IMA205 Challenge 2023 Cardiac Pathology Prediction Report

albert adwit-Yiming Wei
École Polytechnique, Institut Polytechnique de Paris
Palaiseau
{albert adwit-Yiming Wei}yiming.wei@polytechnique.edu

Abstract

Cardiac magnetic resonance imaging (CMRI) plays a critical role in diagnosing cardiac pathologies, enabling early intervention and effective management of patients. In this study, I developed a pipeline for automatic cardiac diagnosis using CMRI images and machine learning techniques. My approach involved two main steps: segmentation of the left ventricle cavity and extraction of relevant cardiac parameters for classification.

I first addressed the issue of partial segmentation in the test set by employing a technique to classify areas within the myocardium as left ventricle cavity. This method was based on the observation that the left ventricle cavity is enclosed by the myocardium. Next, I calculated cardiac parameters such as ejection fraction and ventricular volume from the segmentation masks at end-systole and diastole.

These features were then used to train two machine learning classifiers: random forest and XGBoost. I achieved promising results, with the random forest classifier reaching an accuracy of 89% and the XGBoost classifier attaining an accuracy of 85% for automatic diagnosis of cardiac diseases. My approach demonstrates the potential of combining CMRI data and machine learning techniques for early detection and improved management of cardiac pathologies.

***K*eywords** Cardiac MRI · Automatic cardiac diagnosis

1 Introduction

Cardiac pathologies, if left undetected, can lead to severe complications such as heart failure and sudden cardiac arrest. Early identification and diagnosis are crucial for effective patient management and therapeutic decisions. In recent years, non-invasive computer-aided diagnosis (CAD) techniques have been developed to analyze cardiac magnetic resonance imaging (CMRI) data for detecting various cardiac conditions.

Several researchers have proposed methods to segment the left ventricle and extract relevant features from CMRI data. One such method is the Automatic Left Ventricle Segmentation Using Iterative Thresholding and an Active Contour Model With Adaptation on Short-Axis Cardiac MRI (ITHACA) algorithm, which showed significant improvement over the commercial MASS Analysis software in defining myocardial borders. Another study presented a fully automated deformable model technique for myocardium segmentation in 3D MRI, which demonstrated accuracy and robustness[1]. A novel approach for automatic segmentation of the left ventricle cavity and myocardium in MRI data was developed using a diffusion-based filter followed by an unsupervised clustering technique[2]. Furthermore, an algorithm for left ventricle segmentation using active contour methods with gradient vector flow field forces in short-axis MRI was proposed, demonstrating accuracy and robustness[3]. Layered spatio-temporal forests have also been employed for left ventricle segmentation from 4D cardiac MR datasets, as well as regression forests for direct estimation of cardiac bi-ventricular volumes[4].

In recent years, medical image segmentation has witnessed several innovative approaches to improve efficiency and accuracy, particularly in the domain of cardiac MRI segmentation. These approaches aim to overcome various challenges such as inter-class indistinction, intra-class inconsistency, and high computational complexity.

For instance, the Boundary-Aware Lightweight Transformer (BATFormer) method builds cross-scale global interaction with lower computational complexity and has shown impressive results on ACDC and ISIC 2018 datasets[5]. Another recent development, ShapePU[6], is a scribble-guided method for cardiac segmentation based on the Positive-Unlabeled (PU) learning framework and global consistency regularization, which surpasses fully supervised approaches on the ACDC and MSCMRseg datasets.

Moreover, the Bidirectional Copy-Paste method reduces the empirical distribution gap in semi-supervised medical image segmentation datasets[7]. Lastly, the Learning Directional Feature Maps method exploits directional feature maps to address the challenges of inter-class indistinction and intra-class inconsistency in cardiac MRI segmentation[8].

Building on these works, I aim to develop a robust pipeline for automatic diagnosis of cardiac diseases from CMRI images by classifying them into five diagnostic classes: healthy controls, myocardial infarction, dilated cardiomyopathy, hypertrophic cardiomyopathy, and abnormal right ventricle. To achieve this goal, I first segment the left ventricle cavity from the myocardium and then extract relevant cardiac parameters, such as ejection fraction and ventricular volume, at two different time points in the cardiac cycle (end of cardiac contraction and end of dilation). These features are subsequently used to train machine learning classifiers for subject classification.

In this paper, I present a comprehensive approach for left ventricle segmentation and feature extraction, followed by the implementation of random forest and XGBoost classifiers for automatic cardiac diagnosis. I detail the methods and techniques used for each step of the pipeline, providing insights into the decision-making process and the rationale behind the choices made.

The results obtained from the random forest and XGBoost classifiers are thoroughly analyzed, and I discuss the implications of these findings for the field of cardiac diagnosis.

2 Dataset

This challenge provides a dataset of 150 subjects' MRI images, corresponding segmentations, and metadata such as subject height and weight. The goal is to classify the subjects in the test set into one of five diagnostic classes using only the provided data. The classes are healthy control, myocardial infarction, dilated cardiomyopathy, hypertrophic cardiomyopathy, and abnormal right ventricle. The MRI images are provided at two different time points in the cardiac cycle: end diastole and end systole. The provided segmentations contain three substructures: left ventricle cavity, right ventricle cavity, and myocardium. The left ventricle cavity label is missing in the test set segmentations and must be segmented using any suitable computer vision or machine learning technique as an intermediate step. The data has already been split into a training-validation set (100 subjects) and a test set (50 subjects). Only the classification for the training-validation set is provided, and the challenge is to estimate the correct class for each subject in the test set.

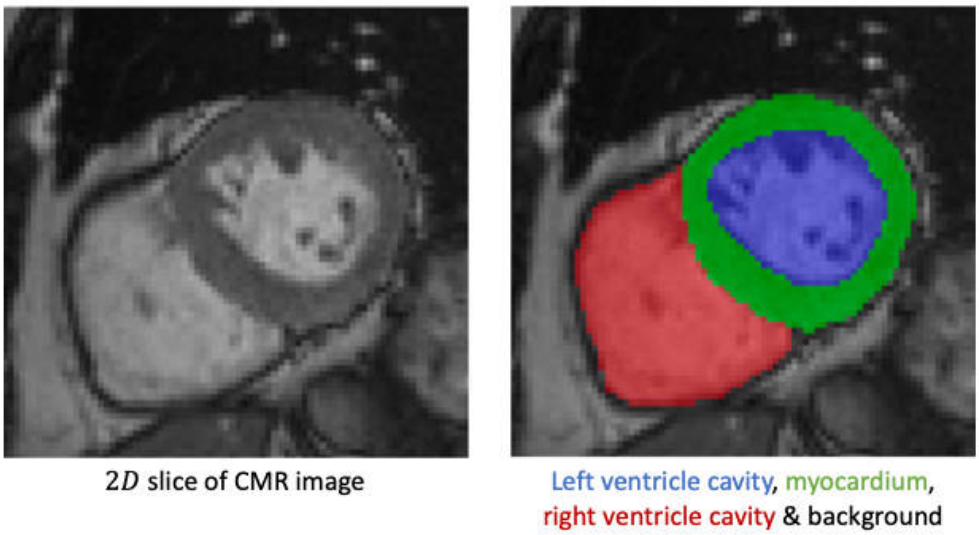

Figure 1: Graph showing the segmentations of dataset.

3 Method

3.1 LV cavity segmentation by U-net

Firstly, I implement a U-Net model for the segmentation of the left ventricle (LV) in cardiac MRI images. The U-Net architecture comprises a series of convolutional, dropout, and pooling layers in the contracting path, followed by transposed convolutional layers for upsampling in the expanding path, with skip connections between corresponding layers.

```
Input
|
v
[ENCODER]
(Conv -> ReLU -> Dropout -> Conv -> ReLU) -> MaxPool -> 32 filters
|
v
(Conv -> ReLU -> Dropout -> Conv -> ReLU) -> MaxPool -> 64 filters
|
v
(Conv -> ReLU -> Dropout -> Conv -> ReLU) -> MaxPool -> 128 filters
|
v
(Conv -> ReLU -> Dropout -> Conv -> ReLU) -> MaxPool -> 256 filters
|
v
[BOTTLENECK]
(Conv -> ReLU -> Dropout -> Conv -> ReLU) -> 512 filters
|
v
[DECODER]
Transposed Conv -> Concatenate -> (Conv -> ReLU -> Dropout -> Conv -> ReLU) -> 256 filters
|
v
Transposed Conv -> Concatenate -> (Conv -> ReLU -> Dropout -> Conv -> ReLU) -> 128 filters
|
v
Transposed Conv -> Concatenate -> (Conv -> ReLU -> Dropout -> Conv -> ReLU) -> 64 filters
|
v
Transposed Conv -> Concatenate -> (Conv -> ReLU -> Dropout -> Conv -> ReLU) -> 32 filters
|
v
[OUTPUT]
(Conv -> Sigmoid) -> 1 filter
|
v
Output
```

Figure 2: U-net Structure

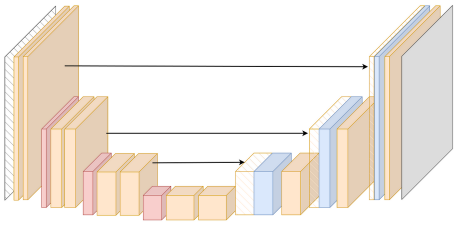

Figure 3: U-net Structure 2

The model is compiled using the Adam optimizer and binary cross-entropy loss function, with accuracy as the performance metric. We train the model using a batch size of 16 for a specified number of epochs, utilizing the validation dataset to monitor the model's performance during training.

The binary cross-entropy loss function, also known as the logarithmic loss or log loss, is a common loss function used for binary classification tasks, such as in our study for the segmentation of the left ventricle in cardiac MRI images. Given a set of true binary labels, $y_i \in 0, 1$, and their corresponding predicted probabilities, p_i, the binary cross-entropy loss function is defined as:

$$L(y, p) = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log p_i + (1 - y_i) \log (1 - p_i)] \tag{1}$$

where N denotes the number of samples. The binary cross-entropy loss function measures the difference between the true labels and the predicted probabilities, penalizing incorrect predictions. It encourages the model to produce probabilities close to the true labels, thereby improving the segmentation performance.

In the context of our study, the true labels correspond to the ground truth segmentation masks of the left ventricle, where $y_i = 1$ indicates the presence of the left ventricle, and $y_i = 0$ indicates the background. The predicted probabilities p_i are generated by the U-Net model. By minimizing the binary cross-entropy loss during training, our model learns to accurately segment the left ventricle in cardiac MRI images.

After training, I evaluate the model on the validation set to obtain the validation loss and accuracy. Subsequently, I use the trained model to make predictions on the validation set and apply a threshold of 0.5 to convert the predicted probabilities into binary masks, representing the segmented left ventricle see figure:4.

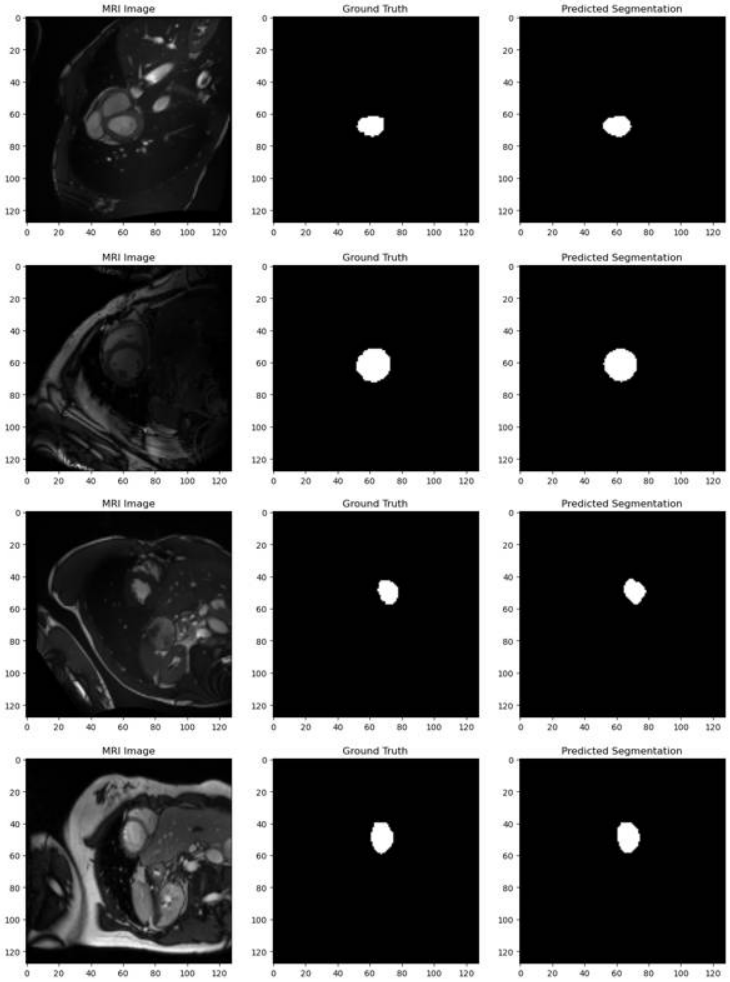

Figure 4: Result of U-net model

According to the segmentation prediction results, the cross-section is fine, but the coronal and sagittal results are very poor, which might be due to the fact that the model was trained only on axial slices. To improve the model's performance, I can try the following approaches:

- Train on multiple orientations: Instead of training only on axial slices, include slices from coronal and sagittal orientations in the training data. This will allow the model to learn features from all three orientations and improve its performance on each view.
- Use a 3D segmentation model: Instead of using a 2D model that processes each slice independently, consider using a 3D segmentation model that takes into account the spatial information between slices.
- Post-processing: Apply post-processing techniques to improve the segmentation results. For example, I can use morphological operations (e.g., opening, closing, or hole-filling) or connected component analysis to remove small, disconnected regions in the predicted masks.

3.2 LV cavity segmentation by Image operation

But for time reasons, I gave up on continuing with the deep learning approach because of the amount of time needed to train the model.

In test dataset, I observed that the area enclosed by the myocardium is the left ventricle cavity. To account for this observation, I developed a method to label the area enclosed by the myocardium as the left ventricle cavity in the test dataset. The method is implemented in the *classify_area_within_myocardium* function, which takes a segmentation

mask as input and returns an updated segmentation mask with the enclosed areas labeled correctly.The method is implemented as follows:

1. Iterate through each slice in the segmentation mask.
2. Extract the contours of the myocardium using the OpenCV library.
3. Fill the contours with the label of the left ventricle cavity (label value 3).
4. Update the original segmentation mask, replacing the area enclosed by the myocardium with the left ventricle cavity label.

As you can see figure:5, the segmentation effect is much more accurate and more beneficial to our feature extraction later.

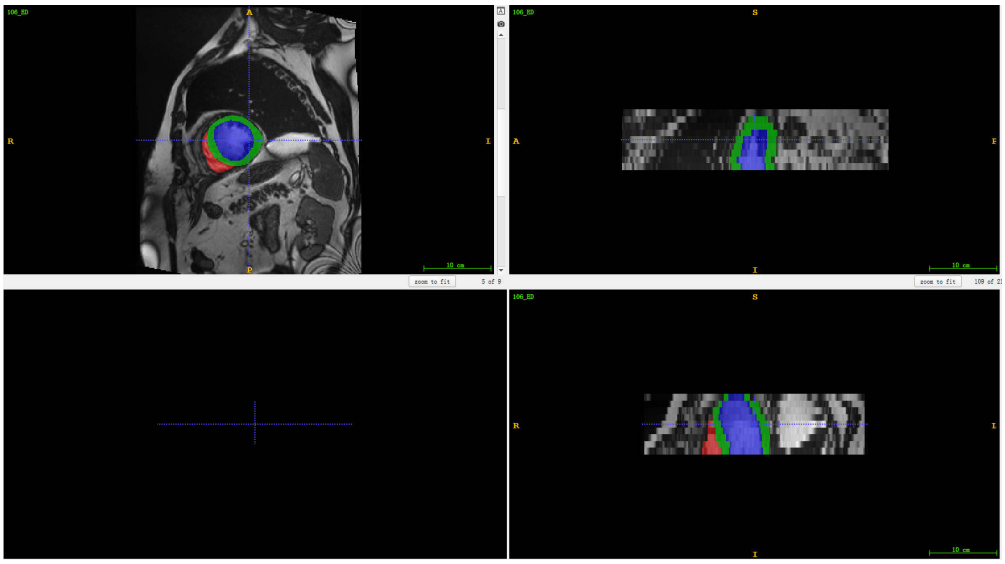

Figure 5: Result of Image operation method (test106)

4 Feature extraction and Classification

In order to develop an automated cardiac diagnosis system, we utilize the following 10 attributes extracted from the training dataset[9]:

- Ejection fraction of the left ventricle (LV EF)

$$LV\_EF = \frac{LV\_EDV - LV\_ESV}{LV\_EDV} \times 100 \tag{2}$$

- Ejection fraction of the right ventricle (RV EF)

$$RV\_EF = \frac{RV\_EDV - RV\_ESV}{RV\_EDV} \times 100 \tag{3}$$

- Volume of the left ventricle at end-systole (LV ESV)

$$LV\_ESV = \sum_{i}^{N} V_{LV\_ESV_i} \tag{4}$$

- Volume of the left ventricle at end-diastole (LV EDV)

$$LV\_EDV = \sum_{i}^{N} V_{LV\_EDV_i} \tag{5}$$

- Volume of the right ventricle at end-systole (RV ESV)

$$RV\_ESV = \sum_{i}^{N} V_{RV\_ESV_i} \tag{6}$$

- Volume of the right ventricle at end-diastole (RV EDV)

$$RV\_EDV = \sum_{i}^{N} V_{RV\_EDV_i} \tag{7}$$

- Mass of the myocardium at end-diastole (Myocardium Mass)

$$Myocardium\_Mass = \sum_{i}^{N} V_{Myocardium\_ED_i} \tag{8}$$

- Volume of the myocardium at end-systole (Myocardium Volume)

$$Myocardium\_Volume = \sum_{i}^{N} V_{Myocardium\_ES_i} \tag{9}$$

- Patient height
- Patient weight

We calculate these attributes using a Python function called *process_subject*, which takes the data directory and subject ID as inputs, and processes the corresponding subject's MRI data. The function loads the end-diastole (ED) and end-systole (ES) MRI images and their corresponding segmentation masks. It then calculates the voxel volume for both ED and ES images, and utilizes the *calculate_volumes* function to compute the volumes of the left and right ventricles and the myocardium. Finally, the function calculates the ejection fraction for both ventricles and returns all extracted attributes.

- NOR: Examination with normal cardiac anatomy and function. The ejection fraction is greater than 50%, the wall thickness in diastole is lower than 12 mm, the LV diastolic volume is below 90 mL/m2 for men and 80 mL/m2 for women. The RV is normal for each patient (RV volume less than 100 mL/m2 and RV ejection fraction above 40%). The visual analysis of the segmental LV and RV myocardial contraction is normal.
- MINF: Patients with a systolic heart failure with infarction. Subjects have an ejection fraction below 40% and abnormal myocardial contractions. Some subjects have a high diastolic LV volume due to a remodeling of the LV to compensate for the myocardial infarction.
- DCM: Patients with dilated cardiomyopathy have an ejection fraction below 40%, a LV volume greater than 100 mL/m2 and a wall thickness in diastole smaller than 12 mm. As a consequence of dilated LV, some patients of this category have a dilated RV and/or a high LV mass.
- HCM: Patients with hypertrophic cardiomyopathy, i.e. a normal cardiac function (ejection fraction greater than 55%) but with myocardial segments thicker than 15 mm in diastole. In this category, patients can present abnormal cardiac mass indices with values above 110 g/m2.
- ARV: Patients with abnormal right ventricle have a RV volume greater than 110 mL/m2 for men, and greater than 100 mL/m2 for women, or/and a RV ejection fraction below 40%. Almost every subject in this subgroup has a normal LV.

Based on the above characteristics[10], I found that it was not straightforward to distinguish the various diseases. These 11 attributes were used for training a Random Forest classifier and XGBoost classifier. Both Random Forest and XGBoost were chosen in this study due to their unique characteristics that make them suitable for the task of classifying cardiac pathologies using CMRI-derived features.

Random Forest Classifier[11]: Random Forest is an ensemble learning technique that constructs a collection of decision trees and aggregates their results to make a final prediction. The technique improves the generalization performance and reduces overfitting by averaging the outputs of multiple trees. In a classification problem, the final prediction is the majority class among all the decision trees, while in a regression problem, the final prediction is the average of individual tree predictions.

- It handles both linear and non-linear relationships between features and the target variable, which can be beneficial when dealing with complex medical data.
- Random Forest can handle a mix of continuous and categorical features, making it a versatile choice for a variety of problems.
- It provides robustness against noise in the data, which is an advantage in medical imaging applications where data quality can vary.

A decision tree is built by recursively splitting nodes on features that provide the most information gain (or other criteria like Gini impurity). The information gain is calculated using the following equation:

$$InformationGain = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} \cdot Entropy(S_i) \tag{10}$$

where S is the dataset, S_i is the subset of the dataset after the split, and Entropy is calculated as:

$$Entropy(S) = -\sum_{c=1}^{C} p(c) \cdot \log_2 p(c) \tag{11}$$

where p(c) is the proportion of samples belonging to class c.

XGBoost Classifier[12]: XGBoost (Extreme Gradient Boosting) is an advanced implementation of gradient boosting, a machine learning technique that builds an ensemble of weak learners (decision trees) in a stagewise fashion. At each stage, a new tree is added to the model to correct the errors made by the existing trees. The final prediction is a weighted sum of the individual tree predictions.

- It is highly efficient and scalable, making it suitable for large datasets or computationally intensive tasks.
- XGBoost can handle both linear and non-linear relationships between features and the target variable, similar to Random Forest.
- It offers regularization options that help control overfitting, improving the model's generalization capabilities.

In XGBoost, the objective function is a combination of a loss function and a regularization term:

$$Obj(\theta) = L(y, \hat{y}) + Reg(\theta) \tag{12}$$

where $L(y, \hat{y})$ is the loss function comparing the true labels (y) and the predicted labels ($\hat{y}$), and $Reg(\theta)$ is the regularization term for the model parameters (θ).

The loss function used in XGBoost is typically the log loss for classification or the mean squared error for regression:

$$L(y, \hat{y}) = -\sum_{i=1}^{n} [y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i)] \tag{13}$$

$$L(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{14}$$

The regularization term in XGBoost penalizes the complexity of the trees by considering both the number of leaves and the weights of the splits:

$$Reg(\theta) = \gamma \cdot T + \frac{1}{2} \cdot \lambda \cdot \sum_{j=1}^{T} w_j^2 \tag{15}$$

where T is the number of leaves in the tree, w_j is the weight of the j-th leaf, γ controls the complexity of the trees, and λ controls the L2 regularization of the leaf weights.

The Random Forest classifier is trained to diagnose patients with Dilated cardiomyopathy (DCM), Hypertrophic cardiomyopathy (HCM), Myocardial infarction (MNF), Abnormal right ventricle (ARV), and normal patients (NOR). The dataset is preprocessed, and hyperparameters are tuned using GridSearchCV. The best parameters for the classifier are found to be:

- bootstrap: True
- max_depth: 20
- min_samples_leaf: 1
- min_samples_split: 2
- n_estimators: 100

Cross-validation Scores	[0.95, 0.95, 0.95, 0.7, 0.9]
Average Cross-validation Score	0.89

	Precision	Recall	F1-score	Support
0	1.00	0.75	0.86	4
1	1.00	0.67	0.80	3
2	0.83	1.00	0.91	5
3	0.83	1.00	0.91	5
4	1.00	1.00	1.00	3
Accuracy			0.90	20
Macro avg	0.93	0.88	0.90	20
Weighted avg	0.92	0.90	0.90	20

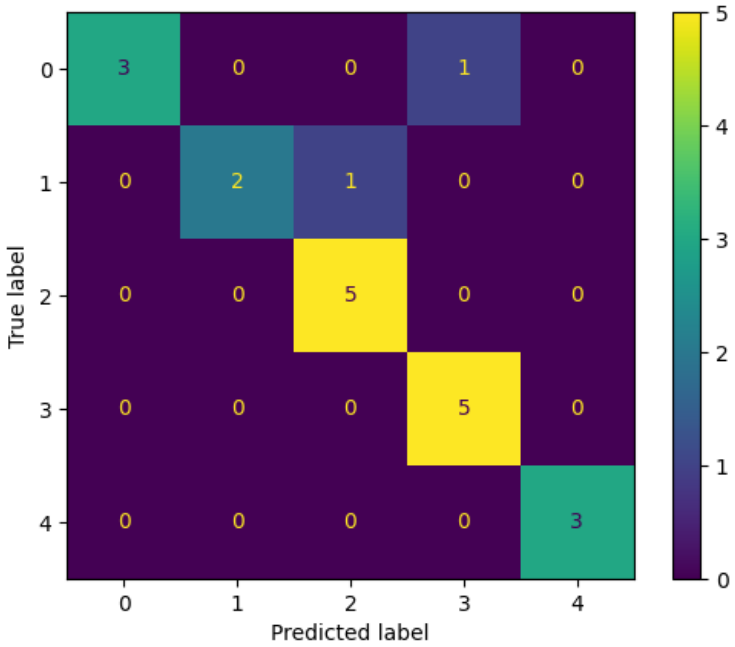

Figure 6: The confusion Matrix of Random Forest

The classifier achieves an average cross-validation score of 0.89. When evaluated on the test dataset, the classifier achieves an overall accuracy of 0.90 with high precision and recall values for each category. The confusion matrix see figure:6 plotted for the model demonstrates its effectiveness in diagnosing patients based on the calculated cardiac parameters.

In addition to the Random Forest classifier, an XGBoost classifier is also employed to diagnose patients based on the 11 attributes derived from the training dataset. The dataset is preprocessed, and hyperparameters are tuned using GridSearchCV. The best parameters for the classifier are found to be:

- colsample_bytree: 0.7
- learning_rate: 0.1
- max_depth: 3
- min_child_weight: 1
- n_estimators: 100
- subsample: 0.7

Cross-validation Scores	[0.85, 0.9, 0.95, 0.7, 0.85]
Average Cross-validation Score	0.85

	Precision	Recall	F1-score	Support
0	1.00	0.75	0.86	4
1	1.00	0.67	0.80	3
2	0.83	1.00	0.91	5
3	0.83	1.00	0.91	5
4	1.00	1.00	1.00	3
Accuracy			0.90	20
Macro avg	0.93	0.88	0.90	20
Weighted avg	0.92	0.90	0.90	20

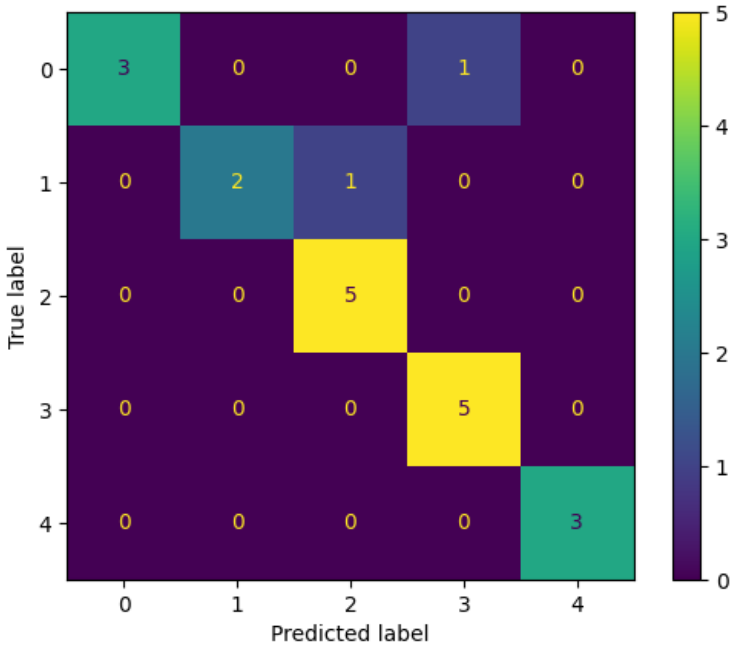

Figure 7: The confusion Matrix of Random Forest

The classifier achieves an average cross-validation score of 0.85. When evaluated on the test dataset, the classifier achieves an overall accuracy of 0.90 with high precision and recall values for each category. The confusion matrix see figure:7 plotted for the model demonstrates its effectiveness in diagnosing patients based on the calculated cardiac parameters, similar to the results obtained using the Random Forest classifier.

Both the Random Forest and XGBoost classifiers exhibit strong performance in diagnosing patients based on the 11 cardiac parameters. The Random Forest classifier achieved an average cross-validation score of 0.89 and an overall test

set accuracy of 0.90. Similarly, the XGBoost classifier achieved an average cross-validation score of 0.85 and an overall test set accuracy of 0.90. These results indicate that both classifiers are effective in diagnosing patients based on the cardiac parameters.

The precision, recall, and F1-scores for both classifiers across all categories are also quite high, indicating that they are able to correctly identify the different categories with few false positives or false negatives. It's worth noting that both classifiers show some room for improvement, particularly in handling certain categories, such as Category 1 (Hypertrophic cardiomyopathy) and Category 0 (Dilated cardiomyopathy), where the recall values were lower than the other categories.

5 Conclusion

In conclusion, this study showcases the potential of using CMRI data in combination with machine learning techniques for early detection and improved management of cardiac pathologies. The proposed pipeline, consisting of left ventricle cavity segmentation and extraction of relevant cardiac parameters, effectively leverages the strengths of Random Forest and XGBoost classifiers for automatic cardiac diagnosis.

The results obtained in this study are promising, with the Random Forest classifier achieving an accuracy of 89% and the XGBoost classifier attaining an accuracy of 85%. These findings highlight the potential of machine learning algorithms in providing accurate and efficient diagnosis of various cardiac conditions using CMRI-derived cardiac parameters.

In future work, further refinement of the segmentation process, exploration of additional features, and investigation of alternative machine learning algorithms may enhance the performance of the proposed pipeline.

References

[1] Hae-Yeoun Lee, Noel C. F. Codella, Matthew D. Cham, Jonathan W. Weinsaft, and Yi Wang. Automatic left ventricle segmentation using iterative thresholding and an active contour model with adaptation on short-axis cardiac mri. *IEEE Transactions on Biomedical Engineering*, 57(4):905–913, 2010.

[2] M. Lynch, O. Ghita, and P.F. Whelan. Automatic segmentation of the left ventricle cavity and myocardium in mri data. *Computers in Biology and Medicine*, 36(4):389–407, 2006.

[3] Tomasz Pieciak. Segmentation of the left ventricle using active contour method with gradient vector flow forces in short-axis mri. volume 7339, pages 24–35, 01 2012.

[4] Ján Margeta, Ezequiel Geremia, Antonio Criminisi, and Nicholas Ayache. Layered spatio-temporal forests for left ventricle segmentation from 4d cardiac mri data. In Oscar Camara, Ender Konukoglu, Mihaela Pop, Kawal Rhode, Maxime Sermesant, and Alistair Young, editors, *Statistical Atlases and Computational Models of the Heart. Imaging and Modelling Challenges*, pages 109–119, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[5] Xian Lin, Li Yu, Kwang-Ting Cheng, and Zengqiang Yan. Batformer: Towards boundary-aware lightweight transformer for efficient medical image segmentation. *IEEE Journal of Biomedical and Health Informatics*, pages 1–13, 2023.

[6] Ke Zhang and Xiahai Zhuang. Shapepu: A new pu learning framework regularized by global consistency for scribble supervised cardiac segmentation. In Linwei Wang, Qi Dou, P. Thomas Fletcher, Stefanie Speidel, and Shuo Li, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*, pages 162–172, Cham, 2022. Springer Nature Switzerland.

[7] Yunhao Bai, Duowen Chen, Qingli Li, Wei Shen, and Yan Wang. Bidirectional copy-paste for semi-supervised medical image segmentation, 2023.

[8] Feng Cheng, Cheng Chen, Yukang Wang, Heshui Shi, Yukun Cao, Dandan Tu, Changzheng Zhang, and Yongchao Xu. Learning directional feature maps for cardiac mri segmentation. In Anne L. Martel, Purang Abolmaesumi, Danail Stoyanov, Diana Mateus, Maria A. Zuluaga, S. Kevin Zhou, Daniel Racoceanu, and Leo Joskowicz, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, pages 108–117, Cham, 2020. Springer International Publishing.

[9] Fabian Isensee, Paul F 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. 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*, pages 120–129. Springer, 2018.

[10] Olivier Bernard, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, Gerard Sanroma, Sandy Napel, Steffen Petersen, Georgios Tziritas, Elias Grinias, Mahendra Khened, Varghese Alex Kollerathu, Ganapathy Krishnamurthi, Marc-Michel Rohé, Xavier Pennec, Maxime Sermesant, Fabian Isensee, Paul Jäger, Klaus H. Maier-Hein, Peter M. Full, Ivo Wolf, Sandy Engelhardt, Christian F. Baumgartner, Lisa M. Koch, Jelmer M. Wolterink, Ivana Išgum, Yeonggul Jang, Yoonmi Hong, Jay Patravali, Shubham Jain, Olivier Humbert, and Pierre-Marc Jodoin. Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: Is the problem solved? *IEEE Transactions on Medical Imaging*, 37(11):2514–2525, 2018.

[11] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[12] Tianqi Chen, Tong He, Michael Benesty, Vadim Khotilovich, Yuan Tang, Hyunsu Cho, Kailong Chen, Rory Mitchell, Ignacio Cano, Tianyi Zhou, et al. Xgboost: extreme gradient boosting. *R package version 0.4-2*, 1(4):1–4, 2015.