Multi-model comparison of Cardiac Segmentation Model for Angiocardiography by Deep Learning

Hsiang-Wei Hu[1*], Wei-Ming Lin[2], Nai-Yun Tung[2], Ren-Syuan Huang[4], Chun-Yi Lee[5], [11*]

[1]Department of Biomedical Engineering, National Cheng Kung University, Taiwan

[2]International Academia of Biomedical Innovation Technology, Taipei, Taiwan

National Taiwan University Hospital,Taipei,Taiwan

[3]National Central University, Taoyuan, Taiwan

[4]National Tsing Hua University, Hsinchu Taiwan

[5]Department of Internal Medicine, National Cheng Kung University Hospital, Tainan, Taiwan

Abstract

Angiocardiography is a key evaluation of cardiac function, and the index of left ventricular ejection fraction (LVEF) can also help interpret cardiac hypertrophy, valve regurgitation, and regional myocardial systolic dysfunction to help medical decision making. However, it is difficult and time-consuming to calculate the LVEF, therefore imaging techniques become much more important for assistance.

In this research, three neural networks including DenseNet, EfficientNet and ResNet are introduced for cardiac area calculation. EfficientNet turns out to be the best model with 0.961 Mean Dice Accuracy. The other two models also show great performance with 0.958 using DenseNet, and 0.955 using ResNet.

With these network models, physicians can automatically select and calculate the area of the heart. With the time-series images of cardiac catheterization, the size of the entire cardiac systole can be determined automatically. Therefore, it can greatly help medical professionals to diagnose abnormal cardiac systole problems immediately.

Keywords: *Angiocardiography, convolutional neural networks, segmentation models, cardiac catheterization, left ventricular ejection rate*

Introduction

Left ventricular angiography is an evaluation of cardiac structure and function that is often performed during cardiac catheterization. In addition to obtaining the index of left ventricular ejection fraction(LVEF), it can also help interpret cardiac hypertrophy, valve regurgitation, and regional myocardial systolic dysfunction to assist clinicians. Treatment and judgment of opening blood vessels. Although left ventricular angiography has been a well-established and internationally recognized detection method, circle selection based on end-diastolic volume (EDV) and end-diastolic volume (ESV) requires sufficient experience and is time-consuming. The use of deep learning methods to estimate 2D ultrasound images has been used to evaluate LVEF [3], and the method for fully automatic volume and EF measurement has been achieved. [4] Another research group also achieved fully automated real-time estimation and proposed improvements towards addressing ultrasound acquisition errors such as apical foreshortening, making it possible to achieve continuous acquisition and measurement workflows in the clinic. [5]

In our research, We compare three segmentation models to calculate the cardiac area for cardiac catheterization, allowing clinicians to quickly calculate the left ventricular ejection rate (LVEF) and confirm the abnormal cardiac function.

Method

A. Data collection

In this study, we gathered 42 patients of cardiac whole volume heart x-ray imaging data from Chi Mei Medical Center. The total collected data included 1717 slices of cardiac images converted into jpg format x-ray images(Fig.1-A) with image size 512*512, then created mask images(Fig.1-B). Mask images were drawn by a group of annotators under the doctor's supervision, which later became binary masks for each image. We randomly divide data into training(95%) and validation(5%) in the training section.

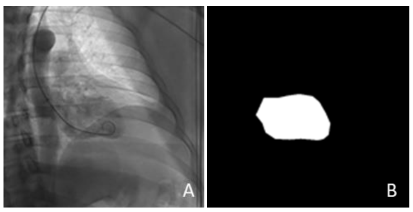


Fig.1 Shows X-ray image and mask image.

B. Data augmentation and preprocessing

Data augmentation is used in order to mimic the condition of a real medical image influenced by either patient position or individual difference. Following is the 7 steps, each of them has a probability of 0.3, and the final probability of augmentation is 0.8:

1. GridDistortion(num_steps=4,distort_limit=0.1)
2. ElasticTransform(alpha=2000,sigma=60)
3. Affine (scale=[0.9,1/0.9], translate_percent=±0.1, rotate=±5, shear=±5)
4. GaussNoise(var_limit=(0,100),mean=0)
5. Blur(blur_limit=7)
6. Downscale (reduce the image quality)(scale_min=0.5,scale_max=0.9)
7. RandomBrightnessContrast(brightness_limit=0.2,contrast_limit=0.2)

Preprocessing is used to normalize and enhance the x-ray image contrast by using CLAHE.

C. Models

We choose Unet as the main architecture for the segmentation task because it's the most commonly used segmentation model in medical images. Unet is composed of two major parts: encoder and decoder. The encoder part involves a series of downsample steps, and it's the conventional architecture for most CNN models. So we conducted a series of experiments to find out which CNN model is the most suitable model for the encoder of Unet in our task.

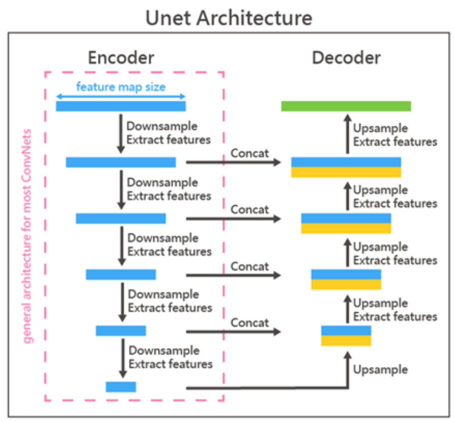
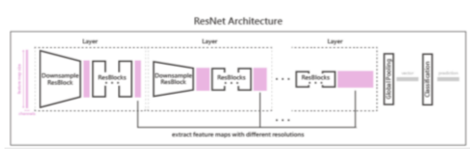
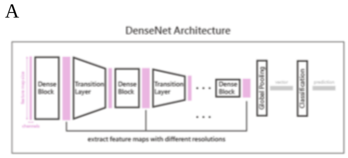


Fig.2 Shows original UNet architecture.

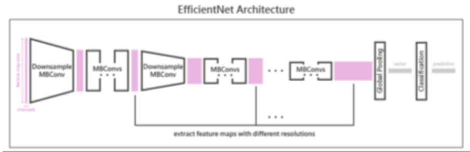
We have three candidate models in this study: ResNet-50, DenseNet-201, EfficientNet-B4. For fair comparison, we choose the subtypes of these models with similar numbers of parameters. Besides model structure itself, we also try to answer the question: Can we benefit from pretrained weights learned from Imagenet dataset even in medical image tasks? So, we further separate each model into two groups: pre-trained or non pre-trained for the discussion.



A



B



C

Fig.3 (A) Shows ResNet architecture, (B) DenseNet architecture, (C) EfficientNet architecture

So there are a total of six models for the exam. We ran each of them ten times to check the stability of the accuracy. And for fair comparison, we generate ten different seeds to set up the training process for every model.

ResNet

ResNet[2] is the most commonly seen CNN model. Its well known residual block alleviates the gradient vanishing problem while training and allows the model to be deeper. So, it's a robust baseline for performance comparison. We chose ResNet as the startup.

DenseNet

DenseNet[3] is also inspired by residual blocks and it directly concatenates all the outputs from previous layers and is called feature reuse. This structure not only prevents gradient vanishing problems but also pays attention to feature maps with different receptive fields. It may be beneficial for our task because each feature map with different receptive fields is important in x-ray images.

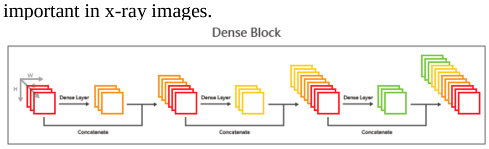


Fig.4 Feature map reuse

EfficientNet

The depth, width, and feature map resolution of EfficientNet[4] is delicately balanced between efficiency and accuracy. Its structure adopted inverted bottleneck MBConv[5][6], squeeze-and-excitation optimization[7] and activation function swish[8][9][10].

Inverted bottleneck MBConv preserves information after a non-linear activation function.

squeeze-and-excitation optimization helps model focus on import channels.

Swish activation function has better differential properties than RelU used in ResNet and DenseNet.

The structures above may all be beneficial for our task and require less calculation resources when interfering.

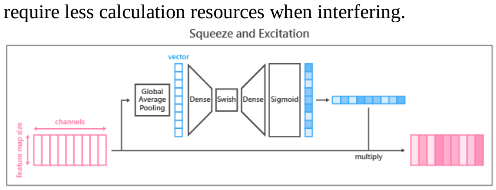


Fig.5 squeeze-and-excitation

D. Loss function

We use a mixture of Dice loss[11] and Binary cross entropy loss for the training.

Loss function(ground truth, prediction)

= DiceLoss(ground truth, prediction) * 1

+ BCELoss(ground truth, prediction) * 0.2

Dice loss provides an outline of what the mask should be like while BCE loss specifically focuses on whether each picture should be 0 or 1. By combining the two, the model could be improved and give prediction masks with more clear contour.

E. Performance validation

To evaluate the model's performance, we use Dice accuracy as a validation index.

binarized prediction = threshold(0.5, prediction)

Dice accuracy = 1-DiceLoss(ground truth, binarized prediction)

Dice accuracy could measure the similarity of the two contours like Dice loss, but 1 is the most accurate.

We calculated Dice accuracy with the validation dataset for twice, one with augmentation another without augmentation.

So, we could make sure it’s not overfitting one one of them. .

Results

We tested three models, ResNet-50, EfficientNet-B4, and DenseNet-201, with two conditions, pretrained and non-pretrained. In Fig.6, all the results show mean Dice accuracy under 0.940. The lowest means of Dice accuracy is EfficientNet-B4 with a pretrained model. It shows mean Dice accuracy = 0.961 in train condition, 0.961 in validation condition without augmentation.

		DenseNet		EfficientNet		ResNet	
		Mean Dice Accuracy	Standard deviation of Dice Accuracy	Mean Dice Accuracy	Standard deviation of Dice Accuracy	Mean Dice Accuracy	Standard deviation of Dice Accuracy
Non-pretrained	train	0.950	0.005	0.948	0.002	0.948	0.006
	validation	0.950	0.003	0.950	0.004	0.949	0.003
	validation _w/o_arg	0.953	0.003	0.953	0.004	0.952	0.003
pretrained	train	0.958	0.003	0.961	0.002	0.954	0.003
	validation	0.956	0.002	0.960	0.002	0.952	0.003
	validation _w/o_arg	0.958	0.002	0.961	0.001	0.955	0.003

Fig.6 Shows mean Dice accuracy in three models - DenseNet, EfficientNet, and ResNet - with pretrained/non-pretrained conditions.

Fig.7 shows trends in 30 epochs of each model; the pre-trained model converges fine in all six conditions.

Fig.8 shows the concentration of 10 runs in each model in validation with augmentation, EfficientNet with a pre-trained model shows lowest Dice accuracy and lowest Dice accuracy standard deviation in 10 runs. Pre-trained models do not always have the lowest Dice accuracy in our results.

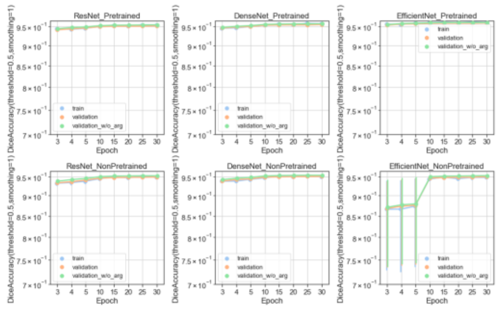


Fig.7 Comparison between the three models with or without pre-trained

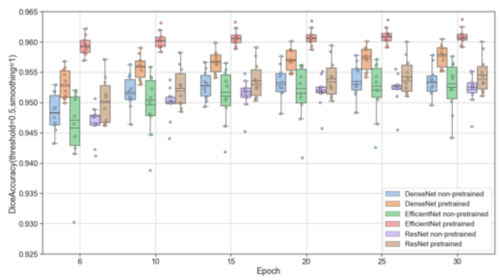


Fig.8 Comparison between the three models with or without pre-trained

Fig.4 shows the segmentation results of pre-trained EfficientNet. Left column is the preprocessed image. Right column is the prediction before binarization. Mid column is the prediction after binarization with a threshold 0.5, green color means true positive (correct prediction), red color means false positive (excess prediction) and blue color means false negative (not predicted).

已註解 [1]: 此部分可再針對圖中內容擴寫

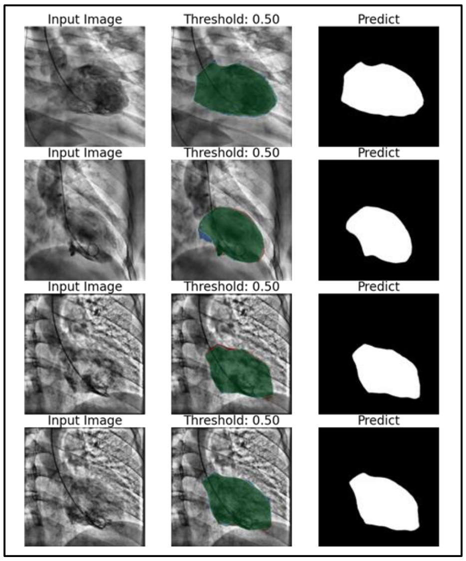


Fig.4 Segmentation results of pre-trained EfficientNet

Discussion

After all the experiments, we tried to give the explanation of the performance of ResNet, DenseNet, EfficientNet. EfficientNet is better than the other two models.

EfficientNet combines inverted bottleneck MBConv, Squeeze and excitation, Swish and well-balanced model depth, width, and feature map resolution.

ResNet is a model from 2015 and many models are based on it and more specialized. It's not the best one, but it still gives us an insight that we can use deep learning methods to work on the task.

DenseNet, which is each layer inside the dense block, is not extracting enough information since it gives a feature map with only a small number of channels. Maybe our task needs more channels to extract enough information. So, it didn't perform as well as we thought.

在於效能上的差異，訓練時運用 GPU 其 ResNet 模型訓練速度是最快的， 對於 EfficientNet 在結構上在 GPU 運行之下反而效率是最差，在實際進行 Infernce 時, EfficientNet 因其運算量少，在一般 CPU 的運作上，效益是最佳，能最快進行預測。

未來能延續心臟長軸計算心臟射出率公式，進而未來達到自動化偵測 EF 的系統

Conclusion

As a result, EfficientNet-b4 with pre-trained weights is the most suitable for the task among the three models. In the future, the clinical staff can automatically select and calculate the area of the heart. With the time-series images of cardiac catheterization, the size of the entire cardiac systole can be automatically determined. The problem of abnormal cardiac systole can be diagnosed in real-time.

References

[1] https://arxiv.org/pdf/1512.03385.pdf (先放連結)

[2] https://arxiv.org/pdf/1608.06993.pdf

[3] https://arxiv.org/pdf/1905.11946.pdf

[4] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. CVPR, 2018

[5] Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., and Le, Q. V. MnasNet: Platform-aware neural architecture search for mobile. CVPR, 2019.

[6] Hu, J., Shen, L., and Sun, G. Squeeze-and-excitation networks. CVPR, 2018.

[7] Ramachandran, P., Zoph, B., and Le, Q. V. Searching for activation functions. arXiv preprint arXiv:1710.05941, 2018.

[8] Elfwing, S., Uchibe, E., and Doya, K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. Neural Networks, 107:3–11, 2018.

[9] Hendrycks, D. and Gimpel, K. Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415, 2016.

[10] https://arxiv.org/pdf/1606.04797.pdf

[11] Abdi AH, Luong C, Tsang T, Allan G, Nouranian S, Jue J, et al. Automatic Quality Assessment of Echocardiograms Using Convolutional Neural Networks: Feasibility on the Apical Four-Chamber View. IEEE Trans Med Imaging. 2017;36(6):1221-30.

[12] Zhang J, Gajjala S, Agrawal P, Tison GH, Hallock LA, Beussink-Nelson L, et al. Fully Automated Echocardiogram Interpretation in Clinical Practice. Circulation. 2018;138(16):1623-35.

[13] Smistad E, Ostvik A, Salte IM, Melichova D, Nguyen TM,

Haugaa K, et al. Real-Time Automatic Ejection Fraction and Foreshortening Detection Using Deep Learning. IEEE Trans Ultrason Ferroelectr Freq Control. 2020;67(12):2595-604.