Analyse pyramidales d'images globale avec une architecture U-Net hautement supervisée

Michele BONA Clémence Gaillotte Tristan Michel Damien Djomby Enzo Masson
Ecole de Technologie Supérieur
{michele.bona.1, clemence.gaillotte.1, tristan.michel.1, damien.djomby.1, enzo.masson.1}@ens.etsmtl.ca

Décembre 2021

Abstract

In this paper we present SubUNet, a new architecture for medical image segmentation (MRIs). Our model is inspired by the UNet architecture and adds the Pyramid Pooling Module (PPM) of the PSPNet architecture. The addition of this module serves mainly to add a global observation of our image by increasing the receptive field of the model in order to detect the various parts of the heart by their shape and according to the organs which surround it. Thus, by combining the precision of Unet for the details of the images and the PPM we manage to detect the right ventricle, the left ventricle and the myocard on MRIs of hearts with a better precision than UNet.

Résumé

Dans ce papier nous présentons SubUNet, une nouvelle architecture pour la segmentation d'images médicales (IRMs). Notre modèle s'inspire de l'architecture de UNet et ajoute le Pyramid Pooling Module (PPM) de l'architecture PSPNet. L'ajout de ce module sert principalement à ajouter une observation globale de notre image et d'augmenter le champ réceptif de notre modèle afin de pouvoir détecter les différentes parties du coeur par leur forme et en fonction des organes qui les entourent. Ainsi, en combinant la précision de UNet pour les détails des images et le PPM nous arrivons à détecter le ventricule droit, le ventricule gauche et la myocarde sur des IRMs de coeurs avec une meilleure précision que UNet.

1 Introduction

1.1 Problématiques

Lors de ces dernières décennies, l'imagerie par résonance magnétique (IRM) est devenue un examen essentiel pour l'étude du coeur humain. Cependant, en raison des caractéristiques des images IRM cardiaques et de la grande variabilité des images entre les patients, le problème de l'identification automatique des différentes parties du coeur dans les IRMs s'est posé et est toujours très étudié.

D'un autre côté la recherche concernant le traitement d'images par des réseaux de neurones avance de plus en plus vite, et le sujet de la segmentation automatiques des IRMs avance avec elle. Ainsi on peut se demander comment identifier automatiquement sur des IRMs de coeurs humains le ventricule droit, le ventricule gauche et la myocarde.

Ce type de segmentation demande des connaissances techniques concernant le coeur humain, et le développement d'un réseau de neurones la faisant automatiquement permettrait d'aider à l'analyse des IRMs et d'accélérer les diagnostics des médecins.

1.2 Contributions

Comme nous allons voir dans la section Méthode, nous nous sommes inspiré des réseaux de neurones les plus efficaces et utilisés, particulièrement U-Net. Après les avoir implémentés et avoir créé notre modèle nous avons remarqué que nous avions une meilleure segmentation que U-Net, et une précision très satisfaisante pour la segmentation du ventricule droit qui pose en généralement le plus de problèmes sur ce jeu de données.

2 État de l'art

L'apprentissage automatique est aujourd'hui au centre de beaucoup de sujets de recherche sur l'imagerie médicale en raison de son importance et des progrès importants que cela pourrait apporter dans le milieu médical. C'est donc l'objet de nombreux challenges d'apprentissage automatique, qui permettent à des chercheurs de proposer des modèles de plus en plus performants pour ces challenges.

2.1 Méthodes de segmentation d'imagerie médicale

Il existe notamment beaucoup de modèles répondant aux problèmes de la segmentation d'imagerie médicale et nous allons nous concentrer sur quelques modèles et fonctions de coûts s'étant démarqués des autres.

2.2 U-Net

C'est le cas du réseau U-Net connu pour son architecture en forme de U, composé d'une partie "encodeur" et d'une partie "décodeur".Il s'agit d'un réseau entièrement convolutif qui est basé sur un réseau fully connected (FCN)

U-Net a été créé par *Olaf Ronneberger, Philipp Fischer et Thomas Brox* en 2015 dans le cadre de leur participation et victoire au challenge ISBI (challenge de segmentation pour l'imagerie médicale) en Novembre 2015. Le score de leur modèle a dépassé de plus de 10% les scores des autres modèles. Le réseau U-net a été présenté plus explicitement dans leur papier *U-Net : Convolutional Networks for Biomedical Image Segmentation* [8]. Depuis le réseau a été repris et légèrement modifié par de nombreuses équipes dans le cadre de la segmentation pour l'imagerie médicale.

En 2018, le réseau U-Net++ [5] conserve le principe d'un sous-réseau d'encodage puis d'un sous-réseau de décodage, une des principales modifications et la liaison entre ces deux sous réseaux avec des Skip connections ainsi qu'un usage de la supervision profonde.

Enfin en 2021, le réseau GA-Unet [4] est créé. Le modèle utilise le réseau de neuronne convolutionnel VGG16 comme encodeur du réseau U-Net. Les couches de pooling présentes dans U-Net sont remplacées par des opéarations d'Upsamplings.

2.3 PSPNet

Le réseau PSPNet a été créé en 2016 [9] dans le cadre d'un challenge de segmentation. Ce réseau n'est pas spécialement prévu pour l'imagerie médicale mais il obtient des résultats très performant pour la segmentation.

Ce réseau est composé d'une architecture encodeur-décodeur et contient des pyramid pooling modules (PPM). Ces modules permettent de donner un contexte global a l'image avant de faire une prédiction locale. Il s'agit d'une amélioration par rapport a ResNet [3] qui est un modèle ayant rencontré des difficultés à capturer le contexte global d'un image malgré un champ réceptif en entrée très grand.

2.4 Fonctions de coûts

Il existe de nombreuses fonctions de coûts mais certaines ont particulièrement démontrés leur efficacité dans les problème de segmentation d'imagerie médicale. Il existe la cross-entropie qui est très souvent utilisée. Cette fonction de coût fonctionne au niveau du pixel et est la base de nombreuses autres fonctions de coûts.

Une fonction de coût régulièrement utilisé en segmentation d'imagerie médicale est la Dice loss. Cette fonction permet de calculer la similarité entre deux images et permet une meilleure définition des contours.

Enfin il existe la Focal Loss introduite dans le cadre de la détection d'objets denses [10]. Il s'agit d'une variation de la Cross-entropie qui permet de mettre plus de poids sur les exemples ayant été difficilement classés et qui fonctionne très bien même si les classes ne sont pas équilibrées. Cette fonction de coût a généralement un bonne impact sur la vitesse de convergence.

3 Méthode

3.1 Données d'entraînement

Nos données d'entraînement sont 1208 images différentes d'IRMs de coeur sur différents patients (et 90 images de validation). Les images montrent un coeur selon différentes coupes transversales, toutes en différents niveaux de gris et de taille 256x256. Le seul pré-traitement que nous avons appliqué est d'équilibrer l'histogramme de niveau de gris des images.

3.2 Augmentation de données

Le corps humains est complexe et il y a une grande variabilité des placements et de la taille des organes. Cela se voit particulièrement lorsque l'on essaye de faire de la segmentation d'images de coeurs humain pour en extraire les différentes parties.
Afin d'éviter l'over-fitting sur notre petit jeu de données, nous avons mis en place une augmentation de données (Data Augmentation, en anglais) qui reste cohérente avec les possibilités que nous pouvons rencontrer dans le monde. Pour cela il a fallu prendre en compte différentes malformations qui peuvent atteindre les patients. Nous avons donc une dizaine de transformations possible, chacune avec une probabilité de 0.2 de s'appliquer. Cette valeur élevé permet d'entraîner notre modèle sur des données très variées. Les transformations possibles sont :

- un vertical flip (mirroir sur l'axe horizontal)
- un mirror horizontal flip
- une rotation avec un angle allant de -30° à 30° choisi aléatoirement
- un crop (Zoom), la taille du sous-cadre vaut entre 0 et 1/2 taille du cadre original, la position du sous-cadre dépend d'un bord choisi aléatoirement
- un padding (Ajout d'une marge noire)
- un z shift
- un brightness shift (changement de luminosité)
- un flou gaussien (qui nous a été très utile)
- un auto-contrast
- un changement de sharpness (d'une valeur entre 0 et 2)

3.3 Choix du modèle

Empiriquement, la recherche a largement prouvé que la structure UNet est l'une des plus efficaces pour la segmentation d'imagerie médicale. Pour conserver la taille de nos images d'entré pour la sortie du modèle, nous avons choisi d'ajouter un padding pour les convolutions, au lieu de faire un padding mirroir sur toutes nos images comme présenté par Ronneberger *et al.*, [8]. Et cela évite le zoom naturellement présent entre les features maps que on retrouve dans UNet, en effet nous ne voulions pas perdre des informations (même si ce n'est que certains pixels sur les bords, car cela pourrait être décisif pour reconnaître les différentes parties du coeur).

Suite à nos expérimentations nous avons des meilleurs résultats avec les modèles de UNet et de PSPNet (grâce au Pyramid Pooling Module). Ces résultat sont parfaitement cohérent car UNet est le meilleur pour la segmentation médicale, donc son encodeur et son décodeur sont excellent. De plus le module PPM de PSPNet permet d'avoir un context plus global dans notre détection et ainsi essayer de repérer des formes qui entour le coeur (par exemple situer le coeur par rapport aux poumons). Nous avons ajouté des modules Deeply-Supervised dans le décodeur de UNet pour essayer de controler l'apprentissage dans la bonne direction mais malheureusement leur efficacité n'est pas concluante, comme nous le présenterons dans la section 4.2.

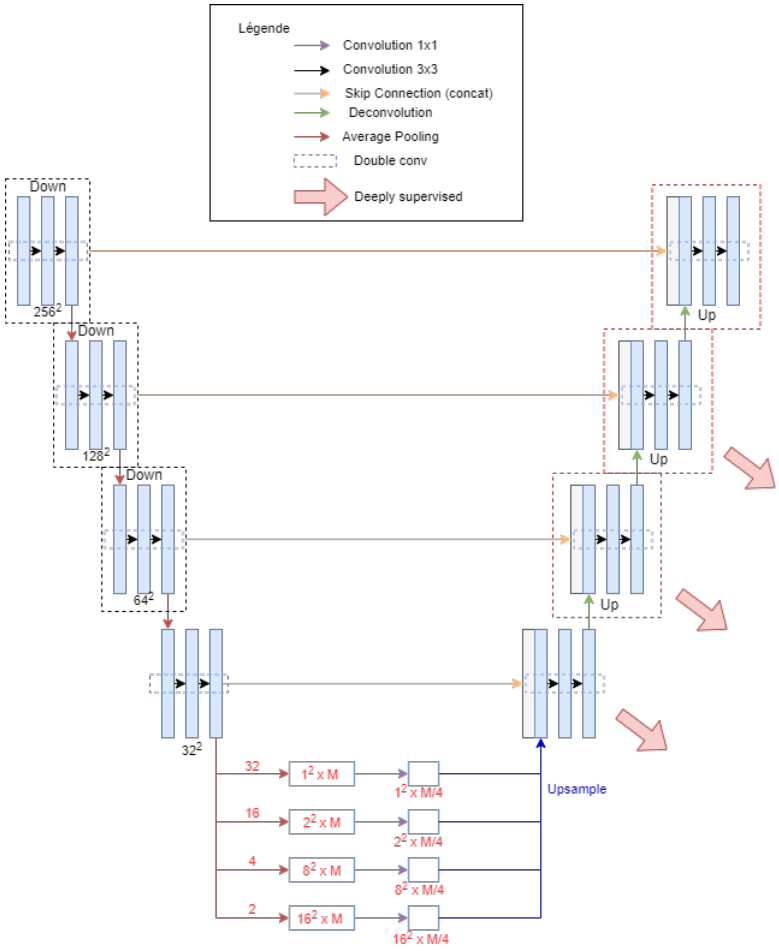

FIGURE 1: Architecture de notre modèle, basée sur un encodeur et un décodeur UNet avec Average pooling et dé-convolution, et un pyramidal pooling module.

Ainsi en combinant l'encodeur et le décodeur de UNet avec le PPM de PSPNet nous devrions avoir un modèle capable de faire une segmentation précise (avec les skip connections de UNet). Le tout en se basant sur une observation globale de notre image afin de pouvoir détecter les différentes parties du coeur par leur forme et en fonction des organes qui l'entoure. De plus d'après l'état de l'art et de nos expérimentations, nous avons utilisé des Average Poolings au lieux de Max Poolings dans l'encodeur UNet car cela fonctionne mieux par la suite avec le module PPM. La section 4.2 présente aussi les aventage de Avg Pooling. Ce modèle, présenté en figure 1, permet donc une segmentation avec cohérence d'environnement en prenant les avantages de UNet et du PPM.

3.4 Entraînement du réseau

Nous avons entraîné notre modèle avec une taille de batch de 56 images. Nos recherches ont montré qu'un grand nombre d'images par batch donnait de meilleurs résultats. Pour la mis-à-jour de nos paramètres nous avons choisi d'utiliser la politique de 'poly' taux d'apprentissage où le taux d'apprentissage de l'epoch est égale au taux d'apprentissage de départ de $3,15 \times 10^{-4}$ multiplié par $(1 - \frac{iter}{max_iter})^{power}$. Pour la valeur de la puissance, nous avons pris la même valeur que celle présenté dans le papier de PSPNet, soit $0,9$. Concernant la fonction de coût, nous avons utilisé une combinaison du coefficient de Dice et de Focal loss $loss(X, Y) = \alpha \cdot FocalLoss_{\gamma}(X, Y) + (1 - \alpha) \cdot DiceLoss(X, Y)$, avec $\alpha = 0,233$ et $\gamma = 1,977$.

Inspiré par Ye *et al.*, [11], nos branches profondément supervisées sont une successions de Convolution transposée 2D permettant de remonté à la taille d'origine de l'image d'entrée. Nous calculons ensuite l'erreur de chaque prédiction (branches principale et supervisées) par rapport à la vérité terrain et nous sommons ensuite ces erreurs. La rétro-propagation est calculé à partir de ce résultat et propagé à tous le modèle.

4 Résultats

4.1 Résultats sur le challenge

Comparatif aux autres modèles Lors de nos recherches préalables, nous avons effectué un grid search afin de déterminer quels étaient les modèles et les fonctions de coût les plus prometteurs pour le défi de segmentation. Nous avons implémenté les modèles SegNet, PSPNet (avec un simple encodeur au lieu de ResNet), UNet ainsi qu'une adapation de VGG permettant la segmentation. Nous avons aussi étudier différentes fonctions de coût dont les recherches et les résultats seront présenté dans la section Étude d'ablation pour la fonction de coût hybride. Le tableau 1 présente les résultats obtenus pour ce gridsearch. Sans surprises, UNet arrive en première position mais PSPNet donne aussi de bon résultats alors que ce dernier avec été implémenté avec un encodeur et un décodeur beaucoup plus simple que celui décris par Zhao *et al.*, [9]. Ainsi nous est venu l'idée d'utiliser le Pyramid Pooling Module au sein d'un UNet.

Modèle	Fonction de coût	Mean IoU(%)	Mean Dice(%)	Mean Soft Dice(%)
VGG	focal tversky loss + focal loss	65,92	77,81	19,55
SegNet	jaccard loss + focal loss	77,18	86,33	19,37
PSPNet	ce loss	*79,44*	*87,86*	*20,35*
UNet	jaccard loss	**85,52**	**92,01**	**92,29**

TABLE 1: Meilleurs résultats obtenus après le grids earch pour chaque modèle, spécifiant la fonction de coût ayant donné ces meilleurs résultats.

Vitesse de convergence Comme nous pouvons le voir sur la figure 2, SubUNet converge plus vite que UNet lorqu'il est entrainé avec notre module profondément supervisé. Néanmoins, la section 4.2 montrera que nous avons obtenus de meilleurs résultat sans ce module mais avec une vitesse de convergence plus faible.

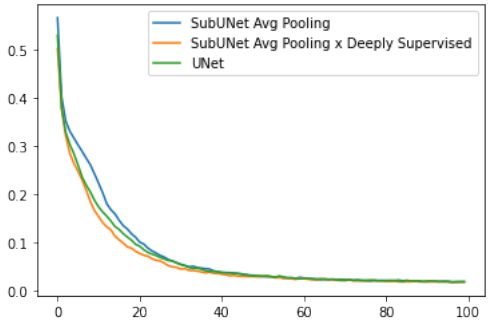

FIGURE 2: Comparaison des convergence de la fonction de coût sur le dataset d'entraînement pour notre baseline et notre modèle avec et sans module profondément supervisé.

Améliorations visuelles Comme le montre la figure 3, notre modèle permet une meilleur prédiction du ventricule droit qui est, selon nous, la classe la plus dure à prédire. Puisqu'elle peut aussi être la classe prépondérante sur certaines images, cette dernière peut baisser de façon considérable le score d'un modèle ayant du mal à généraliser. Nous remarquons que UNet a plus de mal a prédire le ventricule droit. Ce problème de prédiction sur le ventricule droit avec UNet est un problème que nous avons indentifié dés le début du challenge et est le principal point que nous avons essayer de résoudre. Ainsi, à l'issue de nos recherches, le modèle proposé a généralement une prédiction plus proche de la ground truth pour ce ventricule.

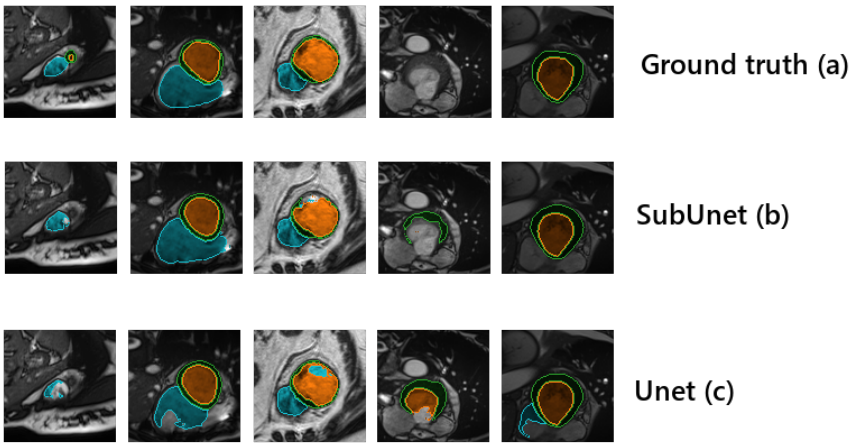

FIGURE 3: Amélioration visuelles sur le dataset de notre modèle (b) comparé à notre baseline(c). Ce figuré présentre le ventricule gauche en *orange*, la myocarde du ventricule gauche en *vert* et le ventricule droit en *bleu*. Les deux modèles, SubUNet et notres baseline ont été entraîné dans les mêmes conditions.

4.2 Étude de la robustesse de notre méthode

Étude d'ablation pour SubUNet Pour étudier les performances de notre modèle, comparé à notre baseline, nous avons effectué différents entraînements avec différentes configurations, notamment le type de pooling utilisé et l'usage ou non de notre module profondément supervisé. Comme listé dans le tableau 2, l'average pooling fonctionne mieux que le max pooling, ce qui rejoint les résultats obtenus par Zhao *et al.*, [9]. On remarque aussi que notre module profondément supervisé n'a pas donné de si bons résultats, ce qui peut s'expliquer par une mauvaise implémentions de notre part ou que envoyons un gradient trop fort sur les branches annexes, baissant ainsi la précision dans la branche principale.

Method	Mean IoU(%)	Mean Dice(%)	Mean Soft Dice(%)
PSPNet	79,62	88,20	90,50
UNet	84,77	91,49	93,10
SubUNet MP	85,10	91,75	93,41
SubUNet MP x Deeply Supervised	85,13	91,78	93,31
SubUNet AP x Deeply Supervised	85,50	92,02	93,58
SubUNet AP	**85,63**	**92,06**	**93,65**

TABLE 2: Étude de notre modèle selon différents paramètres. 'MP' et 'AP' signifient respectivement 'Max Pooling' et 'Average Pooling' et 'DP' correspond à notre module de supervision profonde. Nos résultats ont été obtenus sur le jeu de données de validation après un entraînement avec nos meilleurs hyperparamètres (détaillés en section 4.2) et avec augmentation de données sur 400 epochs.

Étude d'ablation pour la fonction de coût hybride Comme décrit dans la section 4.1, nous avons effectué un gridsearch pour étudier quelles fonctions de coût et quels modèles baseline donnaient les meilleurs résultats. Nous avons donc pu déterminer que pour ce challenge, le couple $DiceLoss + FocalLoss$ permettaient une très bonne performance. L'idée principale était de voir si la combinaison d'une fonction de coût dites 'ensembliste' (qui compare la prédiction et la vérité terrain comme des ensemble) et d'une fonction de coût pixel à pixel (pixel-wise) donnerait de meilleurs résultats. Le résultat de ce gridsearch était évident, une combinaison donne généralement de légèrement meilleurs résultats. Le tableau 3 présente les résultats obtenus par ablation sur la fonction de coût $Dice + Focal$ avec notre modèle SubUNet.

Method	Mean IoU(%)	Mean Dice(%)	Mean Soft Dice(%)
Focal loss	81,29	89,17	90,39
Dice loss	85,59	91,78	93,32
Dice loss + Focal loss	**85,75**	**92,08**	**93,68**

TABLE 3: Étude par ablation des fonctions de coût utilisées. Nos résultats on été obtenus sur notre modèle SubUnet avec nos meilleurs hyperparamètres et augmentation de données sur 200 epochs. La combinaison $Dice + Focal$ a été obtenue avec un $\alpha = 0,5$. La section 4.2 présente la valeur optimale que nous avons obtenu pour ce paramètre.

Hyperparamètres Pour déterminer les hyperparamètres de notre modèle, nous avons effectué un random search sur les paramètres suivants : la taille de batch, le α et le γ de notre fonction de coût présenté dans la section Entraînement du réseau ainsi que le taux d'apprentissage. Pour notre random search, nous avons effectué 200 entraînements de 400 epochs chacun, le taux d'apprentissage

a été tiré selon une distribution normale puis a été amené à la puissance 5 pour obtenir de petites valeurs. Soit r une valeur aléatoire entre 0 et 1, $lr = r^5$. Le tableau 4 présente les résultats de cette recherche et les meilleurs paramètres que nous avons obtenu et utilisé pour l'entraînement de nos modèles. Néanmoins il semblerait que des batchs encore plus grand auraient pu donner de meilleurs résultat mais nous nous sommes retrouvé face à un problème technologique : au-delà de 60 images par batchs, nous arrivions à la limite de RAM de nos cartes graphiques (nos modèles ont été entraîné sur une carte Nvidia A100 40Go par entraînement).

Batch	α	γ	lr	mean iou(%)	mean dice(%)	mean soft dice(%)
$[1; 100]$	$[0; 1]$	$[0; 4]$	$[0; 1]$			
63	0,054	1,351	$1,52 \times 10^{-2}$	87,53	93,24	94,70
67	0,461	2,676	$7,15 \times 10^{-4}$	88,12	93,60	94,95
62	0,879	3,604	$3,93 \times 10^{-4}$	88,54	93,82	95,09
73	0,490	1,737	$6,29 \times 10^{-4}$	88,91	94,05	95,38
56	**0,233**	**1,977**	$3,15 \times 10^{-4}$	**89,35**	**94,28**	**95,47**

TABLE 4: Ce tableau présente les meilleurs résultats obtenus lors de notre random search. La première section présente les intervalles utilisé et la dernière ligne de la deuxième section présente nos meilleurs paramètres.

5 Conclusion

Finalement, notre modèle est assez performant pour le problème de segmentation d'imagerie médicale et présente un interêt certain. Nous avons amélioré U-Net en ajoutant le système de Pyramid Pooling Module (PPM) de PSPNet. Notre modèle donne une précision moyenne satisfaisante pour détecter le ventricule droit, le ventricule gauche et la myocade sur IRMs de coeurs humains.

Pour chaque paramètre, toutes nos données sont stockés avec leurs mesures associés. Il serait donc possible d'étudier statistiquement l'impact des hyperparamètres sur la performance afin de trouver expérimentalement les meilleurs valeurs pour les hyperparamètres.

Références

[1] Feature engineering : cinq conseils essentiels pour améliorer vos modèles IA
https://www.lemagit.fr/conseil/Feature-engineering

[2] Les cinq métiers les plus menacés par l'intelligence artificielle
https://www.lesechos.fr/tech-medias/intelligence-artificielle/les-cinq-metiers-les-plus-menaces-par-lintelligence-artificielle-137080

[3] Deep Residual Learning for Image Recognition
urlhttps ://arxiv.org/abs/1512.03385

[4] *Amrita Kaur, Lakhwinder Kaur et Ashima Singh* GA-UNet : UNet-based framework for segmentation of 2D and 3D medical images applicable on heterogeneous datasets
https://link.springer.com/article/10.1007/s00521-021-06134-z

[5] UNet++ : A Nested U-Net Architecture for Medical Image Segmentation
https://link.springer.com/chapter/10.1007/978-3-030-00889-5_1

[6] ENet : A Deep Neural Network Architecture for Real-Time Semantic Segmentation
https://arxiv.org/abs/1606.02147

[7] FastVentricle : Cardiac Segmentation with ENet
https://link.springer.com/chapter/10.1007/978-3-319-59448-4_13

[8] U-Net : Convolutional Networks for Biomedical Image Segmentation
https://link.springer.com/chapter/10.1007%2F978-3-319-24574-4_28

[9] Pyramid Scene Parsing Network
https://arxiv.org/pdf/1612.01105.pdf

[10] Focal Loss for Dense Object Detection
urlFocal Loss for Dense Object Detection, Tsung-Yi Lin, 2017

[11] Multi-Depth Fusion Network for Whole-Heart CT Image Segmentation
urlhttps ://ieeexplore.ieee.org/abstract/document/8642875