Enabling ECG Classification with Hierachical End-To-End Training

Tizian C. Dege[1], Maurice Rohr[1], Christoph Hoog Antink[1]

[1]KIS*MED - AI Systems in Medicine,
Technische Universität Darmstadt, Darmstadt, Germany

Abstract

*This paper introduces a novel approach for ECG classification using hierarchical end-to-end training. We focus on the automated classification of ECG images, as part of The George B. Moody PhysioNet Challenge 2024, a crucial task for early detection of cardiovascular diseases (CVDs). Our method leverages the PTB-XL dataset for generating synthetic ECG images and employs a two-step model: a You Only Look Once (YOLO) model for lead extraction and Convolutional Neural Networks (CNNs) for classification. The models can be trained separately on low-memory GPUs, making them suitable for deployment in resource-constrained environments. Our approach achieved a macro F_1-Score of 0.873 on our local validation dataset, indicating high performance in detecting multiple ECG abnormalities. This solution has the potential to enhance diagnostic capabilities, particularly in regions with limited access to advanced medical infrastructure. Our model for team DeTECRohr on the PhysioNet2024 hidden test achieves a F_1-score of 0.153 (Rank 14/16). **Disclaimer**: In this acrticle we present a reworked version of the model, for which we have no official score.*

1. Introduction

Cardiovascular diseases (CVDs) are a leading cause of mortality worldwide and are expected to increase in prevalence over the coming years [1]. In 2021, 20.5 million deaths were attributed to CVDs, 80% of which occur in low- to middle-income countries [2]. CVDs can lead to various conditions, such as coronary or ischemic heart disease, arrhythmias, and cardiac arrests. Early detection of these conditions is crucial for patients to receive appropriate treatment and reduce mortality.

The Electrocardiogram (ECG) remains the primary tool for detecting CVDs. ECGs are non-invasive diagnostic tools that measure the electrical activity of the heart. However, trained cardiologists are required for accurate diagnosis, as ECGs can display diverse patterns that require professional expertise for correct interpretation. In recent years, machine learning (ML) has demonstrated the capability to match the diagnostic performance of trained cardiologists or assist new cardiologists in their diagnoses. Especially, for countries with lacking health infrastructure, an automatic diagnostic tool with the help of ML would be beneficial.

Recent advancements in machine learning (ML) have shown noteworthy results in various medical diagnostic fields, significantly boosting accuracy and efficiency. In particular, ML-based ECG diagnostics have proven highly effective. While these results are promising, most studies have focused on diagnostics using digitized signals. However, developing countries often lack access to advanced medical devices and rely on printed ECGs for diagnosis. This limitation makes using modern ML-based solutions impractical, further compromising the quality of healthcare in these regions.

In this paper, we present a novel approach for detecting multiple abnormalities from ECG images as part of the PhysioNet Challenge 2024 [3]. Our solution is divided into three parts: we use You Only Look Once (YOLO) [4] for single-lead extraction from images and employ Convolutional Neural Networks (CNNs) to classify both the whole image and the extracted single leads. Moreover, we have optimized our solutions to be trainable on low-memory GPUs, allowing each part to be trained separately. This enables training on older machines, which is important given that this tool may be more frequently used in low-resource environments. The code and models are publicly available at `https://github.com/TixXx1337/PhysioNet2024`.

2. Methods

The following section introduces the implemented pipeline in the work presented here. First, we introduce the data generation method used. Afterward, we describe the implemented feature extraction using YOLO. Furthermore, we discuss the global and local models, which were trained separately on our dataset. Finally, we show the final model implementation. The pipeline for this work can be seen in Figure 1. All experiments were run on a NVIDIA Quadro RTX 5000 (16GB VRAM).

 ISSN: 2325-887X DOI: 10.22489/CinC.2024.370

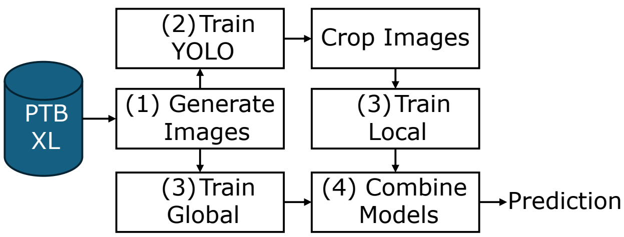

Figure 1: The pipeline of this work is divided into the following steps: (1) Creating synthetic images, (2) Training YOLO, (3) Training the individual models, (4) Combining the trained models.

2.1. Sythetic Data Creation

For creating authentic data, we used the ECG-Image-Kit provided by Shivashankara et al. [5]. The tool allows us to create realistic ECG images from raw 12-lead ECG signals. We use the PTB-XL dataset [6, 7] as raw data. The PTB-XL is one of the largest available 12-lead ECG datasets, containing 21,837 records from 18,885 patients. The dataset provides a diverse pool of ECG records, all sampled at 500 Hz, independently labelled by two cardiologists. Furthermore, the records may have multiple labels, which makes the classification more complicated than a single-class prediction. Since we assume that realistic ECG images are contaminated with artefacts, we added different steps to introduce artefacts to the images. To do this, we modified the configuration to use the augmentations listed in Table 1. All images have 12 short leads showing 2.5-second segments of the ECG and one long lead, which is a random channel of the 12 short leads and shows the entire 10-second segment.

Augmentation	Parameter
Header	20% present
Resolution	50-250 DPI
Black/White	20%
Rotation	0-45°
Padding	0-2 inches
Pulse	20%

Table 1: All Parameters used for the augmentation of the data. The data has a high variety across all created images.

For augmentation we use 6 different settings. The augmentations address various challenges in ECG data analysis. Header presence or absence simulates variability in metadata across different datasets. Varying the resolution between 50-250 DPI ensures the model can handle different image qualities. Converting images to black and white prepares the model for contrast images, while random rotation accounts for potential misalignment. Padding helps the model focus on relevant ECG data, ignoring irrelevant surrounding areas. Including a calibration pulse faciliates a more robust classification in realistic scenarios, where this is a common seen artefact. Finally, the augmentation are combined randomly, which prevents reliance on any specific augmentation, improving generalization of the model.

2.2. Single Lead Extraction

We assume that for classification, the most important information is contained within the independent leads. This approach mirrors how cardiologists classify a printed ECG, similar to the saliency mapping presented in Simonyan et al. [8]. Extracting individual leads reduces the feature space of the input image while increasing the information density. To detect all independent leads, we trained YOLO, a single-shot object detector. We chose YOLOv8m, a state-of-the-art object detector, due to its low memory footprint and high inference speed [9]. The model was trained to detect each short lead independently, as well as the long lead. The model was trained to detect each short lead independently, as well as the long lead. For training, we used all available data generated from the PTB-XL dataset, applying an 80/20 split for training and validation. Cross-validation was not conducted due to the simplicity of the task and the time constraints associated with repeating the training multiple times. The performance of the object detector is presented in Section 3.1.

2.3. Classification Models

For the classification task, two separate models were used. The two models were implemented with PyTorch and used a simple CNN structure for classification. The models are called the global model, which uses the whole image as input for classification, and the local model which uses the independent images of each lead as input. For the local model, each image is split into 13 cropped images which were detected by the YOLO model in the image. Then the input is passed into 13 parallel CNNs and then the output of each model is concatenated over a fully connected layer and used for the final prediction. A closer insight into the models can be found in Figure 2.

Preprocessing: The images were denoised with a Gaussian blur filter (kernel size 3x3) and then either directly used or cropped into sub-images for the local model. Since we can not guarantee all leads are detected when cropping, we assume all were detected for the inputs of our local model. If independent leads are missing, we resample a lead. If no leads are detected, we use the downsampled whole image as input for the local models.

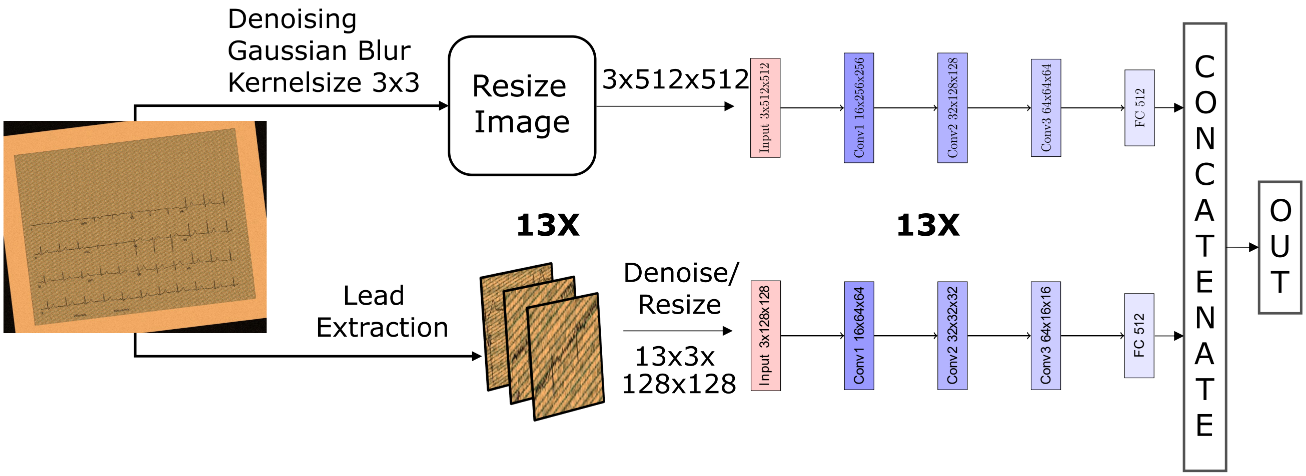

Figure 2: For the final model the Image is used on the global model as a whole. The Image is resized to 512x512 and then used for training. For the local model, the image is first cropped into 13 cropped images resized to 128x128 and then fed into the local model. Then the output of both models are concatenated and used for prediction.

Label	Count
NORM	1379
Old MI	816
CD	753
STTC	746
HYP	402
AFIB/AFL	240
PVC	180
TACHY	115
BRADY	96
PAC	70
Acute MI	24

Before Oversampling

$\rightarrow$

Label	Count
NORM	3702
Old MI	4805
CD	4971
STTC	5620
HYP	3662
AFIB/AFL	2523
PVC	2388
TACHY	1966
BRADY	1639
PAC	1833
Acute MI	1490

After Oversampling

Table 2: The Databalance before and after random resampling. The minority class presence of 0.5% is increased to 4.3%. A perfectly balanced dataset requires a 9.09% presence of each class. The data represents the training data for one fold. The oversampling is only done for the training split.

Training: As we are facing a multi-class and multi-label task, we are using Binary Cross Entropy with Logits Loss (BCEWithLogitsLoss), where the raw output of the model is used for calculating the loss. L is the calculated loss from the output x and the true label y. N is the the number of samples, in our case this represents the batch size. σ is the sigmoid function, which represents the the propabilities for each logit. x_i represents the calculated likelihood for each class, while y_i represents the true label for each class represented either by a 0 or 1.

$$L(x, y) = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i \log(\sigma(x_i)) + (1-y_i)\log(1-\sigma(x_i))\right) \quad (1)$$

Data Balancing As previously described we are facing a multi-label multi-class problem, therefore fully balanced data is almost impossible. Due to this problem, we make the following adjustments for balancing the data. If an image has multiple labels N for one image we randomly oversample the label to be used N times. This increases the weight of images with multiple labels on the training. Furthermore, we try to randomly oversample the minority classes. The exact numbers can be seen in Table 2. While there is no perfect balanced dataset achieved the dataset represents a more realistic balancing, since classes that are often present in combination with other classes are used more often.

3. Results

3.1. Lead Extraction Performance

The feature extraction is a pivotal task for the here-described pipeline to work. We trained the YOLO model for 10 epochs on all available data. Although, the training data has a large variance due to our augmentation described in Section 2, the model still achieves an mean Average Precision (mAP) of 0.995 on our validation dataset. This lets us assume YOLO is suited to be used in our single lead extraction in the local model.

3.2. Classification Results

For testing the performance of our classification model we trained the global, local and combined model on the same balanced training data and used a withheld validation dataset. As a performance metric we used the macro F_1-Score, as this is the same score the PhysioNet Challenge uses. The macro F_1-Score is the unweighted mean of all per class F_1-Scores. The F_1-Score of the global/local and combined model can be seen in Figure 3.

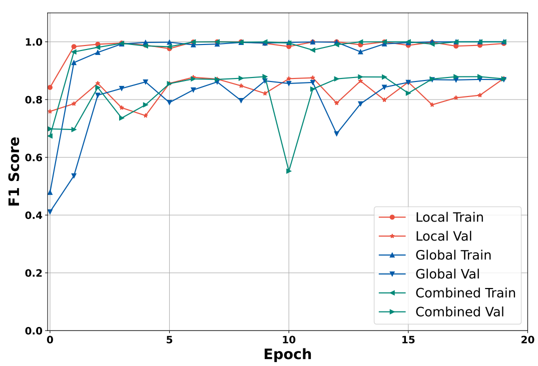

Figure 3: F_1-Score for training and validation data. All three models were trained and tested on the same data. The validation data was withheld from the training data.

Task	Test	Rank
Classification	0.153	14/16

Table 3: Macro F-measure for our selected entry (team DeTECRohr) on the classification task, and the ranking of our team on the hidden test set. This entry does not represent the models described in this work.

4. Conclusion and Future Outlook

All three models show promising results on our own validation set. However, on our dataset, there does not seem to be much benefit, as all three models perform similarly. The F_1-score for all models is close to 0.87. Besides this, the combined model shows higher stability and a faster convergence speed for the validation data. Furthermore, the local model has a higher variance in performance across the epochs. However, the local model already achieves a notable F_1 Score in the first epoch, which confirms our expectation, that the single leads have a higher information density in comparison to the whole image. Furthermore, we assume the lead extraction can be interesting for the digitization task, as we assume the higher information density can be of used for that task as well.

References

[1] Gaidai O, Cao Y, Loginov S. Global Cardiovascular Diseases Death Rate Prediction. Current Problems in Cardiology 2023;48(5):101622.

[2] Di Cesare M, Perel P, Taylor S, Kabudula C, Bixby H, Gaziano TA, McGhie DV, Mwangi J, Pervan B, Narula J, Pineiro D, Pinto FJ. The Heart of the World. Global Heart 2024;19(1):11.

[3] Reyna MA, Deepanshi, Weigle J, Koscova Z, Elola A, Seyedi S, Campbell K, Clifford GD, Sameni R. Digitization and Classification of ECG Images: The George B. Moody PhysioNet Challenge 2024. Computing in Cardiology 2024;51:1–4.

[4] Redmon J, Divvala S, Girshick R, Farhadi A. You Only Look Once: Unified, Real-Time Object Detection. In 29th IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE. ISBN 978-1-4673-8851-1, 2016; 779–788.

[5] Shivashankara KK, Deepanshi, Mehri Shervedani A, Clifford GD, Reyna MA, Sameni R. ECG-Image-Kit: A Synthetic Image Generation Toolbox To Facilitate Feep Learning-Based Electrocardiogram Digitization. Physiological Measurement 2024;45(5).

[6] Wagner P, Strodthoff N, Bousseljot RD, Kreiseler D, Lunze FI, Samek W, Schaeffter T. PTB-XL, a Large Publicly Available Electrocardiography Dataset. Scientific Data 2020; 7(1):154.

[7] Reyna MA, Deepanshi, Weigle J, Koscova Z, Campbell K, Shivashankara KK, Saghafi S, Nikookar S, Motie-Shirazi M, Kiarashi Y, Seyedi S, Clifford GD, Sameni R. ECG-Image-Database: A Dataset of ECG Images with Real-World Imaging and Scanning Artifacts; A Foundation for Computerized ECG Image Digitization and Analysis, 2024. URL `https://arxiv.org/abs/2409.16612`.

[8] Simonyan K, Vedaldi A, Zisserman A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. ICLR, 2014; 1–8.

[9] Glenn Jocher, Ayush Chaurasia, Jing Qiu. Ultralytics YOLOv8, 2023. URL `https://github.com/ultralytics/ultralytics`.

Address for correspondence:

Tizian Dege
Merckstraße 25, 64283 Darmstadt, Germany
dege@kismed.tu-darmstadt.de