IMA205 Project: Cardiac Pathology Prediction

Pegah KHAYATAN (crocodile's aorta)

Contents

1 Introduction

The goal of the challenge is to diagnose four cardiac pathologies using cardiac magnetic resonance imaging (CMRI). This is a non-invasive method (no biopsy is needed in order to make the diagnostic), and can be used on a broader range of patients in less time and resources.
These pathologies are myocardial infarction, dilated cardiomyopathy, hypertrophic cardiomyopathy, and abnormal right ventricle. The task is to classify patients into five categories (the mentioned pathologies and a normal patient); the provided information for the training set is MRI images and their segmentation to four regions (background, right ventricle cavity, myocardium, left ventricle cavity) at end diastole and end systole, height and weight of patients, and their diagnostic. The test set consists of MRI images where the left ventricle is labeled as background, and the diagnostic is missing. Hence, the task can be decomposed into two parts: (a) segmentation of test images (b) classification of test patients using their segmented images and metadata (height and weight).

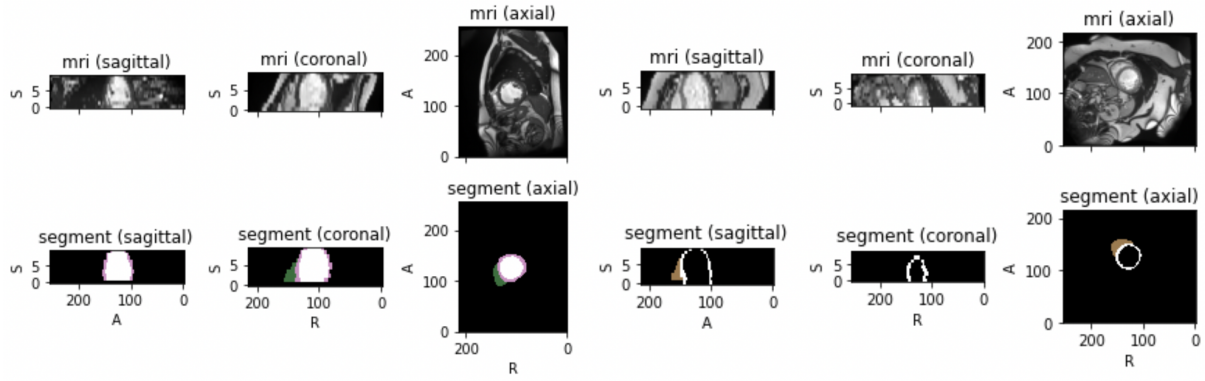

Figure 1: On the left is an MRI image with its segmentation map from the training set (4 segmentation classes), and on the right is an MRI image with its segmentation map from the test set (3 segmentation classes). MRI images can be viewed in different planes: sagittal, coronal, and axial.

2 Data Analysis

2.1 Provided Data

MRI (Magnetic Resonance Imaging) is a medical imaging technique that uses a strong magnetic field and radio waves to produce detailed images of the body's internal structures. MRI images can be acquired in different planes (sagittal, coronal, and axial), and they can provide detailed information about the soft tissues and organs in the body.

Segmentation of MRI images consists of annotating different regions depending on the application. The segmentation information provided at the end of the process can take different forms, *cf.* a binary mask, where each pixel is assigned a label indicating whether it belongs to a particular region or not, or a set of contours.

2.2 Preprocessing

With the help of existing libraries reading these segmentations boils down to a simple script. However, just reading these segmentations is not enough; we will need to apply a series of elementary preprocessing, before constructing data loaders. We chose to resample all volumes to $1.3 \times 1.3 \times 10$ mm per voxel. Cropping or padding (to $300 \times 300 \times 10$) is applied to make all the samples of the same size. The last step of preprocessing is normalization, where the sample is subtracted by its mean and divided by its standard deviation. These operations are implemented using *torch io* library, when creating a dataset *cf.* Figure 2.

2.3 Justifying the preprocessing parameters

The choice of resampling and cropping/padding values is not random, and is based on the values suggested by related papers [Bau+18; Ise+18; Wol+18]; the same range of values can also be concluded from initial statistics of the shape and spacing in the dataset. This is summarized in the Figure 3, where histograms of spacing and shape in mri images are illustrated for the x, y, and z directions. The most repeated spacing value is chosen for resampling and the highest existing shape value is chosen for padding/cropping (except for the z direction, where the most repeated spacing value is chosen in resampling.).

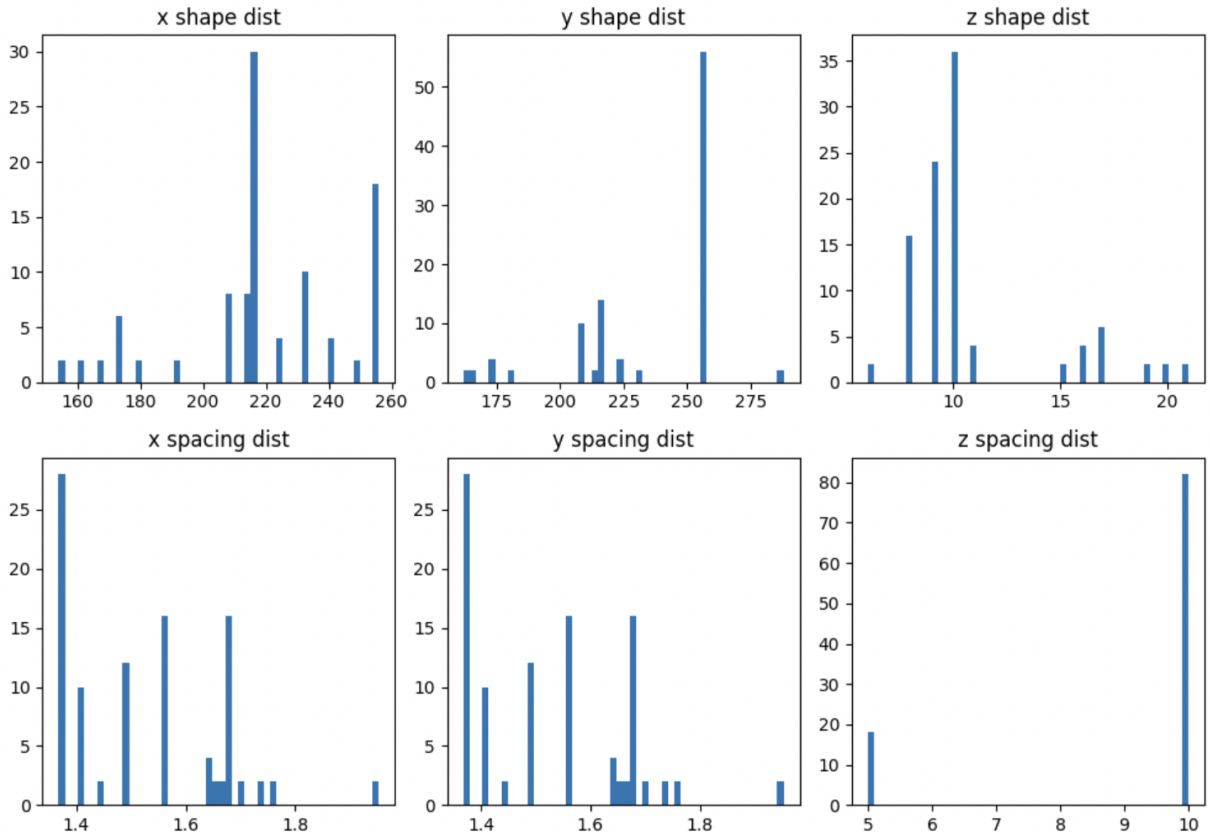

Figure 2: histogram of spacing and shape of mri images, for different axes. The most repeated values is chosen for resampling and the highest existing value is chosen for padding/cropping.

3 Segmentation

Our task of segmentation is easier than those mentioned in related papers: segmentation of the test set consists of three labels, background, right ventricle cavity, myocardium, where the left ventricle is also labeled as background. However, finding the region corresponding to this wrongly labeled region is straight forward, since the left ventricle is surrounded by myocardium.
Hence, our task of segmentation consists of labelling background pixels surrounded by myocardium as left ventricle.

We discuss two different approaches in the following: (a) simply implementing the strategy explained above, where no machine learning method is used. (b) designing a deep neural network to learn the above task. The two methods are discussed below.

3.1 Classic Approach

The approach consists of finding the area labelled as background and surrounded by myocardium at every z-level. For this purpose we convert all the labels to myocardium or background (converting right ventricle to background).
Considering a certain z-level (and there are 10 of them), it is a 2D plane, where possibly there is a part of the myocardium and the left ventricle. The process is decomposed into:

1. consider a bar of pixels with a certain y value at a time.
2. fix the current x value to zero, at the considered z and y level. These values determine the current pixel we are examining.
3. augment the current x value while the label of the current pixel is background. *
4. having reached a pixel with myocardium label, we augment the current x value while the label of the current pixel is myocardium.
5. Having reached a pixel with background label again, we store its value as *start of left ventricle*.
6. we augment the current x value while the label of the current pixel is background. *
7. having reached again a pixel with myocardium label, we store the previous current x value as the *end of left ventricle*.

In steps annotated with *, we should be careful not to go outside the boundary of the image, since there are fixed-yz bars of the image with no myocardium or left ventricle, or only with left ventricle.

The steps discussed above are illustrated in Figure 3.

3.2 Deep Learning Approach

A deep learning approach has also been tried out. We know that this approach will definitely not have a superior performance than the technique discussed above (since regions other than the left ventricle

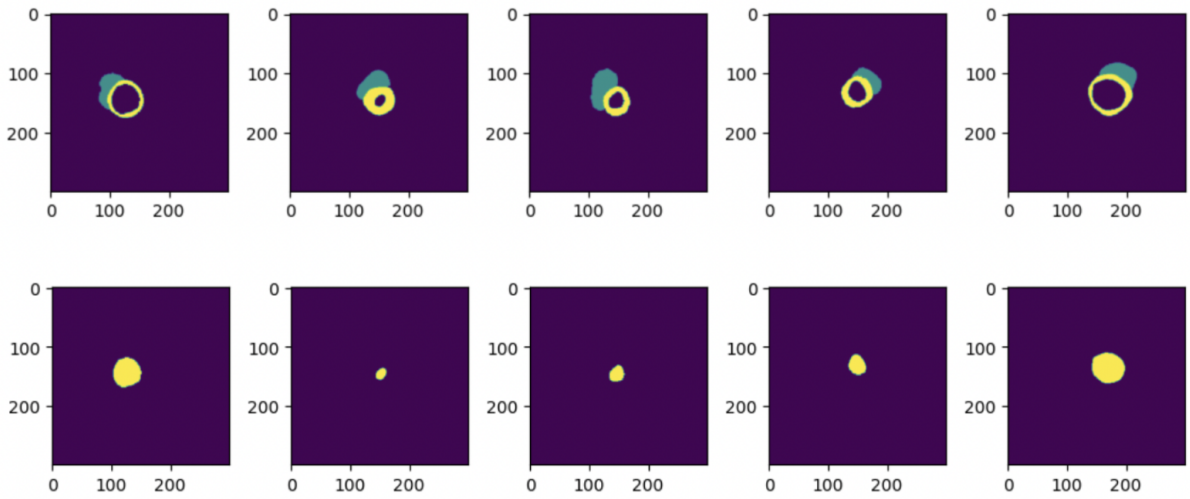

Figure 3: On the first row are 5 of test images where the left ventricle region is not labelled correctly. On the second row are the result of first (classic) approach to find the left ventricle. Both images are from the third z-level.

are hand labelled, and determining the left ventricle is a deterministic process), but the task of segmentation is viewed here as a stand-alone problem.

The segmentation task is defined the same as before: having access to the segmentation where left ventricle and background are labelled as background, and right ventricle and myocardium are correctly labelled, the goal is to find the pixels corresponding to the left ventricle. The input is considered only as the mask of myocardium, and the target is the mask of left ventricle.

The model is an UNet, consisting of a context aggregating pathway followed by a localization pathway. Both are interconnected at various scales to allow for recombination of abstract context features with the corresponding local information. A schema of the model is illustrated in Figure 4.

The optimizer is Adam, with an initial learning rate of 0.005, and the learning rate is reduced to its 98% at the end of each epoch. Batch size is set to 4, and each epoch consists of 200 batches.

We train on the combination of ES and ED segmentation maps, *cf.* we do not consider ED and ES samples separately from one another, and the model should be able to do the segmentation in both cases.

The loss function along with some results are shown in Figure 5.

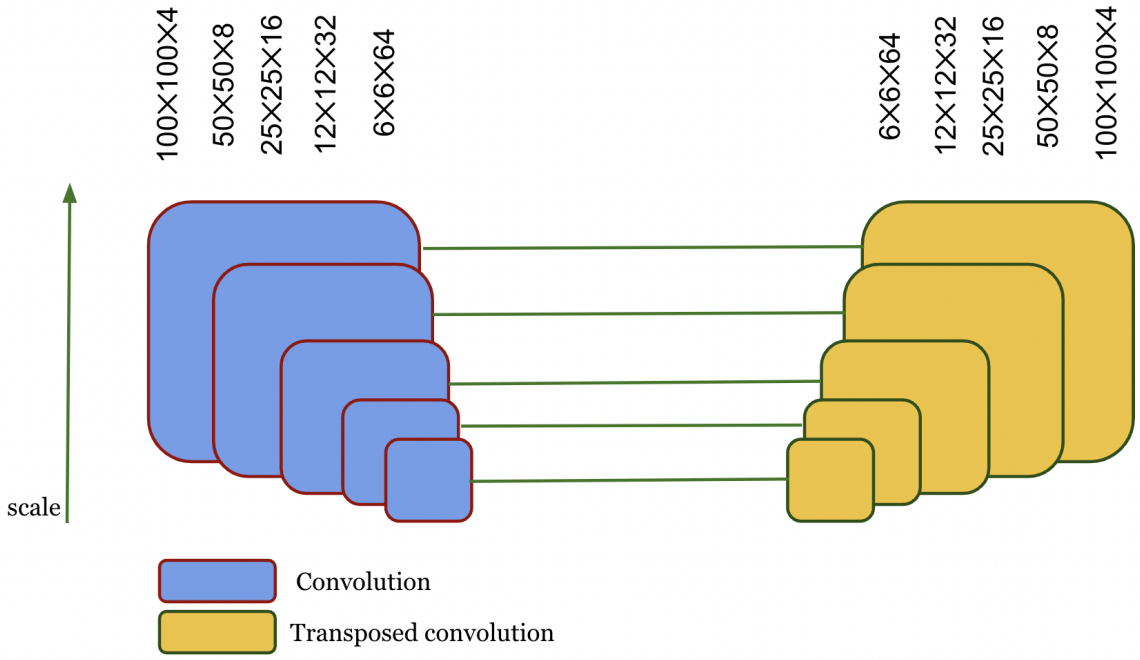

Figure 4: The implemented segmentation model. The dimension in different layers is indicated on top.

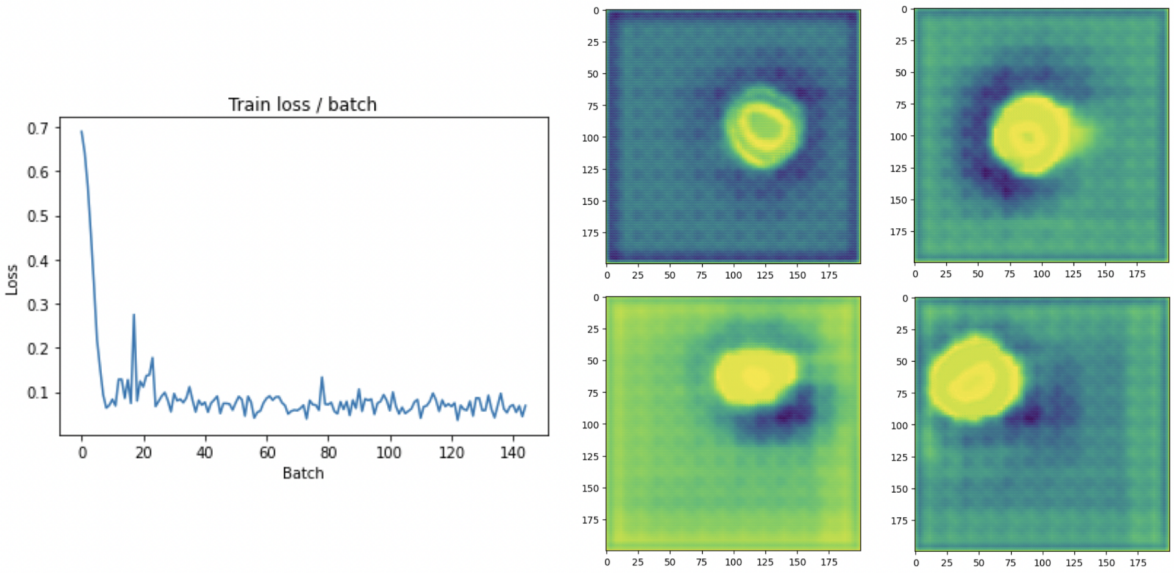

Figure 5: On the left is the curve of training loss. On the right are some results from even early epochs of the model.

4 Classification

In this section, the chosen features are introduced, and also the strategy to approximate some features is explained in extent. In the last subsection, we introduce the machine learning models employed for the task, and discuss their hyperparameters.

4.1 Feature Extraction

The feature extraction has also been inspired by [Bau+18; Ise+18; Wol+18]. There are two types of feature in general: features that concern only one time step in the heart cycle (systole or diastole), or features that relate the two time steps together by comparing some statistics between them.

For the first category, we calculate the volume of the left ventricle, right ventricle, and the myocardium, the ratio of the volume of right to left ventricle, and the ratio of the volume of myocardium to left ventricle. On top of these features, we also calculate max, min, mean, and standard deviation of thickness for different z-levels (features are aggregated for different z values). From the approximated thickness, we also calculate the circumference and the circularity of myocardium. The method used to approximate these features is explained in the next subsection.

For the second category, we calculate the volume change of the left and right ventricle, and the myocardium, the ejection fraction of the right and left ventricle (the ejection fraction is defined as EF = ((EDV - ESV) / EDV) x 100, where EDV corresponds to the volume at the end diastole and ESV to the volume at the end of systole).

We also add the height and weight of the patients to the list of features.

We use the above features to train our model, which we introduce in the coming sections.

4.2 Calculating Features

Here we explain the calculation of myocardium thickness-related features, and the circumference and circularity of myocardium.
We calculate the features for each z-level, and aggregate them over z-levels.

The process for finding the thickness of myocardium consists of the following steps:

1. we first calculate an approximation of the myocardium center by taking the mean of indices where their corresponding pixel is labelled as myocardium.
2. we then find the minimum distance of this center to the points on the myocardium.
3. we trace rays in 30 (can be different) different angles from the center; the length of each ray will be decided in the next step.
4. the length of the ray in a certain angle is initialized with the minimum distance found in the second step. This length is increased incrementally, till it touches a myocardium pixel. The

step size should be small in order not to jump over the myocardium pixels. The found length is considered as the *starting* ray length.

5. we continue increasing the length of the ray till reaching a background pixel. The found length is considered as the *end* ray length.

6. difference between the *starting* and the *end* ray lengths is considered as the thickness of myocardium along the specific angle.

We now have calculated the thickness of myocardium in different angles. From this set of thicknesses, we can infer their mean, std, min, and max statistics. These statistics are then aggregated for different z-levels.

The circumference of myocardium is estimated as the number of all the myocardium pixels divided by the mean thickness calculated above.

The formula for calculating circularity using perimeter (P) and area (A) is: $C = \frac{4\pi A}{P^2}$ where C is the circularity, A is the area of the shape, and P is the perimeter of the shape. The perimeter is estimated in the last step, and the area is the sum of number of pixels labelled as myocardium and left ventricle.

Circumference and circularity are also aggregated for different z-levels.

4.3 Classification Models

We use a combination of an MLP and random forest ensemble as a classification strategy.

Each MLP model has five linear layers with batch normalization and LeakyReLU activation and uses Gaussian noise to add regularization. Overall we train 30 models, and each is trained on 75% of the training set; each model is trained for 300 epochs, and the best epoch is chosen based on the accuracy of the model on the rest of the training set. One epoch consists of 250 batches with a batch size of 20. Adam optimizer is used with an initial learning rate of 0.0005. The model and its training are partially inspired by [Ise+18].

The performance of stand-alone MLP ensemble on the test set is 93%.

The other classification tool is random forest, for which we tuned the random state on the training set, using a seven-fold cross validation. It resulted in choosing the parameter .... as random state. Other parameters were taken from the cited papers, most importantly the number of estimators, which was set to 1000.

The performance of stand-alone random forest with tuned parameters on the test set is 93%.

To combine the classification power of the two above methods, we use a weighting strategy: we use three random forests with different random states (0,1,2), and an ensemble of 30 MLPs. For each test sample, we compute a score which is composed of the sum of outputs of the ensemble models, and the

one-hot encoded score of each random forest multiplied by a fixed number (in the code it's seven, which is largely based on trial and error). The performance of this latter is not different from those above, but it may be more robust when the parameters are finely tuned.

Tuning the number of estimators, min samples leaf, and min samples split on the training set did not generalize well to the test set. The reason can be the low number of samples, which leads to overfitting when fine-tuning (even with cross-validation).

References

[Bau+18] Christian F Baumgartner et al. “An exploration of 2D and 3D deep learning techniques for cardiac MR image segmentation”. In: *Statistical Atlases and Computational Models of the Heart. ACDC and MMWHS Challenges: 8th International Workshop, STACOM 2017, Held in Conjunction with MICCAI 2017, Quebec City, Canada, September 10-14, 2017, Revised Selected Papers 8*. Springer. 2018, pp. 111–119.

[Ise+18] Fabian Isensee et al. “Automatic cardiac disease assessment on cine-MRI via time-series segmentation and domain specific features”. In: *Statistical Atlases and Computational Models of the Heart. ACDC and MMWHS Challenges: 8th International Workshop, STACOM 2017, Held in Conjunction with MICCAI 2017, Quebec City, Canada, September 10-14, 2017, Revised Selected Papers 8*. Springer. 2018, pp. 120–129.

[Wol+18] Jelmer M Wolterink et al. “Automatic segmentation and disease classification using cardiac cine MR images”. In: *Statistical Atlases and Computational Models of the Heart. ACDC and MMWHS Challenges: 8th International Workshop, STACOM 2017, Held in Conjunction with MICCAI 2017, Quebec City, Canada, September 10-14, 2017, Revised Selected Papers 8*. Springer. 2018, pp. 101–110.