Object Detection of Stenosis in Coronary Angiography

Piero Sgrignuoli, Luca Agostinelli

Abstract—**Invasive coronary angiography remains the gold standard for diagnosing coronary artery disease, which may be complicated by both, patient-specifc anatomy and image quality. Deep learning techniques aimed at detecting coronary artery stenoses may facilitate the diagnosis. However, previous studies have failed to achieve superior accuracy and performance for real-time labeling. Our study is aimed at confirming the feasibility of coronary artery stenosis detection using deep learning methods. To reach this goal we trained and tested three promising detectors based on different neural network architectures (SSD ResNet-50, Faster R-CNN ResNet-50 and Efficientdet) using clinical angiography data of 100 patients. One neural networks have demonstrated superior results. The network based on SSD ResNet50 V1 FPN v2 resulted to be the most accurate. The relatively lightweight Efficientdet D0 v2 network proved an optimal accuracy-to-speed ratio model. While the model based on Faster R-CNN v2 has performed the worst in terms of accuracy, probably due to the restricted hardware environment on which we trained all the models. The resultant performance-accuracy balance of the modern neural networks, even if not ready to be used for real-time detection, confirms the feasibility of it. With more works to be done in order to reduce the mean inference time, these models could be used for real-time detection supporting the decision-making process of the Heart Team interpreting coronary angiography findings.**

I. Introduction

Coronary artery disease (CAD) is the leading cause of death worldwide[1], affecting over 120 million people[2]. The main cause of CAD is atherosclerotic plaque accumulation[3] in the epicardial arteries leading to a mismatch between myocardial oxygen supply and myocardial oxygen demand and commonly resulting in ischemia. Chest pain is the most likely symptom that occurs during physical and/or emotional stress, relieved promptly with rest or by taking nitroglycerin. This process can be modified by lifestyle adjustments, pharmacological therapies, and invasive interventions designed to achieve disease stabilization or regression[4]. Despite novel imaging modalities (e.g. coronary CT angiography) have been developed, invasive coronary angiography is the preferred diagnostic tool to assess the extent and severity of complex coronary artery disease according to the 2019 guidelines of the European Society of Cardiology[5][6].

Multivessel coronary artery disease affecting two or more coronary arteries requires interpretive expertise on the assessment of multiple parameters (the number of affected major coronary arteries, the location of lesions, the severity of stenosis, the length of the stenotic segment, tortuosity, etc.) during an intervention. The process of interpreting complex coronary vasculature, image noise, low contrast vessels, and non-uniform illumination is time-consuming[7], thereby posing certain challenges to the operator. Real-time automatic CAD detection and labeling may overcome the abovementioned difficulties by supporting the decision-making process.

To reach this goal we trained and tested three promising detectors based on different neural network architectures (SSD ResNet-50, Faster R-CNN ResNet-50 and Efficientdet) using clinical angiography data of 100 patients. One neural networks have demonstrated superior results. The network based on SSD ResNet50 V1 FPN v2 resulted to be the most accurate. The relatively lightweight Efficientdet D0 v2 network proved an optimal accuracy-to-speed ratio model. While the model based on Faster R-CNN v2 has performed the worst in terms of accuracy, probably due to the restricted hardware environment on which we trained all the models. The resultant performance-accuracy balance of the modern neural networks, even if not ready to be used for real-time detection, confirms the feasibility of it. With more works to be done in order to reduce the mean inference time, these models could be used for real-time detection supporting the decision-making process of the Heart Team interpreting coronary angiography findings.

This article is structured as follows: all related works are presented in section II. Section III discusses the methods used for the dataset and the networks with the related experimental protocols. Section IV presents all the results obtained and the related discussions. The conclusions are presented in Section V.

II. Related Works

A number of approaches for automatic or semi-automatic assessment of coronary artery diseases have been proposed by different research groups. These approaches follow the general scheme: (1) coronary artery tree extraction, (2) calculation of geometric dimensioning, and (3) analysis of the stenotic segment. The key stage that determines the speed and accuracy of such algorithms is based on the coronary artery tree extraction using the centerline extraction[8][9] the graph-based method[10][11][12] superpixel mapping[13][14] and machine/deep learning[15][16][17].

The last, being a powerful tool for computer vision and image classification, has shown great promise in CAD detection due to their performance, tuning flexibility, and optimization. The ultimate purpose that CNN developers and users are trying to meet is to strike an optimal balance between accuracy and speed, the so-called speed/accuracy trade-off[18];.

While some CNNs with high performance and optimal accuracy suitable for real-time segmentation can be used on mobile devices and low-end PCs, others with low performance are highly efficient for object detection (precision, recall, F1-score, mAP). Depending upon the task complexity and scope, this balance may vary and be achieved using the proper CNN architecture.

III. Materials and Methods

This section analyzes the methodology proposed in this paper. In Fig 1 is shown the workflow of the work performed. Section A describes the Dataset. Section B presents the networks trained for the stenosis object detection. Training settings are exposed in Section C. Finally Section D lists the metrics used for evaluation.

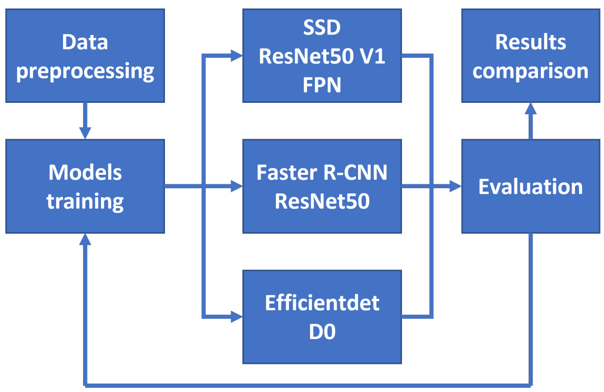

Fig. 1: Workflow.

A. Dataset

The dataset used contains angiographic imaging series of one hundred patients who underwent coronary angiography using Coroscop (Siemens) and Innova (GE Healthcare) at the Research Institute for Complex Problems of Cardiovascular Diseases (Kemerovo, Russia).

A total of 8325 grayscale images (100 patients) of 512 × 512 pixels are present.

The initial dataset have been divided into:

- 6660 (80%) images for the train set,
- 833 (10%) images for the validation set,
- 832 (10%) images for the test set.

In order to correctly estimate model performance, we did not randomly shuffle all 8325 images and then form data subsets. Patient series were first randomly chosen for the training, validation, and testing subsets in an 80:10:10 ratio, and then form those subsets. Such data split allows us to know that the validation and testing are done on the independent subsets of images and avoid bias in performance metrics.

Since the training process is quite time-consuming, we excluded the usage of cross-validation for the models. Data were labeled using LabelBox, a free version of SaaS (Sofware as a Service). It allows joint data labeling and subsequent validation by several specialists. Typical data labeling of the source images is shown in Fig. 2.

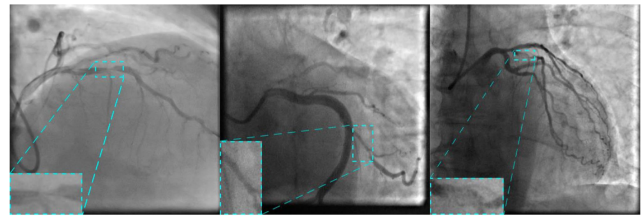

Fig. 2: Data labelling of the source images with the callouts of the detected stenotic lesions.

B. Models and pre-processing

Our goal is to train three different object detection models in order to detect a stenosis given an angiographic image. We examined three different models with various architectures:

- SSD ResNet-50 V1 FPN,
- Faster-RCNN ResNet-50 V1,
- Efficientdet D0.

These models were all taken from the Tensorflow Detection Model Zoo. The models are all pre-trained on the COCO 2017 dataset. While the SSD model and the Efficientdet one provides faster training times, their performance usually are not the best; on the other hand, the Faster R-CNN model usually scores higher in term of precision and accuracy, but the the training process is quite computational heavy.

To better choose training parameters related to the bounding boxes, we generated plots that let us visualize which parameters combination is the most fitting one. The first plot we produced was the aspect ratios histogram (Fig. 3), this plot allowed us to understand which set of aspect ratios was the most appropriate. In fact, training the model with a wrong set of aspect ratios can totally vanish the work, producing bad precision and accuracy results.

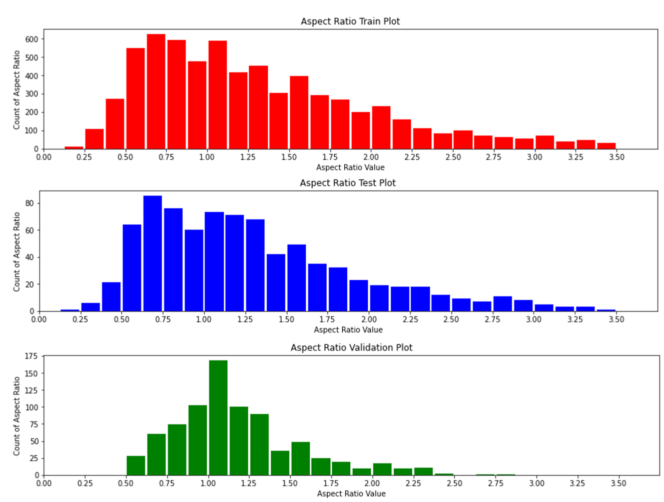

Fig. 3: Aspect ratios distribution respectively from the top for train, test and validation set.

As shown in Fig. 3, the majority of the bounding boxes have an aspect ratio between 0.5 and 3.0. Therefore, we used this information accordingly and trained our models with a set of aspect ratios in this range, with a 0.5 increase between each aspect ratio. Using a smaller step, while producing better results, can heavily slow down the training process.

Similarly, we produced another histogram related to bounding boxes, this time showing the distribution of the ground truth bounding boxes area, as shown in Fig. 4. This information allowed us to create a set of scales for the bounding box generators parameter. Using well thought out aspect ratios and scales parameters can really make a difference between a good performing models and poor one.

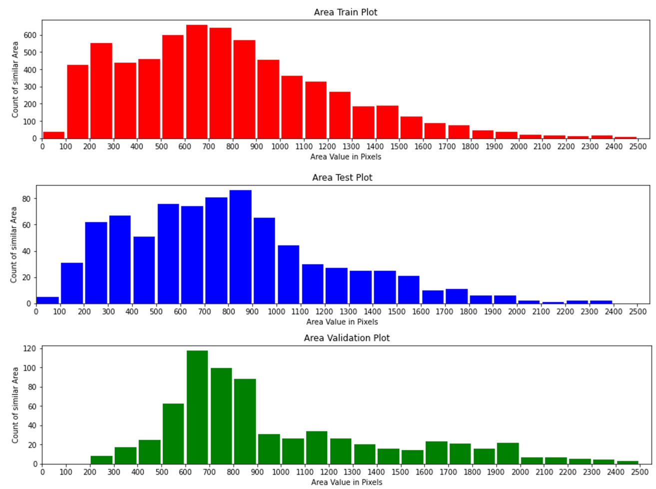

Fig. 4: Aspect ratios distribution respectively from the top for train, test and validation set.

C. Training Settings

The training process for the various models was carried out using Google Colab T4 16 GB GPU instances. Given the limited amount of video memory available with such GPU, we had to train the models with a small batch size; adding to this, we couldn't train the model for a prolonged period since Colab limits the amount of time a runtime can stay alive. When training the models, their base configuration is similar to that used to train on the COCO 2017 dataset. For the unambiguous comparison of the selected models, the total number of training steps was set to 25.000. The SSD model was trained using a manual step learning rate, which decreases the learning rate as the training progresses, and a batch size of 2. The Faster R-CNN model was instead trained with a cosine decaying learning rate and a batch size of 2, the same applies for the Efficientdet model, except that the latter was trained with a batch size of 4 (since is a very memory efficient model).

Furthermore, for all the models, we firstly trained a base version with what we thought were the optimal parameters. Later, a second version for each model was trained, trying to improve on the previous results thanks to the addition of data augmentantion options, changes in the learning rate and in others hyperparameters. The Table I summarize some of the parameters used for each model.

Model	Augmentantions	Batch size	Type of LR
SSD ResNet50 V1 FPN v1	Random Hor. Flip Random Adj. Contrast	2	Manual step
SSD ResNet50 V1 FPN v2	Random Hor. Flip Random Adj. Contrast Random Adj. Brightness Random Pixel value scale	2	Manual step
Faster R-CNN ResNet50 v1	Random Hor. Flip Random Adj. Contrast	2	Cosine decay
Faster R-CNN ResNet50 v2	Random Hor. Flip Random Adj. Contrast Random Adj. Brightness Random Pixel value scale	2	Cosine decay
Efficientdet D0 v1	Random Hor. Flip Random Adj. Contrast	4	Manual step
Efficientdet D0 v2	Random Hor. Flip Random Adj. Contrast Random Adj. Brightness Random Pixel value scale	4	Cosine decay

TABLE I: Summary of training settings

D. Evaluation Metrics

To evaluate and compare the performance of the selected neural networks, the COCO detection metrics were used. Specifically, the mAP (mean average precision) and mAR (mean average recall) were the metrics considered. For mAP a predefned threshold value for Intersection over Union (IoU) equal to 0.5 was used.

The Intersection Over Union (IOU) is a measure based on Jaccard Index that evaluates the overlap between two bounding boxes. It requires a ground truth bounding box and a predicted bounding box. By applying the IoU we can tell if a detection is valid (True Positive) or not (False Positive). IoU is given by the overlapping area between the predicted bounding box and the ground truth bounding box divided by the area of union between them.

We chose to focus on the mAP with an $IoU = 0.5$ metric because for the task at hand it is not fundamental to perfectly fit a bounding box on the stenosis location. In fact, even if a box covers only 50% of the ground truth box, it still guides the specialist eyes to the stenosis location; the specialist can then proceed with a more precise analysis of that area.

IV. Results and Discussion

In this chapter the results of the trained detectors models are presented. Firstly, from the Table II, we can see how much computational heavier the Faster R-CNN model is compared to the other models. Furthermore, since we trained the Efficientdet model using the same number of training steps but with two times the batch size, the number of epochs value is also doubled.

Model	Train time	Epoch
SSD ResNet50 V1 FPN v1	2.53 hours	7.5
SSD ResNet50 V1 FPN v2	2.69 hours	7.5
Faster R-CNN ResNet50 v1	4.16 hours	7.5
Faster R-CNN ResNet50 v2	4.32 hours	7.5
Efficientdet D0 v1	2.04 hours	15
Efficientdet D0 v2	2.05 hours	15

TABLE II: Training summary

As shown in Fig. 5, the model that scores the best precision value is the SSD v1 with a precision of 85%, closely followed by the SSD v2 with 83%. The Efficientdet v2 and Faster R-CNN v2 follow with a slightly lower precision value, respectively 79% and 75%. While the last two models, Efficientdet v1 and Faster R-CNN v1, settle on a much lower value, respectively 48% and 52%, clearly highlighting a problem with the training parameters. However, the Faster R-CNN v1 doesn't appear to be stalling and could probably have benefited from a longer training.

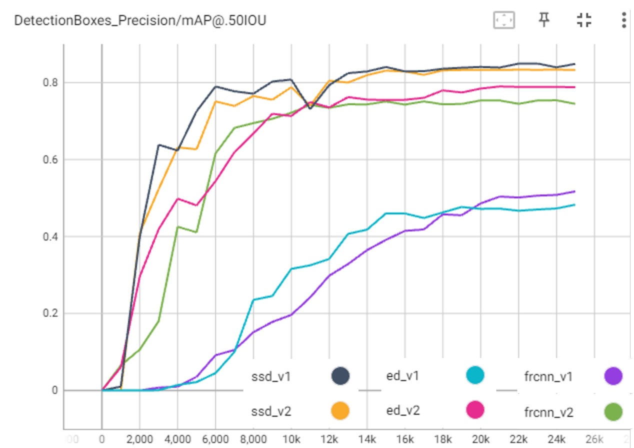

Fig. 5: Precision over the train steps for all models

On the other hand, in Fig. 6, the recall values present more or less the same situation as seen for the precision plot. We can see how the SSD models scores higher with a recall value of 44%) for both versions. On lower recall values we can find the Efficientdet models with a recall of 30% and 38% respectively for version 1 and version 2. Lastly, the Faster R-CNN models settle on a low recall value of 24% for both versions.

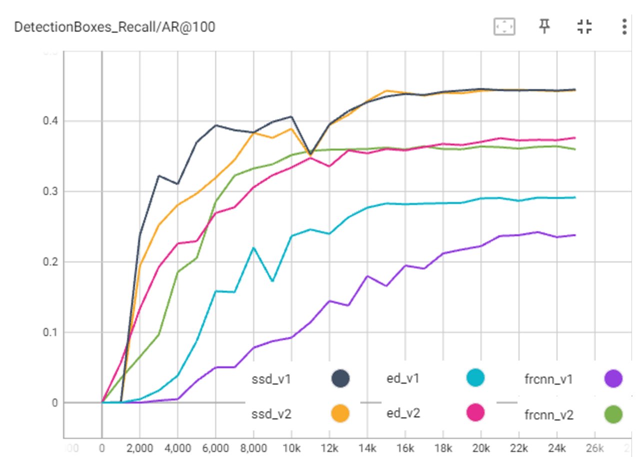

Fig. 6: Recall over the train steps for all models

In Table III, a summary of the results across the different models is presented.

In Fig. 7, some inference examples is shown for the best performing models. The images in the second column is a clear example of a very hard to detect stenosis, in fact, there's a really poor contrast with the background, making the stenosis hard to detect even for the human eye.

Model	Inf. Time	mAP IoU=50	mAR
SSD ResNet50 V1 FPN v1	0.52 s	85%	44%
SSD ResNet50 V1 FPN v2	0.52 s	83%	44%
Faster R-CNN ResNet50 v1	0.40 s	52%	24%
Faster R-CNN ResNet50 v2	0.42 s	74%	24%
Efficientdet D0 v1	0.49 s	48%	30%
Efficientdet D0 v2	0.51 s	79%	38%

TABLE III: Models results summary

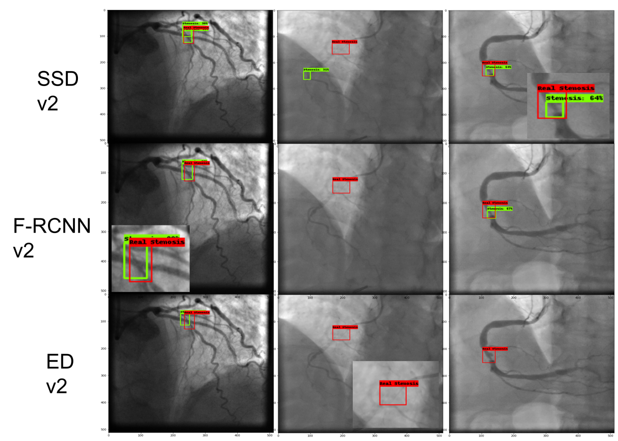

Fig. 7: Some inference examples

V. Conclusion

In this paper we worked with three different object detection architectures, SSD ResNet50 V1 FPN, Faster R-CNN ResNet50 V1 and Efficientdet D0 for the detection of stenosis in coronary angiography images. The main purpose was to compare how different models performed on the same problem. The obtained results for some models, namely Faster R-CNN v1 and Efficientdet v1, are clearly too low for a practical use and they highlight problems with the training parameters used. Other models, such as SSD v2 and Efficientdet v2, performs way better. Another important aspect is the inference time; while the inference times obtained are not too big for a single shot inference on an image, they are too high if the planned use for these detectors is real-time inference during the coronary angiography.

A possible future development could be to increase the training steps using machines with more powerful hardware, in fact, training on Google Colab limited the amount of processing time at our disposal. Another future point could be trying to decrease the inference time in order to allow real-time detection.

References

[1] GBD 2017 Causes of Death Collaborators. Global, regional, and national age-sex-specifc mortality for 282 causes of death in 195 countries and territories, 1980–2017: a systematic analysis for the Global Burden of Disease Study 2017. Lancet (London, England) 392(10159), 1736–1788 (2018).

[2] Virani, S. S. et al. Heart disease and stroke statistics-2020 update: a report from the American heart association. Circulation 141(9), e139–e596 (2020).

[3] Jensen, R. V., Hjortbak, M. V. Bøtker, H. E. Ischemic Heart Disease: An Update. Semin. Nucl. Med. 50, 195–207 (2020).

[4] Knuuti, J. et al. 2019 ESC guidelines for the diagnosis and management of chronic coronary syndromes. Eur. Heart J. 41(3), 407–477 (2020).

[5] Saraste, A. Knuuti, J. ESC 2019 guidelines for the diagnosis and management of chronic coronary syndromes: recommendations for cardiovascular imaging. Herz. Springer Medizin 45(5), 409–420 (2020).

[6] Collet, C. et al. Coronary computed tomography angiography for heart team decision-making in multivessel coronary artery disease. Eur. Heart J. 39(41), 3689–3698 (2018).

[7] Janssen, J. P. et al. New approaches for the assessment of vessel sizes in quantitative (cardio-)vascular X-ray analysis. Int. J. Cardiovasc. Imag. 26(3), 259–271 (2010).

[8] Gao, Y. Sundar, H. Coronary arteries motion modeling on 2D X-ray images. In Medical Imaging 2012: Image-Guided Procedures, Robotic Interventions, and Modeling (eds Holmes, D. R., III. Wong, K. H.) 83161A (SPIE, 2012).

[9] Wan, T. et al. Automated identifcation and grading of coronary artery stenoses with X-ray angiography. Comput. Methods Progr. Biomed. 167, 13–22 (2018).

[10] Fang, H. et al. Greedy sof matching for vascular tracking of coronary angiographic image sequences. IEEE Trans. Circuits Syst. Video Technol. 30, 1466–1480 (2020).

[11] M'Hiri F. et al. Vesselwalker: coronary arteries segmentation using random walks and Hessian-based vesselness flter. In Proceedings: International Symposium on Biomedical Imaging, 918–921 (2013).

[12] M'Hiri F. et al. Hierarchical segmentation and tracking of coronary arteries in 2D X-ray Angiography sequences. In Proceedings: International Conference on Image Processing, ICIP. IEEE Computer Society, vol. 2015-December, 1707–1711 (2015).

[13] M'Hiri, F. et al. A graph-based approach for spatio-temporal segmentation of coronary arteries in X-ray angiographic sequences. Comput. Biol. Med. 79, 45–58 (2016).

[14] Ren, X. Malik J. Learning a classifcation model for segmentation. In Proceedings of the IEEE international Conference on Computer Vision. Institute of Electrical and Electronics Engineers Inc., vol. 1, 10–17 (2003).

[15] Jo, K. et al. Segmentation of the main vessel of the lef anterior descending artery using selective feature mapping in coronary angiography. IEEE Access 7, 919–930 (2019).

[16] Nasr-Esfahani, E. et al. Segmentation of vessels in angiograms using convolutional neural networks. Biomed. Signal Process. Control 40, 240–251 (2018).

[17] Zreik, M. et al. A recurrent CNN for automatic detection and classifcation of coronary artery plaque and stenosis in coronary CT angiography. IEEE Trans. Med. Imag. 38(7), 1588–1598 (2019).

[18] Yang, S. et al. Deep learning segmentation of major vessels in X-ray coronary angiography. Sci. Rep. Nat. Res. 9(1), 1–11 (2019).

[19] Danilov, V.V., Klyshnikov, K.Y., Gerget, O.M. et al. Real-time coronary artery stenosis detection based on modern neural networks. Sci Rep 11, 7582 (2021).