA**	DM	HD
ROI-GAN-A	FA	0.80 (0.22)	8.03(4.41)
FCNN [Our baseline]	FA	0.75(13.12)	13.12(10.36)
Avendi et al. [13]	FA	0.82 (0.16)	8.03 (4.41)
Ringenberg et al 2014 [147]	FA	0.83 (0.18)	8.73 (7.62)
Zuluaga et al 2013 [148]	FA	0.73 (0.27)	12.50 (10.95)
Wang et al 2012 [149]	FA	0.61 (0.34)	22.20 (21.74)
Ou et al 2012 [150]	FA	0.61 (0.29)	15.08 (8.91)
Maier et al 2012 [129]	SA	0.77 (0.24)	9.79 (5.38)
Nambakhsh et al 2013 [151]	SA	0.56 (0.24)	22.21 (9.69)
Bai et al 2013 [152]	SA	0.76 (0.23)	9.77 (5.59)
Grosgeorge et al 2013 [128]	SA	0.81 (0.16)	7.28 (3.58)

Table 6.2 Segmentation performance results on the RV Test2Set MICCAI of public dataset. DI: Dice Index; HD: Hausdorff Distance (mm); FA: fully automatic; SA: semi automatic.

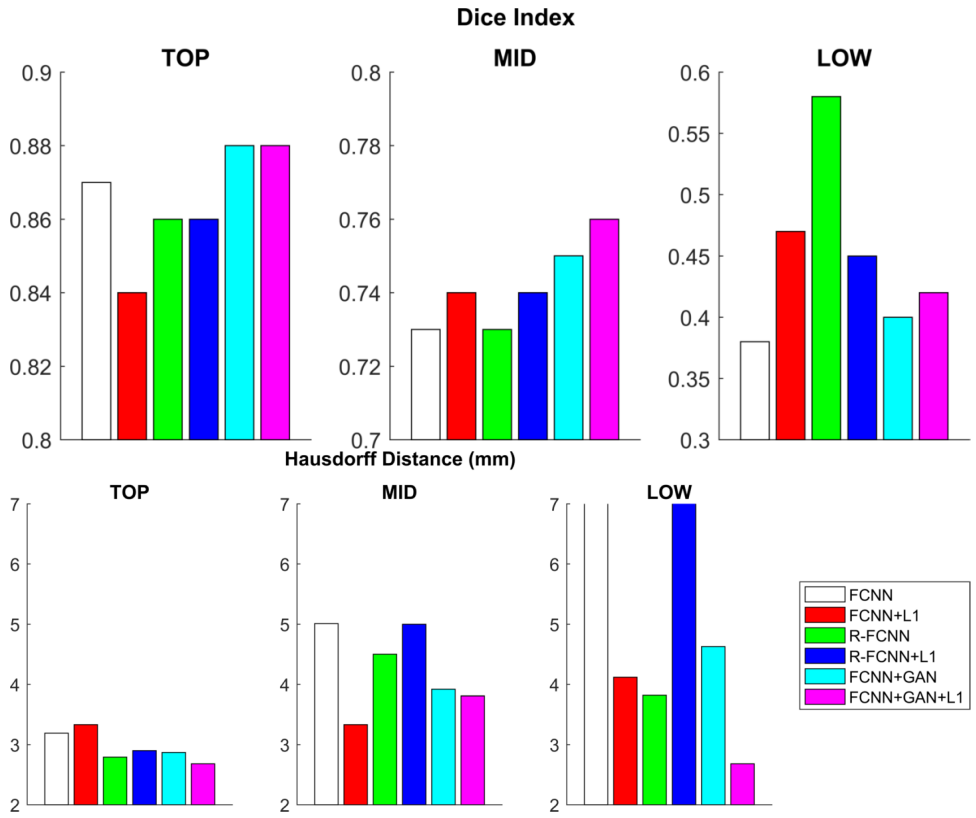

Figure 6.6 Evaluation of the added value of a recurrent unit (R-FCNN), the adversarial training (FCNN+GAN), and their combination with the L1 loss. Note that the HD bars in the LOW region for FCNN and R-FCNN+L1 reach larger values than the ones displayed in the plot.

6.4 Discussion and Conclusions

The performance of the FCNN for the task of RV segmentation has been enhanced by the combination of the three existing approaches (R-FCNN, FCNN-GAN, L1 norm) through the ROI-GAN, a novel interpretation of the coGAN where two GANs are set to cooperate at two fields of view (general and local, or full resolution and ROI-focus). The best combination of the three existing view achieved an improvement of 0.05 and 3.49 mm in DI and HD with respect to the baseline FCNN and that improvement was multiplied with the ROI-GAN-A architecture.

The combination of local and global features through an ROI-GAN has thus provided a benefit. The rationale sought was to enable the global FCNN to learn the useful features through the help of the local FCNN, coordinating their training with a generative adversarial game and sharing parameters (i.e. a CoGAN). CoGANs were proposed to force generators to learn

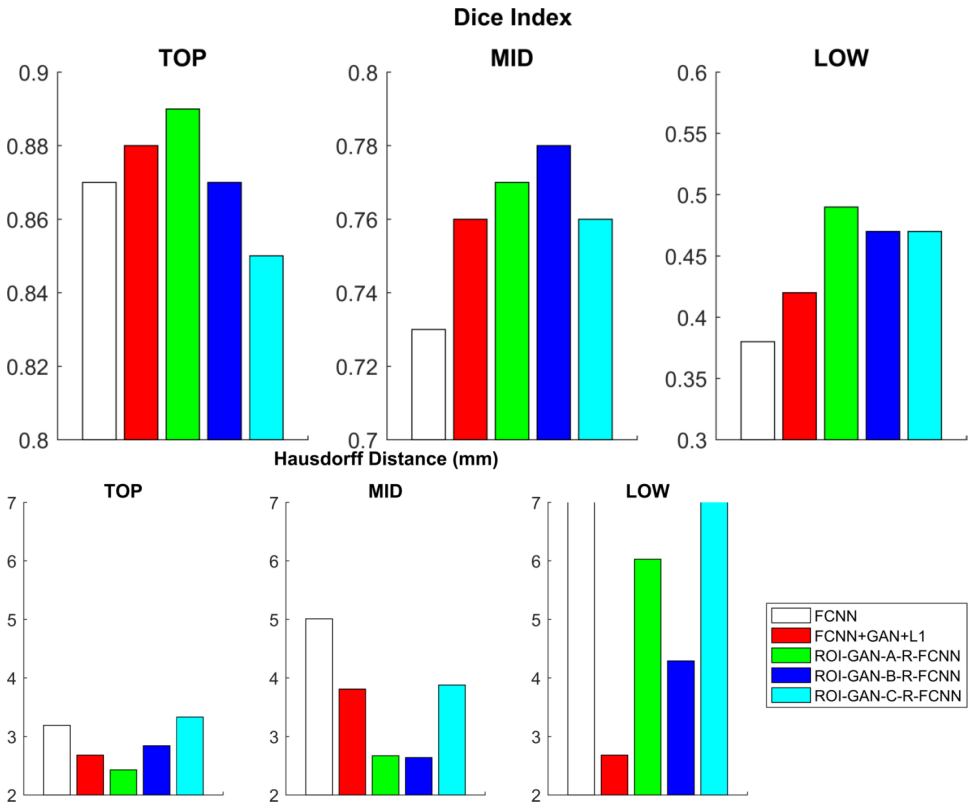

Figure 6.7 Benefit of the ROI-GAN over the baselines FCNN and FCNN+GAN+L1. Note how the gain from an FCNN to an FCNN+GAN+L1 is doubled with an ROI-GAN-A with an R-FCNN in all metrics but the HD of the low apical region.

from different data distributions and reported excellent performance in the problem of Unsupervised Domain Adaptation [141]. However, the problem of image segmentation is linked to generators optimization (FCNN/R-FCNN) where the latent information, extracted from the different spatial resolutions, resides (i.e. depth and color image, as in [141]), and we interpret this to be the main reason why sharing parameters was actually only beneficial at the generator, and not at the discriminator of the GANs (performance of ROI-GAN-A being superior to the ROI-GAN-C).

An interesting experimental finding was that the ROI-GAN-A architecture achieved the constructive coordination of the R-FCNN and GAN, which otherwise would result in a drop in segmentation performance (see Table 6.1). The GANs have been proved to be strategical for enhancing the learning generalization (i.e. a better loss regularisation) [143, 144], and an R-FCNN models the spatial coherence as a set of connections to the previous slices (data input prepared with image slices from basal to apical of the heart) - these two study are not in appar-

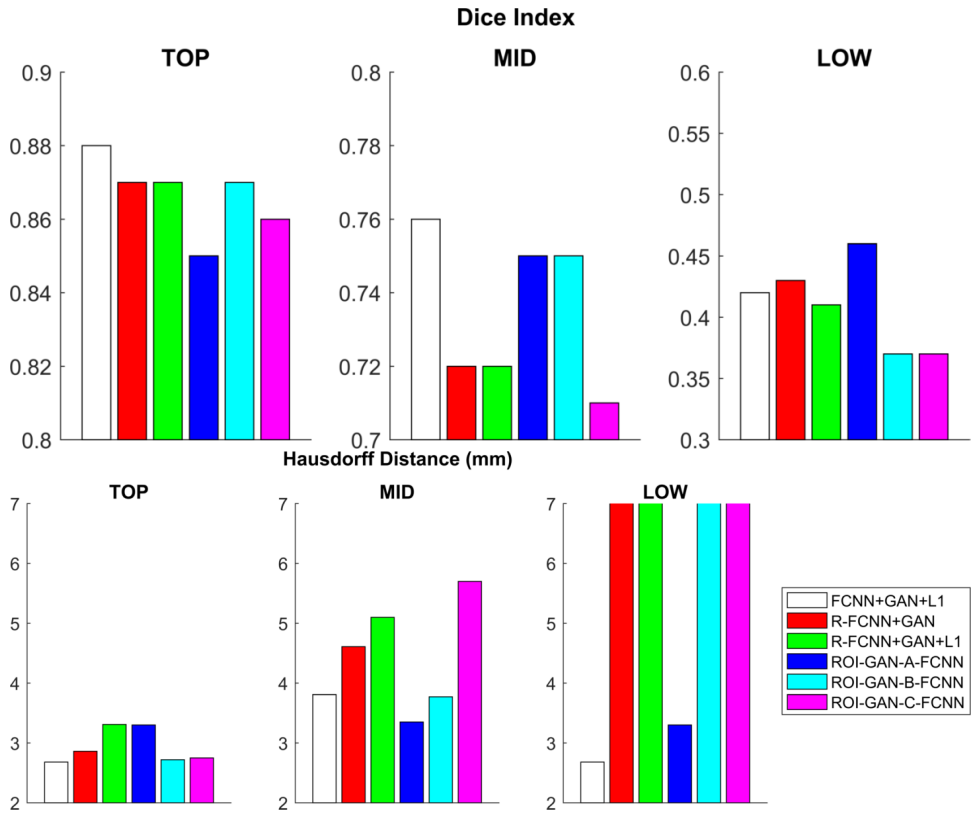

Figure 6.8 Strategies that do not improve the FCNN+GAN+L1 performance: the addition of the recurrent unit, or the ROI-GAN without a recurrent unit.

ent conflict, and the reason why these two concepts did not work together in a single FCNN, remains elusive to us. A possible explanation is the strong penalty that the GAN term imposes on MSE loss, that will be detrimental for a proper back-propagation through the C-GRU unit.

In this work, we illustrate that the problem of RV segmentation is, in fact, the combination of three problems, each of the three sections of the RV presents different challenges. While the basal slices present a minor difficulty in localization but a great anatomical variability, the bottom apical slices present a great difficulty in localization but a much simpler shape (i.e RV collapses towards a circular structure), and mid slices will be an intermediate problem. As a consequence, there is not a single architecture being the optimal solution for all these 3 problems. In the basal and middle slices, an ROI-GAN solution outperforms the rest (A or B configurations for basal or mid slices), but the best solution to capture the apex of the RV is the R-FCNN. Further research is needed to design the architecture that adapts to the anatomical region present in the image.

Redundancy across space, or time [153], is clearly a useful resource in the segmentation task. Exploiting the spatial redundancy is the rationale of the R-FCNN, and our results confirm the initial findings at the apex of the Left Ventricle (LV), where the main gain was observed compared to the rest of the anatomy [87]. Then the RV has a much greater anatomical variability, but this did not prevent a recurrent unit to better constrain the segmentation in the challenging apical slices. Nevertheless, the use of recurrent unit may not be an optimal solution, since it is continuously limited by the problem of vanishing gradient that may decrease the overall performances [36]. Further research is needed to study alternatives such as 3D FCNNs, where 3D convolution capture all the spatial coherence/redundancy [154].

The evaluation methodology followed in this work was designed to examine the thought proposed while minimizing possible confounding factors regarding the learning rate. All FCNNs had the same number of convolution layers, up-convolutions, ReLU and BN units. Besides, the number of epochs and the learning rate were equal in the two datasets used, where at every batch size each network take as input a sequence of consecutive SA images (or the corresponding number of stack images in an R-FCNN). Nevertheless, we cannot claim an independence of all confounding factors (i.e. the characteristics of the datasets used).

Initial evidence of the generality of our findings was provided by testing the final proposed solution in the best public dataset available to our knowledge, where a distinct improvement in comparison with the baseline FCNN was found (i.e. 0.80 (0.22) vs 0.75 (13.12) in DI and HD respectively). This experiment also showed that another solution, based on classical and simple concepts such as thresholding, still achieves better results [147], motivating the hypothesis that a combination of classical and deep learning approaches is an interesting direction of further research.

Future works will explore the use of 3D convolutions within the ROI-GAN, where the 3D generators (i.e global and local) should extract the spatial information in the better way without any R-FCNN vanishing gradient problem [36], or multiple generators able to see different cropping scales at input MRI sequences.

7 Conclusions and future directions

7.1 Overall picture

The current consensus, consider the FCNNs an excellent starting point due to their improved performance in a large variety of medical image analysis problems [155].

One of the key strengths in an FCNN is its ability to extract the spatial patterns in 2D, and its optimal extension to the third dimension or the temporal domain is one of the open research questions where the integration of spatiotemporal information is needed. Particularly, in space, two different strategies are possible: concatenation of CNN and recurrent units [87] or the use of 3D kernels.

However, if on one side the use of LSTM/GRU recurrent units has partially solved the vanishing gradient problem, particularly with long sequences, as explained in Chapter four, on the other hand, the use of 3D kernels is highly motivated by the idea that they should better capture the spatial patterns, initially restricted in a CNN to the 2D domain. Whereas, the recurrent units, are useful when working with 7-8 spatial slices to better segment the apex but not when working with a large number of slices.

Nevertheless, both recurrent units and 3D kernels can be applied for the whole cardiac sequences segmentation, as discussed in Chapter five a combination of R-FCNN is applying for exporting the segmentation in all thirty cardiac phases. While, in Chapter three, the cyclic changes of abdominal fetal aorta signal are taken in thought by the *CyclicLoss* (CL) that can be future translated for signal ECG application [156].

The 3D convolution architecture has also shown its usefulness in temporal problems outside the medical field, as an explicit factorization of 3D kernels in 2D spatial and 1D temporal yields an accuracy gain in different benchmarking datasets [157]. To bring spatial volumetric and temporal information together 4D kernels should be designed where volumetric 3D filters are concatenated together with 1D temporal. In this case the challenge is the variability in the duration of each heartbeat, and its automatic detection, to prepare the data with a regular dimension.

However, for the 3D kernels, the image down-sampling is needed because the actual available GPU memory is not enough to contain huge volumetric sequences at high-resolution. A partial solution to avoid downsampling problem is related to crop the anatomical region and at the same time focus the attention on the Region Of Interest (ROI).

Indeed, in Chapter 6, a novel generative adversarial solution was proposed to be able to combine the local resolution with the full once but further analysis with 3D kernels are also needed.

7.2 Future work

The cardiovascular temporal dynamics, and its modeling through deep-learning systems, allows creating anatomical motion shape priors that can be given as input to machine learning systems to learn latent survival features [158]. However, the temporal aspect, taken into consideration in this thesis, should be also linked to the volume and complexity of the available data. Whereas, multimodal imaging, lifestyle factors and genotyping they are all aspects to be considered together [159].

Then, the future work should pay attention to the intersection of multiple acquisition modalities such as MRI and Ultrasound together with genetic data, where the main purpose is creating accurate predictive survival systems. Therefore, the development of new losses and the creation of new GAN networks should also take into consideration and to be inserted in the multimodal data aspect.

Bibliography

[1] O. Bernard, A. Lalande, C. Zotti, F. Cervenansky, X. Yang, P. Heng, I. Cetin, K. Lekadir, O. Camara, M. A. G. Ballester, G. Sanroma, S. Napel, S. Petersen, G. Tziritas, E. Grinias, M. Khened, V. A. Kollerathu, G. Krishnamurthi, M. Roh, X. Pennec, M. Sermesant, F. Isensee, P. Jger, K. H. Maier-Hein, C. F. Baumgartner, L. M. Koch, J. M. Wolterink, I. I?gum, Y. Jang, Y. Hong, J. Patravali, S. Jain, O. Humbert, and P. Jodoin. Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: Is the problem solved? *IEEE Transactions on Medical Imaging*, pages 1–1, 2018.

[2] Masakazu Matsugu, Katsuhiko Mori, Yusuke Mitari, and Yuji Kaneda. Subject independent facial expression recognition with robust face detection using a convolutional neural network. *Neural Networks*, 16(5):555 – 559, 2003. Advances in Neural Networks Research: IJCNN '03.

[3] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):640–651, 2017.

[4] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

[5] Eunbyung Park. Manifold learning with variational auto-encoder for medical image analysis. 2015.

[6] Wei Dai, Joseph Doyle, Xiaodan Liang, Hao Zhang, Nanqing Dong, Yuan Li, and Eric P. Xing. SCAN: structure correcting adversarial network for chest x-rays organ segmentation. *CoRR*, abs/1703.08770, 2017.

[7] Simon Kohl, David Bonekamp, Heinz-Peter Schlemmer, Kaneschka Yaqubi, Markus Hohenfellner, Boris Hadaschik, Jan-Philipp Radtke, and Klaus H. Maier-Hein. Adversarial networks for prostate cancer detection. *CoRR*, abs/1711.10400, 2017.

[8] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016.

[9] Christian Szegedy, Sergey Ioffe, and Vincent Vanhoucke. Inception-v4, inception-resnet and the impact of residual connections on learning. *CoRR*, abs/1602.07261, 2016.

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. pages 1097–1105, 2012.

[11] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *CoRR*, abs/1210.5644, 2012.

[12] Laura Sevilla-Lara, Deqing Sun, Varun Jampani, and Michael J. Black. Optical flow with semantic segmentation and localized layers. *CoRR*, abs/1603.03911, 2016.

[13] M. R. Avendi, Arash Kheradvar, and Hamid Jafarkhani. Automatic segmentation of the right ventricle from cardiac mri using a learning-based approach. *Magnetic resonance in medicine*, 78 6:2439–2448, 2017.

[14] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. Understanding the exploding gradient problem. *CoRR*, abs/1211.5063, 2012.

[15] Hinrich B. Winther, Christian Hundt, Bertil Schmidt, Christoph Czerner, Johann Bauersachs, Frank K. Wacker, and Jens Vogel-Claussen. v-net: Deep learning for generalized biventricular cardiac mass and function parameters. *CoRR*, abs/1706.04397, 2017.

[16] Abdel Aziz Taha and Allan Hanbury. An efficient algorithm for calculating the exact hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.

[17] Nicoló Savioli, Silvia Visentin, Erich Cosmi, Enrico Grisan, Pablo Lamata, and Giovanni Montana. Temporal convolution networks for real-time abdominal fetal aorta analysis with ultrasound. *Springer International Publishing*, pages 148–157, 2018.

[18] Nicoló Savioli, Miguel Silva Vieira, Pablo Lamata, and Giovanni Montana. Automated segmentation on the entire cardiac cycle using a deep learning work - flow. IEEE Xplore:153–158, Oct 2018.

[19] Nicoló Savioli, Giovanni Montana, and Pablo Lamata. V-fcnn: Volumetric fully convolution neural network for automatic atrial segmentation. In *Statistical Atlases and Computational Models of the Heart. Atrial Segmentation and LV Quantification Challenges*, pages 273–281, Cham, 2019. Springer International Publishing.

[20] Nicolo Savioli, Miguel Silva Vieira, Pablo Lamata, and Giovanni Montana. A generative adversarial model for right ventricle segmentation. *arXiv:1810.03969*, Sep 2018.

[21] Lon Bottou. Large-scale machine learning with stochastic gradient descent. 2010.

[22] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, 12:2121–2159, July 2011.

[23] Matthew D. Zeiler. Adadelta: An adaptive learning rate method. *CoRR*, abs/1212.5701, 2012.

[24] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[25] David H. Hubel and Torsten N. Wiesel. Receptive fields and functional architecture of monkey striate cortex. *Journal of Physiology (London)*, 195:215–243, 1968.

[26] Kunihiko Fukushima and Sei Miyake. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. pages 267–285, 1982.

[27] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998.

[28] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Robust object recognition with cortex-like mechanisms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):411–426, March 2007.

[29] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. CNN features off-the-shelf: an astounding baseline for recognition. *CoRR*, abs/1403.6382, 2014.

[30] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

[31] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038, 2014.

[32] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In Geoffrey Gordon, David Dunson, and Miroslav Dudk, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 315–323, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR.

[33] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[34] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[35] Oriol Vinyals, Suman V. Ravuri, and Daniel Povey. Revisiting recurrent neural networks for robust asr. March 2012.

[36] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. Understanding the exploding gradient problem. *CoRR*, abs/1211.5063, 2012.

[37] Sepp Hochreiter and Jrgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[38] KyungHyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *CoRR*, abs/1409.1259, 2014.

[39] Lars M. Mescheder, Sebastian Nowozin, and Andreas Geiger. Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks. *CoRR*, abs/1701.04722, 2017.

[40] John F. Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950.

[41] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. pages 2672–2680, 2014.

[42] Nardelli Giovanni Battista Di Camillo Barbara Grisan Enrico Cosmi Erich Visentin Silvia, Grumolato Francesca. Early origins of adult disease: Low birth weight and vascular remodeling. *Atherosclerosis, Elsevier*, 237:391–399, 2014.

[43] Elisa Veronese, Giacomo Tarroni, Silvia Visentin, Erich Cosmi, Marius George Linguraru, and Enrico Grisan. Estimation of prenatal aorta intima-media thickness from ultrasound examination. *Physics in medicine and biology*, 59(21):6355–71, 2014.

[44] G. Tarroni, S. Visentin, E. Cosmi, and E. Grisan. Fully-automated identification and segmentation of aortic lumen from fetal ultrasound images. pages 153–156, Aug 2015.

[45] Mennatullah Siam, Sepehr Valipour, Martin Jägersand, and Nilanjan Ray. Convolutional gated recurrent networks for video segmentation. *CoRR*, abs/1611.05435, 2016.

[46] Filippo Molinari, Guang Zeng, and Jasjit S. Suri. A state of the art review on intima-media thickness (imt) measurement and wall segmentation techniques for carotid ultrasound. *Computer Methods and Programs in Biomedicine*, 100(3):201–221, 2010.

[47] Loizou C.P. A review of ultrasound common carotid artery image and video segmentation techniques. *Med and Biol Eng and Comp*, 52(12):1073–1093, 2014.

[48] Bejnordi B.E. Setio A.A.A. Ciompi F. Ghafoorian M. van der Laak J.A.W.M. van Ginneken B. Litjens G., Kooi T. A survey on deep learning in medical image analysis. *Med Image Anal*, 42:60–88, 2017.

[49] Hurst R.T. Kendall C.B. Liang J. Shin J.Y., Tajbakhsh N. Automating carotid intima-media thickness video interpretation with convolutional neural networks. *IEEE CVPR Conference*, pages 2526–2535, 2016.

[50] Noble J.A Huang W., Bridge C.P. Automating carotid intima-media thickness video interpretation with convolutional neural networks. *MICCAI 2017, LNCS*, 10434:341–349, 2017.

[51] Ni D. Cheng J.-Z. Qin J. Li S. Heng P.-A. Chen H., Dou Q. Automating carotid intima-media thickness video interpretation with convolutional neural networks. *Automatic fetal ultrasound standard plane detection using knowledge transferred recurrent neural networks.*, pages 2526–2535, 2016.

[52] Fanelli T. Mautone A.J.-Zanardo V Cosmi E., Visentin S. Aortic intima media thickness in fetuses and children with intrauterine growth restriction. *Obs Gyn*, 114:1109–1114, 2009.

[53] Harmer J.A.-Celermajer D.S. Skilton M.R., Evans N. Griffiths K.A. Aortic wall thickness in newborns with intrauterine growth restriction. *Lancet*, 365:1484–6, 2005.

[54] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[55] C. E. Bonferroni. Statistical theory of classes and calculation of probabilities. *Publications of the Royal Higher Institute of Economic and Commercial Sciences of Florence*, 1936.

[56] Olive Jean Dunn. Multiple comparisons among means" journal of the american statistical association. *Polish Journal of Radiology*, 56:52?64, 1961.

[57] Manuel D. Cerqueira, Neil J. Weissman, Vasken Dilsizian, Alice K. Jacobs, Sanjiv Kaul, Warren K. Laskey, Dudley J. Pennell, John A. Rumberger, Thomas Ryan, and Mario S Verani. Standardized myocardial segmentation and nomenclature for tomographic imaging of the heart. *Circulation*, 105(4):539–542, 2002.

[58] Valliappan Raman, Patrick Then, and Annuar Rapaee. A brief study on automated non contrast cardiac mri segmentation by machine vision techniques. *IACSIT International Journal of Engineering and Technology*, 4(5), 2012.

[59] Ebeling C. Barbier, L. Johansson, L. Lind, H. Ahlstrm, and T. Bjerner. The exactness of left ventricular segmentation in cine magnetic resonance imaging and its impact on systolic function values. *Acta Radiologica*, 48(3):285–291, 2007. PMID: 17453498.

[60] Caroline Petitjean and Jean-Nicolas Dacher. A review of segmentation methods in short axis cardiac mr images. *Medical Image Analysis*, 15(2):169–184, 2011.

[61] A. Goshtasby and D. A. Turner. Segmentation of cardiac cine mr images for extraction of right and left ventricular chambers. *IEEE Transactions on Medical Imaging*, 14(1):56–64, Mar 1995.

[62] Myocardial border detection by branch-and-bound dynamic programming in magnetic resonance images. *Computer Methods and Programs in Biomedicine*, 79(1):19 – 29, 2005.

[63] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, Feb 2001.

[64] Tuan Anh Ngo and Gustavo Carneiro. Left ventricle segmentation from cardiac MRI combining level set methods with deep belief networks. *IEEE International Conference on Image Processing, ICIP 2013, Melbourne, Australia, September 15-18, 2013*, pages 695–699, 2013.

[65] Tuan Anh Ngo and Gustavo Carneiro. Fully automated non-rigid segmentation with distance regularized level set evolution initialized and constrained by deep-structured inference. *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 3118–3125, 2014.

[66] M. R. Avendi, Arash Kheradvar, and Hamid Jafarkhani. A combined deep-learning and deformable-model approach to fully automatic segmentation of the left ventricle in cardiac MRI. *CoRR*, abs/1512.07951, 2015.

[67] Phi Vu Tran. A fully convolutional neural network for cardiac segmentation in short-axis MRI. *CoRR*, abs/1604.00494, 2016.

[68] Rudra P. K. Poudel, Pablo Lamata, and Giovanni Montana. Recurrent fully convolutional neural networks for multi-slice MRI cardiac segmentation. *Reconstruction, Segmentation, and Analysis of Medical Images-First International Workshops, RAMBO 2016 and HVSMR 2016, Held in Conjunction with MICCAI 2016, Athens, Greece, October 17, 2016, Revised Selected Papers*, pages 83–94, 2016.

[69] Jianxu Chen, Lin Yang, Yizhe Zhang, Mark S. Alber, and Danny Z. Chen. Combining fully convolutional and recurrent neural networks for 3d biomedical image segmentation. *CoRR*, abs/1609.01006, 2016.

[70] Johan Montagnat and Herv Delingette. 4d deformable models with temporal constraints: application to 4d cardiac image segmentation. *Medical Image Analysis*, 9(1):87 – 100, 2005.

[71] M. Lynch, O. Ghita, and P.F. Whelan. Left-ventricle myocardium segmentation using a coupled level-set with a priori knowledge. *Computerized Medical Imaging and Graphics*, 30(4):255 – 262, 2006. Medical Imaging and Graphics in SIBGRAPI/SIACG.

[72] A. Elkington-R. Mohiaddin D. Rueckert M. Lorenzo-Valds, G. Sanchez-Ortiz. Segmentation of 4d cardiac mr images using a probabilistic atlas and the em algorithm. *Med. Image Anal*, page 255?265, 2004.

[73] Raghavendra Chandrashekara, Anil Rao, Gerardo Ivar Sanchez-Ortiz, Raad H. Mohiaddin, and Daniel Rueckert. Construction of a statistical model for cardiac motion analysis using nonrigid image registration. pages 599–610, 2003.

[74] O. Gerard, A. C. Billon, J. M. Rouet, M. Jacob, M. Fradkin, and C. Allouche. Efficient model-based quantification of left ventricular function in 3-d echocardiography. *IEEE Transactions on Medical Imaging*, 21(9):1059–1068, Sept 2002.

[75] Nikos. Paragios. Hybrid optical flow and segmentation technique for lv motion detection. *IEEE transactions on medical imaging*, pages 773–776, 2003.

[76] Xin Yang, Cheng Bian, Lequan Yu, Dong Ni, editor="Pop Mihaela Heng, Pheng-Ann", Maxime Sermesant, Pierre-Marc Jodoin, Alain Lalande, Xiahai Zhuang, Guang Yang, Alistair Young, and Olivier Bernard. Class-balanced deep neural network for automatic ventricular structure segmentation. In *Statistical Atlases and Computational Models of the Heart. ACDC and MMWHS Challenges*, pages 152–160, Cham, 2018. Springer International Publishing.

[77] Christian Rupprecht, Elizabeth Huaroc, Maximilian Baust, and Nassir Navab. Deep active contours. *CoRR*, abs/1607.05074, 2016.

[78] Jay Patravali, Shubham Jain, and Sasank Chilamkurthy. 2d-3d fully convolutional neural networks for cardiac MR segmentation. *CoRR*, abs/1707.09813, 2017.

[79] Fabian Isensee, Paul Jaeger, Peter M. Full, Ivo Wolf, Sandy Engelhardt, and Klaus H. Maier-Hein. Automatic cardiac disease assessment on cine-mri via time-series segmentation and domain specific features. *CoRR*, abs/1707.00587, 2017.

[80] R. Mehta and J. Sivaswamy. M-net: A convolutional neural network for deep brain structure segmentation. pages 437–440, April 2017.

[81] Mahendra Khened, Varghese Alex, editor="Pop Mihaela Krishnamurthi, Ganapathy", Maxime Sermesant, Pierre-Marc Jodoin, Alain Lalande, Xiahai Zhuang, Guang Yang, Alistair Young, and Olivier Bernard. Densely connected fully convolutional network for short-axis cardiac cine mr image segmentation and heart diagnosis using random forest. 2018.

[82] Clément Zotti, Zhiming Luo, Alain Lalande, Olivier Humbert, and Pierre-Marc Jodoin. Novel deep convolution neural network applied to MRI cardiac segmentation. *CoRR*, abs/1705.08943, 2017.

[83] Valdes AM Spector TD Moayyeri A, Hammond CJ. Cohort profile: Twinsuk and healthy ageing twin study. *Int J Epidemiol.*, 42(1):76–85, 2013.

[84] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, abs/1312.6229, 2013.

[85] Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555, 2014.

[86] Laura Sevilla-Lara, Deqing Sun, Varun Jampani, and Michael J. Black. Optical flow with semantic segmentation and localized layers. *CoRR*, abs/1603.03911, 2016.

[87] Rudra P. K. Poudel, Pablo Lamata, and Giovanni Montana. Recurrent fully convolutional neural networks for multi-slice MRI cardiac segmentation. *Reconstruction, Segmentation, and Analysis of Medical Images-First International Workshops, RAMBO 2016 and HVSMR 2016, Held in Conjunction with MICCAI 2016, Athens, Greece, October 17, 2016, Revised Selected Papers*, pages 83–94, 2016.

[88] Eric N. Prystowsky, D. Woodrow Benson, Valentin Fuster, Robert G. Hart, G. Neal Kay, Robert J. Myerburg, Gerald V. Naccarelli, and D. George Wyse. Management of patients with atrial fibrillation. *Circulation*, 93(6):1262–1277, 1996.

[89] Marta Varela, Felipe Bisbal, Ernesto Zacur, Antonio Berruezo, Oleg V. Aslanidi, Lluis Mont, and Pablo Lamata. *Frontiers in Physiology*, 8:68, 2017.

[90] Raymond J. Kim, Edwin Wu, Allen Rafael, Enn-Ling Chen, Michele A. Parker, Orlando Simonetti, Francis J. Klocke, Robert O. Bonow, and Robert M. Judd. The use of contrast-enhanced magnetic resonance imaging to identify reversible myocardial dysfunction. *New England Journal of Medicine*, 343(20):1445–1453, 2000. PMID: 11078769.

[91] Patrick M. Boyle, Sohail Zahid, and Natalia A. Trayanova. Towards personalized computational modelling of the fibrotic substrate for atrial arrhythmia. *EP Europace*, 18:9, 2016.

[92] Tao Qian, Ipek Esra Gucuk, Shahzad Rahil, Berendsen Floris F., Nazarian Saman, and van der Geest Rob J. Fully automatic segmentation of left atrium and pulmonary veins in late gadolinium-enhanced mri: Towards objective atrial scar assessment. *Journal of Magnetic Resonance Imaging*, 44(2):346–354, 2016.

[93] Aliasghar Mortazi, Rashed Karim, Kawal S. Rhode, Jeremy Burt, and Ulas Bagci. Cardiacnet: Segmentation of left atrium and proximal pulmonary veins from MRI using multi-view CNN. *CoRR*, abs/1705.06333, 2017.

[94] Zhiqiang Hu Aimin Hao Qing Xia, Yuxin Yao. Automatic 3D Atrial Segmentation from GE-MRIs using Volumetric Fully Convolutional Networks. 16 2018.

[95] V Abdi S Engelhardt CJ Preetha, S Haridasan. Segmentation of the Left Atrium from 3D Gd-Enhanced MRI With Convolutional Neural Networks. 16 2018.

[96] J Fernandez E Fok, J Zhao. Ensemble of convolutional neural networks for heart segmentation. 16 2018.

[97] A Maier S Vesal, N Ravikumar. Dilated Convolution in Neural Networks for Left Atrial Segmentation in 3D Late Gadolinium Enhanced-MRI. 16 2018.

[98] Z Wang H Delingette-X Pennec P Jas H Cochet M Sermesant S Jia, A Despinasse. Automatically Segmenting the Left Atrium from Cardiac Images Using Successive 3D U-Nets. 16 2018.

[99] Cong Yan Kuanquan Wang Yashu Liu, Yangyang Dai. Deep Learning Based Method for Left Atrial Segmentation in GE-MRI. 16 2018.

[100] L Esposito A Andalo-C Fabbri C Corsi D Borra, A Masci. A semantic-wise convolutional neural network approach for 3D left atrium segmentation from LGE-MRI. 16 2018.

[101] O Razeghi S Niederer-J Pluim K Rhode R Karim C Vente, M Veta. Convolutional Neural Networks for Segmentation of the Left Atrium from Gadolinium-Enhancement MR Images. 16 2018.

[102] S Zheng J Ma-YA Liu R Nezafat PA Heng Y Zheng C Bian, X Yang. Pyramid Network with online Hard Example Mining for Accurate Left Atrium Segmentation. 16 2018.

[103] D Rueckert C Chen, W Bai. Multi-Task Learning for Left Atrial Segmentation on GE-MRI. 16 2018.

[104] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[105] Y Khoudli Y Xu-J Lacotte T Graud E Puybareau, Z Zhou. Left Atrial Segmentation in a Few Seconds Using Fully Convolutional Network and Transfer Learning. 16 2018.

[106] Y Wang X Wang-R Nezafat D Ni PA Heng X Yang, N Wang. Combating Uncertainty with Novel Losses for Automatic Atrium Segmentation. 16 2018.

[107] R Whitaker J Cates-N Marrouche S Elhabian T Sodergren, R Bhalodia. Mixture Modeling of Global Shape Priors and Autoencoding Local Intensity Priors for Left Atrium Segmentation. 16 2018.

[108] Z. Xiong, V. V. Fedorov, X. Fu, E. Cheng, R. Macleod, and J. Zhao. Fully automatic left atrium segmentation from late gadolinium enhanced magnetic resonance imaging using a dual fully convolutional neural network. *IEEE Transactions on Medical Imaging*, pages 1–1, 2018.

[109] X Liao W Si-Y Sun Q Wang PA Heng C Li, Q Tong. Attention based hierarchical aggregation network for 3D Left atrial segmentation. 16 2018.

[110] R Geest Q Tao M Qiao, Y Wang. Fully Automated Left Atrium Cavity Segmentation from 3D GE-MRI by Multi-Atlas Selection and Registration. 16 2018.

[111] G Sanroma L Li-L Xu C Butakoff O Camara M Nunez, X Zhuang. Left atrial segmentation combining multi-atlas whole heart labeling and shape-based atlas selection. 16 2018.

[112] Marc-Michel Rohé, Maxime Sermesant, and Xavier Pennec. Automatic Multi-Atlas Segmentation of Myocardium with SVF-Net. September 2017.

[113] Fabian Isensee, Paul Jaeger, Peter M. Full, Ivo Wolf, Sandy Engelhardt, and Klaus H. Maier-Hein. Automatic cardiac disease assessment on cine-mri via time-series segmentation and domain specific features. *CoRR*, abs/1707.00587, 2017.

[114] H. Kaur and J. Rani. Mri brain image enhancement using histogram equalization techniques. pages 770–773, March 2016.

[115] Paolo Cignoni, Claudio Rocchini, and Roberto Scopigno. Metro: Measuring error on simplified surfaces. 1996.

[116] Jianxu Chen, Lin Yang, Yizhe Zhang, Mark S. Alber, and Danny Z. Chen. Combining fully convolutional and recurrent neural networks for 3d biomedical image segmentation. *CoRR*, abs/1609.01006, 2016.

[117] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *CoRR*, abs/1606.02147, 2016.

[118] Z. Xiong, V. V. Fedorov, X. Fu, E. Cheng, R. Macleod, and J. Zhao. Fully automatic left atrium segmentation from late gadolinium enhanced magnetic resonance imaging using a dual fully convolutional neural network. *IEEE Transactions on Medical Imaging*, pages 1–1, 2018.

[119] Andre Xian Ming Chang, Berin Martini, and Eugenio Culurciello. Recurrent neural networks hardware implementation on FPGA. *CoRR*, abs/1511.05552, 2015.

[120] Catherine Kreatsoulas and Sonia S. Anand. The impact of social determinants on cardiovascular disease. *Canadian Journal of Cardiology*, 26:8–13, 2010.

[121] Hundley WG. Stacey RB. Magnetic resonance (cmr) and computed tomography (cct) in facilitating heart failure management. current treatment options in cardiovascular medicine. *doi:10.1007/s11936-013-0253-6.*, 2013;15(4):373-386.

[122] Jianxu Chen, Lin Yang, Yizhe Zhang, Mark S. Alber, and Danny Z. Chen. Combining fully convolutional and recurrent neural networks for 3d biomedical image segmentation. *CoRR*, abs/1609.01006, 2016.

[123] P A Davlouros, K Niwa, G Webb, and M A Gatzoulis. The right ventricle in congenital heart disease. *Heart*, 92(suppl 1):i27–i38, 2006.

[124] Right ventricle segmentation from cardiac mri: A collation study. *Medical Image Analysis*, 19(1):187 – 202, 2015.

[125] Maria Lorenzo-Valds, Gerardo I. Sanchez-Ortiz, Andrew G. Elkington, Raad H. Mohiaddin, and Daniel Rueckert. Segmentation of 4d cardiac mr images using a probabilistic atlas and the em algorithm. *Medical Image Analysis*, 8(3):255 – 265, 2004. Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003.

[126] S. C. Mitchell, B. P. F. Lelieveldt, R. J. van der Geest, H. G. Bosch, J. H. C. Reiver, and M. Sonka. Multistage hybrid active appearance model matching: segmentation of left and right ventricles in cardiac mr images. *IEEE Transactions on Medical Imaging*, 20(5):415–423, May 2001.

[127] Hamilton M Bucciarelli-Ducci C. Moolan-Feroze O, Mirmehdi M. Segmentation of the right ventricle using diffusion maps and markov random fields. *In Medical image computing and computer-assisted intervention?MICCAI 2014.*, New York: Springer; 2014. p 682?689.

[128] Dacher J-N Ruan S. Grosgeorge D, Petitjean C. Graph cut segmenta- tion with a statistical shape model in cardiac mri. comput vis image. *Comput Vis Image Underst*, 2013.

[129] Santos A-Ledesma-Carbayo MJ. Maier OMO, Jimenez D. Segmenta- tion of rv in 4d cardiac mr volumes using region-merging graph cuts. *Comput. Cardiology*, 2012.

[130] D. Mahapatra and J. M. Buhmann. Automatic cardiac rv segmentation using semantic information with graph cuts. pages 1106–1109, April 2013.

[131] Phi Vu Tran. A fully convolutional neural network for cardiac segmentation in short-axis mri. *CoRR*, abs/1604.00494, 2016.

[132] G. Luo, R. An, K. Wang, S. Dong, and H. Zhang. A deep learning network for right ventricle segmentation in short-axis mri. pages 485–488, Sept 2016.

[133] Jesse Lieman-Sifry, Matthieu Lê, Felix Lau, Sean Sall, and Daniel Golden. Fastventricle: Cardiac segmentation with enet. *CoRR*, abs/1704.04296, 2017.

[134] MR Avendi, Arash Kheradvar, and Hamid Jafarkhani. Fully automatic segmentation of heart chambers in cardiac mri using deep learning. *Journal of Cardiovascular Magnetic Resonance*, 18(1):P351, Jan 2016.

[135] Fully automatic roi extraction and edge-based segmentation of radius and ulna bones from hand radiographs. *Biocybernetics and Biomedical Engineering*, 37(4):718 – 732, 2017.

[136] Zehan Wang, Lijun Zhu, and Jiandong Qi. Roi extraction in dermatosis images using a method of chan-vese segmentation based on saliency detection. 274:197–203, 01 2014.

[137] Seokwon Yeom. Multi-level segmentation of infrared images with region of interest extraction. *International Journal of Fuzzy Logic and Intelligent Systems*, 16(4):246–253, 2016.

[138] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017.

[139] Marvin Teichmann, Michael Weber, J. Marius Zöllner, Roberto Cipolla, and Raquel Urtasun. Multinet: Real-time joint semantic reasoning for autonomous driving. *CoRR*, abs/1612.07695, 2016.

[140] Nikita Dvornik, Konstantin Shmelkov, Julien Mairal, and Cordelia Schmid. Blitznet: A real-time deep network for scene understanding. *CoRR*, abs/1708.02813, 2017.

[141] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. *CoRR*, abs/1606.07536, 2016.

[142] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. *CoRR*, abs/1703.00848, 2017.

[143] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. *CoRR*, abs/1604.07379, 2016.

[144] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. *CoRR*, abs/1611.07004, 2016.

[145] Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555, 2014.

[146] Mennatullah Siam, Sepehr Valipour, Martin Jäampersand, and Convolutional Gated Recurrent Networks for Video Segmentation Nilanjan Ray. Convolutional gated recurrent networks for video segmentation. *CoRR, abs/1611.05435, 2016, http://arxiv.org/abs/1611.05435*.

[147] Jordan Ringenberg, Makarand Deo, Vijay Devabhaktuni, Omer Berenfeld, Pamela Boyers, and Jeffrey Gold. Fast, accurate, and fully automatic segmentation of the right ventricle in short-axis cardiac mri. *Computerized Medical Imaging and Graphics*, 38(3):190 – 201, 2014.

[148] Modat M Ourselin S Zuluaga MA, Cardoso MJ. Multi-atlas propaga- tion whole heart segmentation from mri and cta using a local nor- malized correlation coefficient criterion. in: Functional imaging and modeling of the heart. *New York: Springer; 2013*, 174?181., 2013.

[149] Chen H Wang C, Peng C. A simple and fully automatic right ventri- cle segmentation method for 4-dimensional cardiac mr images. *a MICCAI Segmentation Challenge*, 2012.

[150] Erus G Davatzikos C u Y, Doshi J. Multi-atlas segmentation of the cardiac mr right ventricle. proceedings of 3d cardiovascular imaging: a miccai segmentation challeng. *Nice, France*, 2012.

[151] Punithakumar K Goela A Rajchl M Peters TM Ayed IB Nambakhsh CM, Yuan J. Left ventricle segmentation in mri via convex relaxed distribution matching. *Med Image Anal 2013*, 17:1010?1024, 2013.

[152] O Regan DP Tong T Wang H Jamil-Copley S Peters NS Rueckert D. Bai W, Shi W. A probabilistic patch-based label fusion model for multi-atlas segmentation with registration refinement:application to cardiac mr images. *IEEE Trans Med Imaging 2013*, 32:1302?1315, 2013.

[153] Nicoló Savioli, Miguel Silva Vieira, Pablo Lamata, and Giovanni Montana. Automated segmentation on the entire cardiac cycle using a deep learning work-flow. *IEEExplore*, 2018.

[154] Nicoló Savioli, Giovanni Montana, and Pablo Lamata. V-fcnn: Volumetric fully convolution neural network for automatic atrial segmentation. *MICCAI 2018 Atrial Segmentation Challenge, Springer*, 2018.

[155] Park Hyunjin Kim Jonghoon, Hong Jisu. Prospects of deep learning for medical imaging. *Precis Future Med*, 2(2):37–52, 2018.

[156] Pranav Rajpurkar, Awni Y. Hannun, Masoumeh Haghpanahi, Codie Bourn, and Andrew Y. Ng. Cardiologist-level arrhythmia detection with convolutional neural networks. *CoRR*, abs/1707.01836, 2017.

[157] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. *CoRR*, abs/1711.11248, 2017.

[158] Ghalib A. Bello, Timothy J. W. Dawes, Jinming Duan, Carlo Biffi, Antonio de Marvao, Luke S. G. E. Howard, J. Simon R. Gibbs, Martin R. Wilkins, Stuart A. Cook, Daniel Rueckert, and Declan P. O'Regan. Deep learning cardiac motion analysis for human survival prediction. *CoRR*, abs/1810.03382, 2018.

[159] D.P. O'Regan. Putting machine learning into motion: applications in cardiovascular imaging. *Clinical Radiology*, 2019.