



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

Avaliação de Eficiência de Segmentador de Imagens Cardíacas

Rodrigo Ferreira Guimarães

Monografia apresentada como requisito parcial
para conclusão do Curso de Engenharia de Computação

Orientador
Prof. Dr. Díbio Leandro Borges

Brasília
2021

Dedicatória

Eu dedico este trabalho aos meus pais, Elzimar e Sandro, e à minha irmã, Emily, por sempre me apoiarem nos meus desafios e nas suas resoluções, desde uma simples inspiração até a fomentação de *insights*. Dedico também àqueles que precisam de avanços na medicina cardíaca, mesmo que por não-médicos, para melhorar e manter suas vidas: espero que esta pesquisa lhes seja útil.

Agradecimentos

Agradeço aos meus pais, Elzimar e Sandro, por me proporcionarem a vida e a experiência livre para a evolução. Me deram todas as condições possíveis para o aprendizado da vida, fui, sou e serei eternamente grato por tudo que temos. A Emily, minha irmã, toda a amizade e companheirismo que possa ser escrito não seria suficiente e nem faria jus.

Em segundo lugar, agradeço ao Prof. Dr. Díbio Leandro Borges por todas as orientações que me foram passadas direta e indiretamente, pela calma, evolução lógica e vasto conhecimento. Conhecendo-o desde o meu segundo semestre, admiro a maneira que ministra suas aulas, o entendimento sobre a educação e a interdisciplinaridade de diversos assuntos.

Agradeço, também, à Universidade de Brasília (UnB) pelo conhecimento teórico, prático e humano que tive a oportunidade de conhecer e aprender. Valendo para servidores, outros estudantes e professores. Em especial destaque o professor Alexandre Zaghetto, que esteve presente desde o início desta minha jornada, como professor, orientador, parceiro de projetos e amigo.

Agradeço aos meus amigos Pedro e Camila, que estiveram ainda mais próximo mesmo que à distância, por me instigarem, me incentivar e me acompanhar às suas próprias maneiras; foram essenciais.

Enfim, agradeço a todos que estiveram presentes, de alguma forma, na minha vida, pois fazem parte do caminho que trilhei e que continuarei trilhando.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES), por meio do Acesso ao Portal de Periódicos.

Resumo

A identificação das estruturas cardíacas a partir de Imagem de Ressonância Magnética (IRM) ajuda a salvar vidas. Este processo exige tempo e delicadeza por parte dos especialistas que as indicam. Dada esta dificuldade e complexidade, o uso deste conhecimento por técnicas computacionais torna-se importante, vantajoso e vital. Por este motivo, este trabalho avalia a eficiência de Rede Neuronal Completamente Convolutiva otimizada para a segmentação de imagens cardíacas providas por IRM. Esta análise foi conduzida a partir da base de dados *Automated Cardiac Diagnosis Challenge (ACDC) 2017* (por caracterizar a cavidade do ventrículo direito, o miocárdio e a cavidade do ventrículo esquerdo) e possui o *Dice Similarity Coefficient (DSC)* como principal métrica de desempenho. Ponderou-se o impacto nos resultados para (i) distintas funções de perda, alternando entre a entropia cruzada, a *Dice Loss*, a *Focal Loss (FL)* (com fator $\gamma = 2$) e a combinação das duas últimas; e (ii) dimensões manuseáveis reduzidas. Com esta fundação, a redução das dimensões favoreceu a caracterização otimizada das regiões de interesse e a obtenção de DSC superior a 0,95 ao final deste estudo. Utilizar a combinação de funções de perda mostrou-se como uma excelente alternativa ao conseguir unir as vantagens que cada uma possuía isoladamente. Isto posto, avaliações futuras incluem o comportamento deste segmentador manipulando outras bases de dados, evidenciando as porções das classes anotadas, empregando outros fatores γ à FL e outras ponderações na combinação das funções de perda.

Palavras-chave: Estruturas Cardíacas. Imagens de Ressonância Magnética. Aprendizagem de Máquinas. Segmentação de Imagens. Análise de Segmentação. Redes Neurais Artificiais. Funções de perda

Abstract

Recognition of cardiac multi-structures from Magnetic Resonance Imaging (MRI) helps to save lives, which requires time and delicacy from those experts who nominate them. Given this difficulty and complexity, the use of this knowledge by computational techniques becomes important, advantageous and vital. For this reason, this work evaluates the efficiency of a Fully Convolutional Neural Network (FCNN) optimized for segmentation of cardiac images provided by MRI. This analysis was conducted using images from Automated Cardiac Diagnosis Challenge (ACDC) 2017 dataset (which provide the right ventricular cavity, the myocardium and the left ventricular cavity as annotated categories) and has the Dice Similarity Coefficient (DSC) as the main metric. The impact on the results was evaluated for *(i)* distinct loss functions, alternating between Cross Entropy, Dice Loss, Focal Loss (FL) (with γ factor equals 2) and the combination of the last two of them; and *(ii)* reduced dimensions. With this foundation, the reduction of dimensions facilitated the optimized characterization of the regions of interest and the achievement of a DSC greater than 0.95 at the end of this study. Using the loss functions combined proved to be an excellent alternative as it managed to mix the advantages that each one had separately. That being said, future works include the behavior of this model using other datasets, highlighting the annotated categories weight, testing other γ factors to FL and other weights into the combination of loss functions.

Keywords: Cardiac Multi-structures. Magnetic Resonance Imaging. Machine Learning. Image Segmentation. Segmentation Analysis. Artificial Neural Networks. Loss functions

Sumário

1	Introdução	1
1.1	Objetivo Principal	3
1.2	Objetivos Específicos	3
1.3	Organização do Trabalho	3
2	Fundamentação Teórica	4
2.1	Aprendizagem de Máquina para Segmentação de Imagens	4
2.1.1	Tipificando a aprendizagem	4
2.1.2	Agente segmentador de imagens	5
2.1.3	Treinando e avaliando um agente segmentador de imagens	8
2.2	Segmentador de Imagens Utilizado	12
2.3	Revisão da Literatura	13
3	Materiais e Métodos	15
3.1	Configurações de <i>Hardware</i>	15
3.2	Base de Dados	16
3.3	Configurações de <i>Software</i>	17
3.4	Hiperparametrização do Segmentador de Imagens Proposto	19
3.5	Escopos Evolutivos do Segmentador de Imagens Proposto	19
3.5.1	Escopo 0: Primeiro contato	20
3.5.2	Escopo 1: Aprendendo com os erros	21
3.5.3	Escopo 2: Firmando a aprendizagem	22
4	Resultados e Discussão	24
4.1	Escopo 0: Primeiro contato	24
4.2	Escopo 1: Aprendendo com os erros	24
4.3	Escopo 2: Firmando a aprendizagem	26
4.4	O que pôde ser aprendido	27
5	Conclusões	35

Lista de Figuras

1.1	(A) Revisão geral da quantidade de artigos publicados e (B) Disponibilização de dados públicos sobre segmentação, ambos sob a ótica das técnicas de captura de imagens médicas utilizada. As abreviações são elucidadas como as seguintes traduções livres: <i>MR.</i> para Imagem de Ressonância Magnética e <i>CT.</i> para Tomografia Computadorizada.	2
1.2	Exemplo de segmentação de imagens cardíacas a partir de Imagem de Ressonância Magnética (IRM), tendo à esquerda a imagem original e à direita as classes anotadas, gabarito, sobrepostas à imagem original. Em vermelho tem-se a cavidade do ventrículo direito, em verde o miocárdio e em azul a cavidade do ventrículo esquerdo (Adaptado: (BERNARD et al., 2018)). . .	2
2.1	Estrutura de uma Rede Neuronal Convolutiva (RNC) aplicada à segmentação de imagens, com indicação da quantidade de filtros por convolução. . .	6
2.2	Estrutura de uma Rede Neuronal Completamente Convolutiva (RNCC) aplicada à segmentação de imagens, com indicação da quantidade de filtros por convolução.	7
2.3	(A) Estrutura de uma Rede Neuronal Recorrente (RNR) em conjunto com uma Rede Neuronal Completamente Convolutiva (RNCC) aplicadas à segmentação de imagens e (B) Destaque estrutural da RNR utilizada. . . .	7
2.4	Visualização da métrica <i>Dice Similarity Coefficient</i> (DSC) ao considerar dois conjuntos binários no espaço.	8
2.5	Interpretações comumente adotadas dos resultados numéricos da métrica <i>Intersection-Over-Union</i> (IoU).	9
2.6	Visualização da construção da métrica <i>Mean Intersection-Over-Union</i> (MeanIoU) num contexto de duas classes anotadas.	9
2.7	Visualização da construção da métrica <i>Hausdorff distance</i> (HD).	10
2.8	Comparativo visual entre a Entropia cruzada original ($\gamma = 0$) com a <i>Focal Loss</i> (FL) em alguns fatores de modulação distintos.	11

2.9	Visão abstrata do método de busca automatizada por hiperparâmetros otimizados para um agente segmentador de imagens cardíacas proposto por Mortazi e Bagci (2018) (Adaptado: (MORTAZI; BAGCI, 2018)).	12
2.10	Estrutura-base utilizada na busca automática por hiperparâmetros, à esquerda, e os detalhes da renomeada camada densa, à direita. As abreviações são elucidadas como as seguintes traduções livres: <i>Conv.</i> para convolucional; <i>Concat.</i> para concatenação e <i>B.N.</i> para normalização ao conjunto sob avaliação (Adaptado: (MORTAZI; BAGCI, 2018)).	13
3.1	Exemplificando o par repetição temporal, à direita, e sua segmentação, à esquerda, disponibilizado pela base de dados fornecida pelo <i>Automated Cardiac Diagnosis Challenge</i> (ACDC) 2017.	18
3.2	Arquitetura otimizada encontrada a partir da busca de hiperparâmetros por Mortazi e Bagci (2018). A abreviação <i>Avg.</i> corresponde a média (Adaptado: (MORTAZI; BAGCI, 2018)).	19
4.1	Desempenho do modelo treinado no Escopo 0 através das métricas DSC (em roxo) e <i>MeanIoU</i> (em ocre) e da função de perda Entropia cruzada (em bordô).	25
4.2	Comparativo visual de uma predição (à direita) do modelo treinado no Escopo 0 dadas as anotações (ao centro) e a IRM original (à esquerda) de referência.	25
4.3	Desempenho do modelo treinado no Escopo 1 sob a métrica DSC ao considerar diferentes funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).	26
4.4	Desempenho do modelo treinado no Escopo 1 sob a métrica <i>Jaccard</i> ao considerar diferentes funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).	27
4.5	Desempenho do modelo treinado no Escopo 1 sob a métrica <i>MeanIoU</i> ao considerar diferentes funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).	27
4.6	Desempenho do modelo treinado no Escopo 1 sob a métrica <i>IoU</i> ao considerar diferentes funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).	28
4.7	Desempenho do modelo treinado no Escopo 1 sob a métrica <i>F1-Score</i> ao considerar diferentes funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).	28

4.8	Desempenho do modelo treinado no Escopo 1 com avaliação direta entre as funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).	29
4.9	Comparativo visual de uma predição (à direita) do modelo treinado no Escopo 1 dadas as anotações (ao centro) e a IRM original (à esquerda) de referência, tendo como função de perda a DiL.	29
4.10	Comparativo visual de uma predição (à direita) do modelo treinado no Escopo 1 dadas as anotações (ao centro) e a IRM original (à esquerda) de referência, tendo como função de perda a FL.	29
4.11	Desempenho do modelo treinado no Escopo 2 através das métricas DSC (em roxo), $F1-Score$ (em ocre) e $MeanIoU$ (em verde) e da função de perda Combinação C_{FP} (Equação 3.2, em bordô).	30
4.12	Comparativo visual de duas predições (à direita) do modelo treinado no Escopo 2 dadas as anotações (ao centro) e as IRMs originais (à esquerda) de referência.	30

Lista de Tabelas

3.1	Comparativo das configurações de <i>hardware</i> empregadas neste trabalho. As menores capacidades dentre os recursos foram destacadas.	16
3.2	Resumo das amálgamas de configurações adotadas tanto por Mortazi e Bagci (2018) quanto ao longo dos escopos evolutivos. As especificações na forma “-” sinalizam aquelas que são desconhecidas. As abreviações são elucidadas como: Red. para redimensionamento, R.A. para razão de aprendizagem e E.C. para Entropia cruzada.	23
4.1	Coletânea dos desempenhos do modelo treinado no Escopo 1 sob diversas métricas e considerando três funções de perda: DiL, FL e a Combinação C_{FP} (Equação 3.2). Os melhores índices estão em destaque para cada uma das métricas disponibilizadas.	26

Lista de Abreviaturas e Siglas

ACDC Automated Cardiac Diagnosis Challenge.

AM Aprendizagem de Máquinas.

API Application Programming Interface.

AVCs Acidentes Vasculares Cerebrais.

CNN Convolutional Neural Network.

Colab Google Colaboratory.

CPU Central Process Unit.

CT Computed Tomography.

DDR Double Data Rate.

DiL Dice Loss.

DL Deep Learning.

Drive Google Drive.

DSC Dice Similarity Coefficient.

FCNN Fully Convolutional Neural Network.

FL Focal Loss.

GB Giga bytes.

GDDR Graphics Double Data Rate.

GPU Graphics Processing Unit.

HBM High Bandwidth Memory.

HD Hausdorff distance.

IA Inteligência Artificial.

IoU Intersection-Over-Union.

IRM Imagem de Ressonância Magnética.

MeanIoU Mean Intersection-Over-Union.

MRI Magnetic Resonance Imaging.

NIFTI Neuroimaging Informatics Technology Initiative.

NPY Objetos NumPy persistidos.

NVMe Non-Volatile Memory Express.

OPAS Organização Pan-Americana da Saúde.

RNC Rede Neuronal Convolutacional.

RNCC Rede Neuronal Completamente Convolutacional.

RNN Recurrent Neural Network.

RNR Rede Neuronal Recorrente.

RVSC Right Ventricle Segmentation Challenge.

s16bit signed 16-bit raw data.

SCD Sunnybrook Cardiac Data.

SSD Solid State Drive.

TC Tomografia Computadorizada.

u8bit unsigned 8-bit raw data.

UnB Universidade de Brasília.

Capítulo 1

Introdução

As doenças cardiovasculares são a principal causa de morte no mundo, segundo a Organização Pan-Americana da Saúde (2021) (OPAS). Estima-se que 17,9 milhões de pessoas morreram por doenças cardiovasculares em 2016, em sua maioria de ataques cardíacos e Acidentes Vasculares Cerebrais (AVCs).

Para combater esta elevada taxa de mortalidade, tanto o diagnóstico quanto o tratamento de doenças cardiovasculares evoluíram, a exemplo da realização de exames por imagens: combinação do qualitativo de método não-invasivo com o quantitativo da descrição das estruturas cardíacas. Dentre as técnicas de captura que seguem e incorporam esta combinação, há de se destacar a *Magnetic Resonance Imaging* (MRI), a *Computed Tomography* (CT)¹ e o ultrassom.

A partir dos resultados dos exames por imagem, a segmentação das imagens cardíacas é de fundamental importância para inúmeras aplicações por proporcionar o agrupamento de regiões semanticamente equivalentes, segundo Chen et al. (2020). Estes agrupamentos são responsáveis pela caracterização da cavidade do ventrículo direito, do miocárdio, da cavidade do ventrículo esquerdo, da espessura das paredes cardíacas e outras estruturas cardíacas conhecidas. Isto posto, Petitjean e Dacher (2011) reforça que anotações manuais de tais estruturas demandam bastante tempo, paciência e cuidado dos especialistas que as fazem.

Frente à dificuldade e à complexidade de anotações manuais, algoritmos na estratégia de Aprendizagem de Máquinas (AM), mais especificamente de *Deep Learning* (DL), surgem como uma alternativa válida por possibilitarem a detecção e a segmentação das estruturas cardíacas internas, como bem apontado por Chen et al. (2020).

Compreendendo que as IRMs fornecem imagens volumétricas de alta resolução, que se traduzem em informações precisas sobre as estruturas cardíacas (PANAYIDES et al.,

¹Tais termos possuem correspondências traduzidas em Imagem de Ressonância Magnética (IRM, em português) e Tomografia Computadorizada (TC, em português) para MRI e CT respectivamente.

2020a; PETITJEAN; DACHER, 2011), e, portanto, torna-se uma técnica de captura de imagens médicas desejável de ser trabalhada. A Figura 1.1 corrobora com a relevância desta técnica, visto que configura maioria entre os artigos publicados sobre segmentação de imagens cardíacas desde 2017 e possui mais dados públicos disponíveis. Além disso, as segmentações de tais imagens correspondem, em sua maioria, às seguintes estruturas cardíacas: a cavidade do ventrículo direito, o miocárdio e a cavidade do ventrículo esquerdo; conforme ilustrado na Figura 1.2.

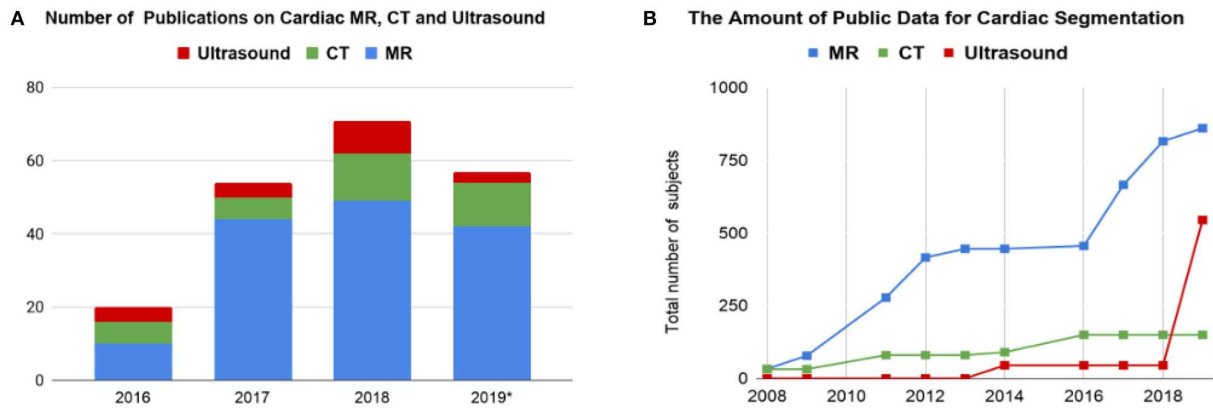


Figura 1.1: (A) Revisão geral da quantidade de artigos publicados e (B) Disponibilização de dados públicos sobre segmentação, ambos sob a ótica das técnicas de captura de imagens médicas utilizada. As abreviações são elucidadas como as seguintes traduções livres: *MR*. para Imagem de Ressonância Magnética e *CT*. para Tomografia Computadorizada (Fonte: (CHEN et al., 2020)).

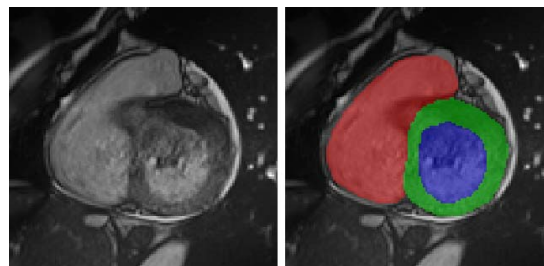


Figura 1.2: Exemplo de segmentação de imagens cardíacas a partir de Imagem de Ressonância Magnética (IRM), tendo à esquerda a imagem original e à direita as classes anotadas, gabarito, sobrepostas à imagem original. Em vermelho tem-se a cavidade do ventrículo direito, em verde o miocárdio e em azul a cavidade do ventrículo esquerdo (Adaptado: (BERNARD et al., 2018)).

1.1 Objetivo Principal

Neste trabalho, almeja-se avaliar a eficiência de segmentador de imagens cardíacas providas por Imagem de Ressonância Magnética (IRM), visto que o emprego de estratégias de Aprendizagem de Máquinas (AM) com esta finalidade vem conquistando mais destaque e vem apresentando diversas possibilidades de manipulação (MORTAZI; BAGCI, 2018).

1.2 Objetivos Específicos

Para tanto, a arquitetura Rede Neuronal Completamente Convolutacional (RNCC) proposta por Mortazi e Bagci (2018) será utilizada como ponto de partida, por ter sido gerada visando a otimização de segmentação de imagens cardíacas. Em vista disso, propõe-se o uso de funções de perda distintas, com a finalidade de melhorar o desempenho do segmentador proposto, e a redução das dimensões das camadas internas, com o intuito de amenizar a dependência de *hardware* sem prejudicar a atuação do segmentador.

1.3 Organização do Trabalho

Para o melhor desenvolvimento e entendimento deste trabalho, tem-se sua divisão em outros quatro capítulos. Neste sentido, o Capítulo 2 apresentará os conceitos e os fundamentos que sustentam este trabalho. Seguido pelo Capítulo 3, que evidencia a forma como tais conceitos foram empregados e manipulados, assim como os recursos de *hardware* e *software*. As escolhas realizadas até este ponto refletiram nos resultados elencados no Capítulo 4. Por último, o Capítulo 5 discorre sobre as conclusões que foram tomadas e refletidas fundadas nos resultados encontrados, não se obstando à proposição de caminhos evolutivos alicerçados neste trabalho.

Capítulo 2

Fundamentação Teórica

Os conceitos e seus entendimentos fundamentais e necessários para a compreensão das técnicas, termos e escolhas feitas por este trabalho nos demais capítulos são detalhados neste capítulo.

2.1 Aprendizagem de Máquina para Segmentação de Imagens

A Aprendizagem de Máquinas (AM) é um ramo da Inteligência Artificial (IA) caracterizável como um problema de busca, visto que a máquina aprende a identificar conceitos a partir de símbolos a eles relacionados e seu objetivo torna-se, portanto, buscar as melhores descrições de tais símbolos. Para a realização de tal processo, é possível fazer-se uso de um conhecimento prévio sobre a busca e até de dados que auxiliem no seu êxito (MITCHELL, 1997).

Dito isso, é preciso compreender os processos de aprendizagem possíveis de serem realizados com esta máquina. Sendo descritos, portanto, na próxima seção.

2.1.1 Tipificando a aprendizagem

A máquina que se encontra em processo de aprendizagem é denominada de agente e este só está aprendendo a partir do momento que demonstra uma melhora de desempenho na realização da tarefa que lhe foi imposta (RUSSELL; NORVIG, 2010). Todavia, para conseguir avaliar o desempenho é preciso compreender a qual regime de treinamento o agente está inserido. Dessa forma, ainda segundo Russell e Norvig (2010), há somente as seguintes tipificações de aprendizagem:

- **Não supervisionada:** o agente aprende ao distinguir padrões nos dados que possui, sem a necessidade de especificação a priori nem de sinalização de qualidade progressiva dos padrões por ele identificados;
- **Por reforço:** o agente aprende ao receber recompensas ou punições sobre as ações realizadas. Portanto, está a critério do próprio agente identificar quais foram as escolhas que tiveram mais influência sobre ação final e a ponderar sobre as mesmas;
- **Supervisionada:** o agente aprende ao conseguir representar uma dada entrada como uma saída a mais próxima possível da que era esperada, por já ter tido acesso a outros exemplos de entrada-saída;
- **Semi-supervisionada:** o agente aprende mesmo quando o tipo não foi claramente assinalado, como não supervisionada ou supervisionada, seja por ruído presente nos dados seja presente nas suas identificações, dessa forma a tipificação torna-se uma linha tênue entre ambos.

Dentre as tipificações acima, a aprendizagem supervisionada é a mais empregada no contexto de segmentação de imagens. O próximo passo, é o entendimento sobre a concepção do agente capaz de realizar tal tarefa.

2.1.2 Agente segmentador de imagens

Agentes baseados em redes neurais¹, uma evolução dos *perceptrons*², são comumente utilizados no estado-da-arte dos algoritmos segmentadores de imagens (CHEN et al., 2020). Ainda segundo Chen et al. (2020), há de se destacar, portanto, as possíveis variações:

- **Rede Neuronal Convolutacional (RNC):** corresponde ao tipo de rede neuronal mais comum dentre as análises de imagens, em inglês *Convolutional Neural Network* (CNN). Por possuir um vetor de tamanho pré-fixado como saída, esta variação é mais aplicada para classificação (os dados do vetor correspondem às probabilidades de cada classe), regressão (os dados do vetor correspondem à valores reais, \mathbb{R} , para a estimação de uma categoria) ou detecção e localização de objetos (os dados do vetor correspondem às coordenadas dos retângulos que englobam os objetos), porém pode ser aplicada para segmentação a partir de ajustes sobre a classificação. Para que isso ocorra é preciso recortar a imagem original em muitos pedaços e classificar o

¹Tradução livre para o português do termo *neural network* (rede de “neurônios” interconectados) a ser utilizada neste trabalho. Esta tradução é mais precisa em significado que a versão mais difundida, “rede neural”, por ser uma analogia a neurônios e não a nervos.

²Unidade lógica que representa um neurônio para a máquina. A partir de ponderações das conexões de entrada aplica-se uma função de ativação, uma transformação sobre a ponderação realizada, e determina seu estado de saída. Sua saída pode ser utilizada como entrada em outro *perceptron*.

centro deste pedaço, para, em seguida, combinar todas as classificações realizadas, conforme ilustrado na Figura 2.1. Apesar de possível, este processo é fortemente desaconselhado: demanda bastante tempo para tratar todos os pedaços e apresenta redundância de processamento entre os pedaços. Pode ser utilizada, entretanto, como uma etapa de pré-processamento da segmentação: detecção e localização das estruturas cardíacas, a fim de otimizar a segmentação a ser performeda;

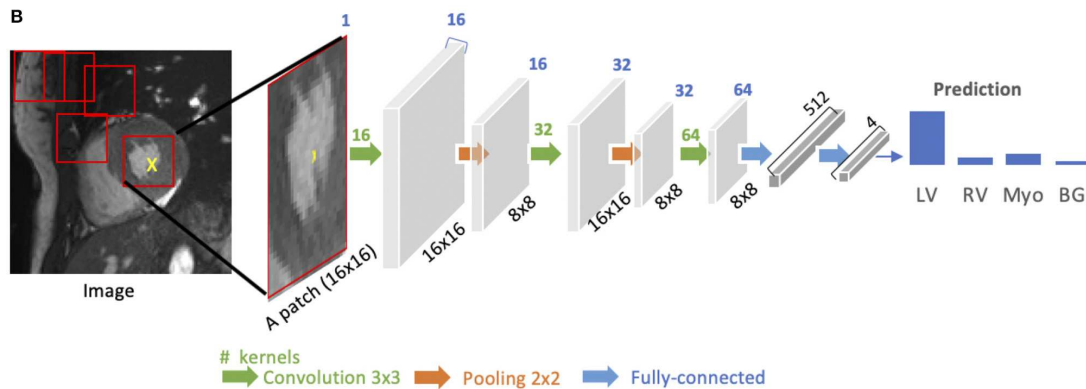


Figura 2.1: Estrutura de uma Rede Neuronal Convolutiva (RNC) aplicada à segmentação de imagens, com indicação da quantidade de filtros por convolução (Fonte: (CHEN et al., 2020)).

- **Rede Neuronal Completamente Convolutiva (RNCC, em tradução livre para o português):** do inglês *Fully Convolutional Neural Network* (FCNN), corresponde a um tipo especial de RNC que dispensa a necessidade dos recortes por manipular a imagem inteira. Conforme ilustrado na Figura 2.2, esta rede constitui-se de uma estrutura *encoder-decoder*, onde a primeira trabalha em extrair as características da entrada e a segunda em reconstruir espacialmente a imagem segmentada a partir de tais características. Infelizmente, há perda de detalhes espaciais que prejudicam a precisão da segmentação;
- **Rede Neuronal Recorrente (RNR):** do inglês *Recurrent Neural Network* (RNN), corresponde ao tipo de rede neuronal capaz de manipular dados sequenciais e de “lembrar” o que foi visto. Desta forma, a segmentação da imagem anterior auxilia na segmentação da imagem atual, por isso costuma ser empregada em conjunto com a RNCC para melhorar a segmentação em contexto tridimensional, conforme ilustrado pela Figura 2.3.

Dentre as variações de redes neurais citadas, variações de uma RNCC é a escolha mais empregada no contexto de segmentações de imagens bidimensionais.

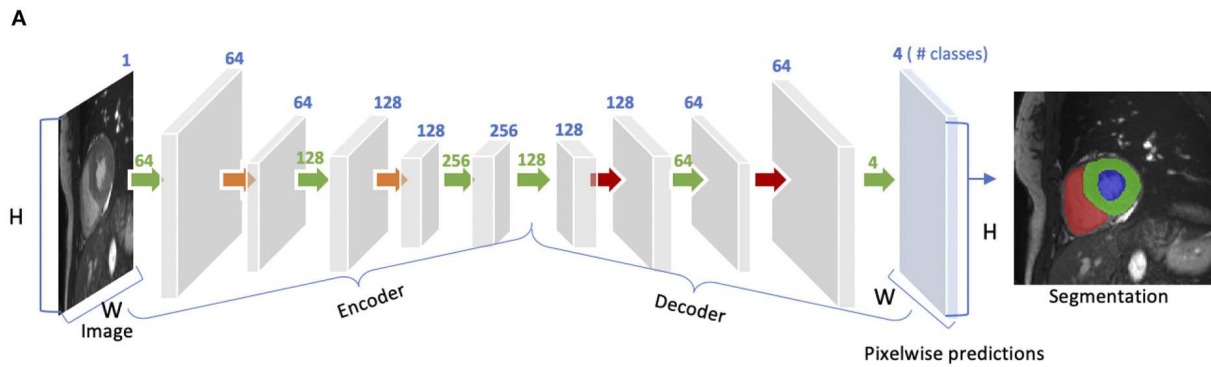


Figura 2.2: Estrutura de uma Rede Neuronal Completamente Convolutiva (RNCC) aplicada à segmentação de imagens, com indicação da quantidade de filtros por convolução (Fonte: (CHEN et al., 2020)).

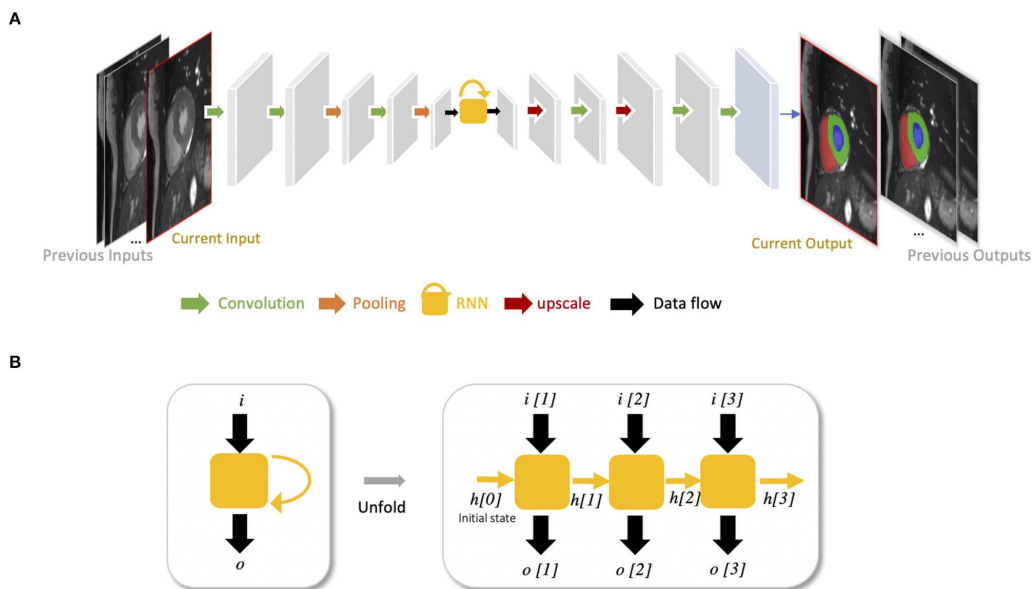


Figura 2.3: (A) Estrutura de uma Rede Neuronal Recorrente (RNR) em conjunto com uma Rede Neuronal Completamente Convolutiva (RNCC) aplicadas à segmentação de imagens e (B) Destaque estrutural da RNR utilizada (Fonte: (CHEN et al., 2020)).

Entendendo-se a melhor concepção do agente segmentador de imagens, é preciso compreender como ajustá-lo às nossas necessidades, seja por método empírico seja por método científico. Tais alterações são feitas nos hiperparâmetros, que são os parâmetros que modelam a arquitetura proposta a partir de uma dada estrutura-base. No contexto da estrutura *encoder-decoder* das RNCCs, definir a quantidade de filtros a ser considerado nas camadas convolucionais, as dimensões desses filtros e as funções de *pooling* usadas é ajustar seus hiperparâmetros.

Por fim, é preciso assimilar as possibilidades de avaliação de desempenho expresso por tal agente e de asserção de progresso em sua aprendizagem.

2.1.3 Treinando e avaliando um agente segmentador de imagens

Até o presente momento foi visto que as IRMs fornecem grandes possibilidades de aplicações, por isso ao empregar um agente mais adequado, uma variação de uma RNCC, em regime de aprendizagem supervisionada é possível obter bons resultados para a segmentação das estruturas cardíacas. Desde que bem avaliado.

Segundo Chen et al. (2020), a segmentação de imagens médicas comumente demanda três tipos de métricas de desempenho: as baseadas em porções volumétricas; as baseadas em distância de superfície e as baseadas em desempenho clínico. Dentre aquelas que pertencem aos primeiros dois tipos é válido destacar as seguintes métricas:

- ***Dice Similarity Coefficient (DSC)***: segundo Dice (1945) pode ser definida como sendo a sobreposição espacial entre duas segmentações, sendo calculada através da Equação 2.1 e ilustrada pela Figura 2.4;

$$DSC(A, B) = 2 \cdot \frac{|A \cap B|}{|A| + |B|} \quad (2.1)$$

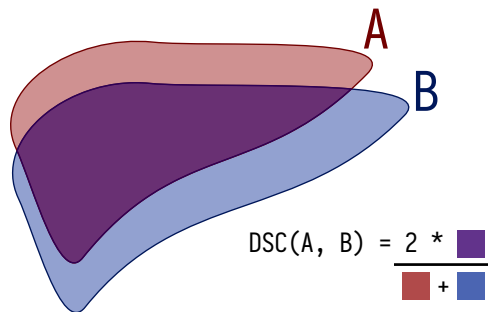


Figura 2.4: Visualização da métrica *Dice Similarity Coefficient* (DSC) ao considerar dois conjuntos binários no espaço.

- ***Intersection-Over-Union (IoU)***: segundo Rezatofighi et al. (2019) pode ser nomeada também como *Jaccard index* e mensura a sobreposição entre duas segmentações independentemente da escala adotada, sendo calculada através da Equação 2.2 e interpretada aconselhadamente segundo a Figura 2.5;

$$IoU(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (2.2)$$



Figura 2.5: Interpretações comumente adotadas dos resultados numéricos da métrica *Intersection-Over-Union* (IoU) (Fonte: (Interstellar Engine, 2021)).

- **Mean Intersection-Over-Union (MeanIoU):** enquanto a IoU manipula segmentações binárias, a MeanIoU considera segmentações multi-classes, sendo calculada através da Equação 2.3 e ilustrada pela Figura 2.6. Nela, C é o total de classes a ser segmentadas e c é a c -ésima classe segmentada;

$$\text{MeanIoU}(A, B) = \frac{1}{C} \sum_{c=0}^C \text{IoU}(A_c, B_c) \quad (2.3)$$

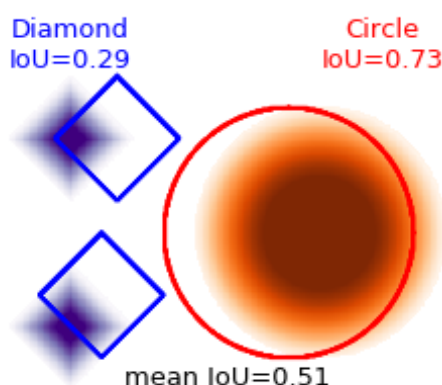


Figura 2.6: Visualização da construção da métrica *Mean Intersection-Over-Union* (MeanIoU) num contexto de duas classes anotadas (Fonte: (MONTEUX, 2019)).

- **Hausdorff distance (HD):** segundo Huttenlocher, Klanderman e Rucklidge (1993) mensura-se o grau de distanciamento espacial entre dois conjuntos, sendo calculada através da Equação 2.4 e ilustrada pela Figura 2.7. Considera-se um ponto do primeiro conjunto que está mais distante de qualquer ponto do segundo e vice-versa.

$$HD(A, B) = \max(h(A, B), h(B, A)) \quad (2.4)$$

onde

$$h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\| \quad (2.5)$$

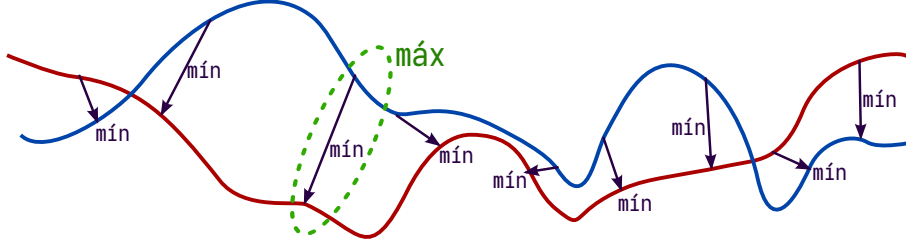


Figura 2.7: Visualização da construção da métrica *Hausdorff distance* (HD).

Para que um agente expresse melhora à ótica de qualquer uma dessas métricas é preciso realizar um bom treinamento. Para tanto, o agente realiza iterações sobre os dados à sua disposição (exames em IRM e suas classes anotadas) almejando predições mais próximas às desejadas. Esta avaliação de proximidade é realizada pelas funções de perda (RUSSELL; NORVIG, 2010). As funções de perda $\mathcal{L}(x, y, \hat{y})$ correspondem a quantidade de utilidade desperdiçada pela predição $h(x) = \hat{y}$ quando na verdade deveria ser $f(x) = y$:

$$\begin{aligned} \mathcal{L}(x, y, \hat{y}) = & \text{Utilidade}(\text{resultado dado como } y \text{ para a entrada } x) \\ & - \text{Utilidade}(\text{resultado dado como } \hat{y} \text{ para a entrada } x) \end{aligned} \quad (2.6)$$

Dessa forma, no contexto de segmentação de imagens, destaca-se as seguintes funções de perda:

- **Entropia cruzada:** segundo Cover e Thomas (2006) pode ser nomeada também como entropia relativa $\mathcal{D}(p||q)$ e determina a “distância” entre duas distribuição de probabilidades p e q , sendo calculada através da Equação 2.7. No contexto atual, o resultado esperado é a distribuição q e o obtido é a p ;

$$\mathcal{D}(p||q) = \sum_x p(x) \log \frac{p(x)}{q(x)} \quad (2.7)$$

- **Focal Loss (FL):** segundo Lin et al. (2017) a forte presença de classes desbalanceadas entre os resultados desejados prejudica a corretude na análise através da $\mathcal{D}(p||q)$. Isto posto, inclui-se um fator de modulação à análise da Entropia cruzada, sendo calculada através da Equação 2.8 (em sua notação simplificada). Quando o parâmetro de foco γ é nulo, esta função de perda exhibe um comportamento semelhante

à sua fonte, conforme ilustrado na Figura 2.8;

$$FL(p_t) = -(1 - p_t)^\gamma \log(p_t) \quad (2.8)$$

- **Dice Loss (DiL):** com derivação direta da métrica DSC (DICE, 1945), sendo calculada através da Equação 2.9.

$$\text{DiL} = 1 - \text{DSC} \quad (2.9)$$

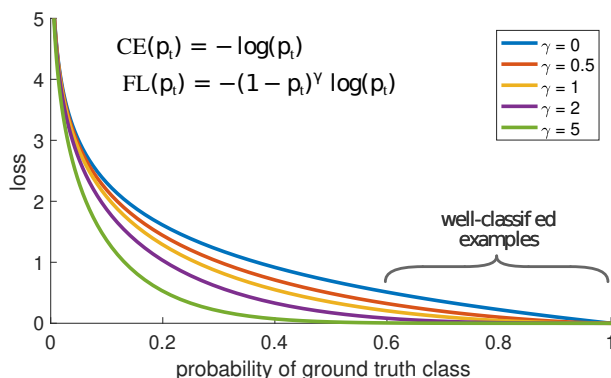


Figura 2.8: Comparativo visual entre a Entropia cruzada original ($\gamma = 0$) com a *Focal Loss* (FL) em alguns fatores de modulação distintos (Fonte: (LIN et al., 2017)).

Conforme salientado por Chen et al. (2020), não é possível realizar uma comparação direta entre agentes distintos, mesmo sendo avaliados à ótica das mesmas métricas de desempenho. Para que isso seja possível é preciso treiná-los com a mesma base de dados.

Assim sendo, esta base deve disponibilizar entradas e suas saídas conhecidas (fontes-verdade) por estar inserido na abordagem de aprendizagem supervisionada. A Figura 1.2 exemplifica um conjunto entrada-saída válido neste contexto, onde as entradas são as IRMs e as fontes-verdade são as classes anotadas.

Além disso, recomenda-se a fragmentação dos dados à disposição em dois ou três conjuntos: treinamento/teste ou treinamento/validação/teste. Desta maneira, o agente aprenderá a partir de interações com o conjunto de treinamento e será posto à prova com os demais, por serem a priori desconhecidos pelo agente e, portanto, ostentariam avaliações genuínas. Enquanto a primeira divisão dos conjuntos peca em disponibilizar dados nunca vistos pelo agente para uma avaliação final, por já tê-los utilizados nas avaliações parciais das épocas, a segunda divisão supre este contratempo com o conjunto de validação, recorrendo ao de teste somente para a finalização.

2.2 Segmentador de Imagens Utilizado

O segmentador de imagens cardíacas arquitetado por Mortazi e Bagci (2018) atende às boas características de agente e de ambiente de treinamento e avaliação discutidos anteriormente, sendo capaz de segmentar três estruturas cardíacas: a cavidade do ventrículo direito, o miocárdio e a cavidade do ventrículo esquerdo.

Com o intuito de encontrar um agente segmentador de imagens otimizado ao contexto das imagens cardíacas por IRM, Mortazi e Bagci (2018) propuseram a busca automatizada, ilustrada na Figura 2.9, por seus hiperparâmetros. Para tanto, foram elegidos: **a)** o número de filtros convolucionais por camada convolucional entre 32 e 208, **b)** e **c)** as alturas e as larguras desses filtros entre 1 e 11, respectivamente, e **d)** as funções de *pooling* entre a por valor médio e a por valor máximo. Essa busca coerente e intrínseca à estrutura-base utilizada, ilustrada na Figura 2.10, por ser uma derivação das RNCCs.

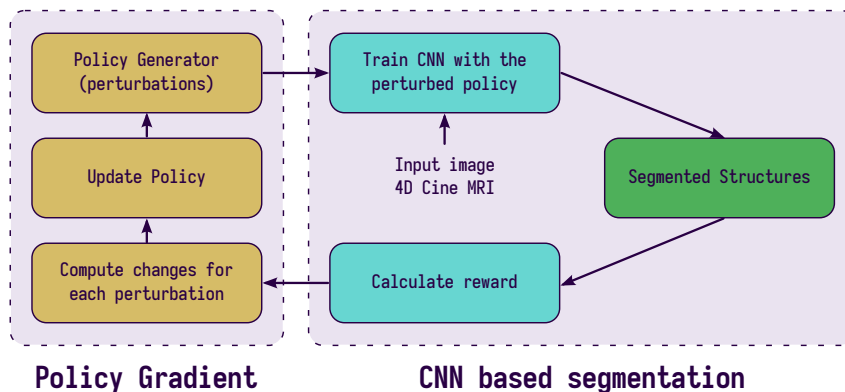


Figura 2.9: Visão abstrata do método de busca automatizada por hiperparâmetros otimizados para um agente segmentador de imagens cardíacas proposto por Mortazi e Bagci (2018) (Adaptado: (MORTAZI; BAGCI, 2018)).

A arquitetura do agente era configurada a partir de uma política de gradiente e este, por sua vez, realizava um treinamento sob aprendizagem supervisionada de até 50 épocas. Ao final de cada treinamento, a arquitetura utilizada é avaliada em regime de aprendizagem por reforço a partir da média da DSC das últimas 5 épocas treinadas.

Vale ressaltar a repaginação utilizada para as *camadas densas*, por ser um componente já conhecido entre as estratégias de AM. A alteração realizada é conjunto conexo de outras quatro camadas internas, onde cada uma representa a sequência de uma camada convolucional, uma normalização por conjunto de entrada e uma ativação *Swish*. Além do mais, as camadas convolucionais consomem tanto a entrada da camada densa quanto a combinação dos resultados das demais camadas internas anteriores. Estas conexões estão ilustradas na Figura 2.10.

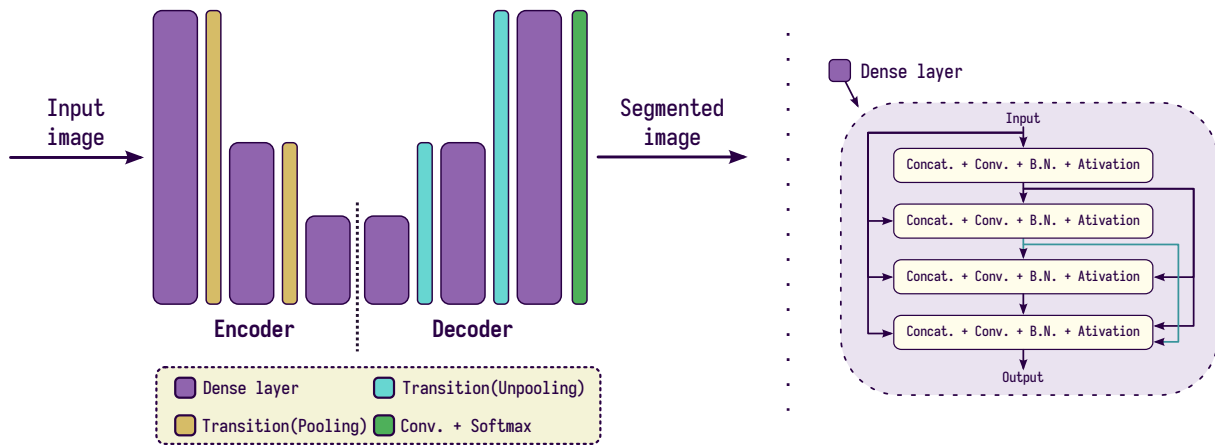


Figura 2.10: Estrutura-base utilizada na busca automática por hiperparâmetros, à esquerda, e os detalhes da renomeada camada densa, à direita. As abreviações são elucidadas como as seguintes traduções livres: *Conv.* para convolucional; *Concat.* para concatenação e *B.N.* para normalização ao conjunto sob avaliação (Adaptado: (MORTAZI; BAGCI, 2018)).

Com a ressignificação da camada densa o agente se tornaria mais consciente sobre a composição espacial das imagens originais, logo, a reconstrução espacial das segmentações realizadas se aproximaria das fontes-verdade fornecidas. Esta estratégia difere da variação das RNCCs proposta por Ronneberger, Fischer e Brox (2015), a U-Net, que considera as características extraídas nas etapas do *encoder* diretamente nas etapas correspondentes do *decoder*, este processo é denominado *skip connections*.

2.3 Revisão da Literatura

A segmentação de imagens cardíacas possui diversas abordagens na literatura por manipulação de IRM com Tran (2016) e Khened, Kollerathu e Krishnamurthi (2019), de TC com Zreik et al. (2016) e Ye et al. (2019) e de ultrassom com Carneiro, Nascimento e Freitas (2010) e Carneiro, Nascimento e Freitas (2011), por exemplo.

A utilização de AM para a segmentação de imagens médicas está presente em diversas pesquisas, como a de Ronneberger, Fischer e Brox (2015) e a de Badrinarayanan, Kendall e Cipolla (2017). Contudo, no contexto específico de imagens cardíacas, Tran (2016) pode ser considerado o pioneiro em utilizar uma RNCC para a segmentação do endocárdio e do epicárdio da cavidade do ventrículo direito e obteve DSC de 0,86 e 0,84 respectivamente, sendo o melhor desempenho na época utilizando a base de dados *Right Ventricle Segmentation Challenge* (RVSC) (PETITJEAN et al., 2015).

Entendendo esta eficiência expressa com o emprego de RNCC, Jang et al. (2018), Yang et al. (2018), Chen, Fang e Liu (2019) investigaram o impacto de diferentes funções de perda como a entropia cruzada ponderada, DiL ponderado e a FL. Tais análises utilizaram as bases de dados *Automated Cardiac Diagnosis Challenge* (ACDC) (BERNARD et al., 2018) e *Sunnybrook Cardiac Data* (SCD) (ALFAKIH et al., 2003) e obtiveram DSCs médios de 0,902, 0,827 e 0,93 respectivamente.

Devido à formação comumente circular da cavidade do ventrículo esquerdo existem aqueles que se baseiam nessa característica para otimizar a zona de busca e conseqüente segmentação, como é o caso de Cocosco et al. (2004) e Jolly (2008). Esta abordagem apresentou resultados visuais robustos, segundo os próprios pesquisadores, uma vez que os resultados numéricos refletem uma acurácia subjetiva. Apesar disso, apresenta-se como uma alternativa para otimizar o processamento das entradas.

Capítulo 3

Materiais e Métodos

Os recursos utilizados para o melhor desenvolvimento deste trabalho são descritos e detalhados nesta seção, assim como as manipulações a estes aplicadas. Igualmente exploradas são as dependências técnicas e o segmentador de imagens proposto neste trabalho¹.

Este trabalho foi segmentado em escopos distintos e progressivos, sendo estes discriminados oportunamente a seguir.

3.1 Configurações de *Hardware*

Dentro das estratégias de AM há aquelas que conseguem atingir seu objeto em tempo hábil com o poder computacional oferecido por *Central Process Unit* (CPU), mas no que diz respeito à segmentação de imagens é mais aconselhado o uso de *Graphics Processing Unit* (GPU), cujo poder computacional é superior à primeira opção.

A demanda por GPUs foi suprida pelo *Google Colaboratory* (Colab) em sua oferta gratuita dos seguintes modelos *NVIDIA*[®]: K80, T4, P4 e P100; com memória GDDR² de 12,7GB aproximadamente. Esta oferta de GPU deve ser explicitamente requisitada pelo usuário e não é possível escolher entre os modelos diretamente, por serem alocados dinamicamente e aleatoriamente por requisição. Além dessa restrição, tais recursos são liberados por algumas horas apenas, quando não bloqueados. Tais restrições podem ser contornadas na versão *Pro* (COLAB, 2021).

O Colab foi utilizado como a solução de *hardware* até o instante em que os arquivos manipuláveis pelo ecossistema apresentaram sinais de corrupção, quando não inexistentes, provável reflexo na comutação de dados exigida com os servidores do *Google Drive* (Drive)

¹O código desenvolvido está disponível no GitHub sob o *link* <<https://github.com/rodrigofegui/cardiac-mri-image-segmentation/>>.

²Assim como a *Double Data Rate* (DDR) estão relacionadas à *Central Process Unit* (CPU), a *Graphics Double Data Rate* (GDDR) está relacionada à *Graphics Processing Unit* (GPU). Diferem-se pela especialização em renderização rápida das GPUs.

Tabela 3.1: Comparativo das configurações de *hardware* empregadas neste trabalho. As menores capacidades dentre os recursos foram destacadas.

Dispositivos		Recursos		
		Núcleos (GPUs)	Memória (GB)	Taxa à memória (GB/s)
Colab	K80	4992	24 GDDR5	480
	T4	2560	16 GDDR6	300
	P4	2560	8 GDDR5	192
	P100	3584	12 HBM2	549
16 HBM2			732	
Notebook	GTX 1650Ti	896	4 GDDR6	192

- armazenamento em nuvem utilizado em conjunto. Por conseguinte, adotou-se uma opção local.

A solução local configura-se como um *notebook* com processador *Intel*[®] *Core*[™] i5-10300H, memória de 16GB DDR4, SSD *NVMe*[®] de 500GB e *NVIDIA*[®] *GeForce*[®] GTX 1650Ti GPU (4GB GDDR6). Esta mudança de ambiente de *hardware* acarretou no decréscimo dos recursos disponíveis, conforme elencado na Tabela 3.1, e, portanto, houve um impacto sobre a estruturação deste trabalho.

3.2 Base de Dados

Considerando as modalidades de imagens médicas disponíveis e entendendo as possibilidades de análises conhecidas para cada uma delas, a base de dados considerada para este trabalho é a disponibilizada através do *Automated Cardiac Diagnosis Challenge* (ACDC) 2017. Esta base de dados faz uso de uma técnica não-invasiva de captura e as imagens que a formam possuem diversas aplicabilidades.

Esta base é constituída por IRMs de exames reais adquiridos no Hospital Universitário de Dijon, França, ao longo de seis anos. Foram utilizados dois equipamentos durante as rotinas clínicas com intensidades magnéticas diferentes: 1,5T (Siemens MAGNETOM Aera) e 3,0T (Siemens MAGNETOM Trio Tim). Com estes equipamentos, foram coletados os exames de 150 pacientes (anônimos e sob regulação ética) divididos igualmente em cinco categorias: uma dos saudável e quatro dos com patologia - infarto no miocárdio, cardiomiopatia dilatada, cardiomiopatia hipertrófica e ventrículo direito anormal. Destes, 100 foram atribuídos ao conjunto de treinamento e o outros 50 ao conjunto de testes (BERNARD et al., 2018).

Ainda segundo Bernard et al. (2018), as IRMs foram anotadas manualmente por dois especialistas independentes e foram consideradas quatro classes de segmentação: **0**, **1**, **2** e **3** representando o plano de fundo, a cavidade do ventrículo direito, o miocárdio e

a cavidade do ventrículo esquerdo, respectivamente. Os exames e suas classes anotadas foram salvas em formato .NIFTI (LAROBINA; MURINO, 2014), preservando sua temporalidade. Dessa forma, os dados de cada paciente foram organizados em diretórios com a seguinte estrutura:

- 1 arquivo `Info.cfg`: informações sobre o paciente como quais são os instantes no ciclo cardíaco que foram anotados, a qual categoria pertence, qual sua altura e seu peso e quantas repetições temporais foram capturadas;
- 1 arquivo no formato `patientXXX_4d.nii.gz`: exame original completo realizado, comumente apresentando 10 fatias espaciais e 30 repetições temporais para cada fatia espacial;
- 2 arquivos no formato `patientXXX_frameYY.nii.gz`: YY° repetição temporal do exame que foi anotado nas classes supracitadas, mantendo as respectivas fatias espaciais;
- 2 arquivos no formato `patientXXX_frameYY_gt.nii.gz`: classes anotadas para o YY° repetição temporal, levando em consideração suas fatias espaciais.

Os pares de arquivos representando um repetição temporal e suas segmentações anotadas são exemplificados na Figura 3.1. As classes anotadas são mapeadas diretamente a valores dos *pixels* da imagem correspondente, sendo registradas em representação `u8bit`. No que tange os recortes temporais, há representações `float` mas em sua maioria como `s16bit`. Além disso, as dimensões das imagens variam de paciente para paciente. Dessa forma, para fazer o *download* desta base de dados de treinamento são necessários 1,7GB de espaço livre no dispositivo de armazenamento desejado.

Vale ressaltar que embora o desafio tenha dividido os exames em conjuntos de treinamento e de teste, apenas o de treinamento é disponibilizado vide um rápido cadastro e, portanto, somente este foi adquirido pelo estudante. Este, por conseguinte, gerou conjuntos de treinamento, validação e testes a partir do conjunto de treinamento original da base ACDC, sendo estes àqueles efetivamente manipulados por este trabalho.

3.3 Configurações de *Software*

Considerando as opções de linguagens de programação passíveis de desenvolvimentos de estratégias em AM, a linguagem escolhida foi o *Python*[™] (ROSSUM; DRAKE, 2009) na sua versão v3.9.2. Até o presente momento, a grande maioria, se não todos, dos comparativos focados nesta aplicabilidade indicam tal linguagem como a melhor opção.

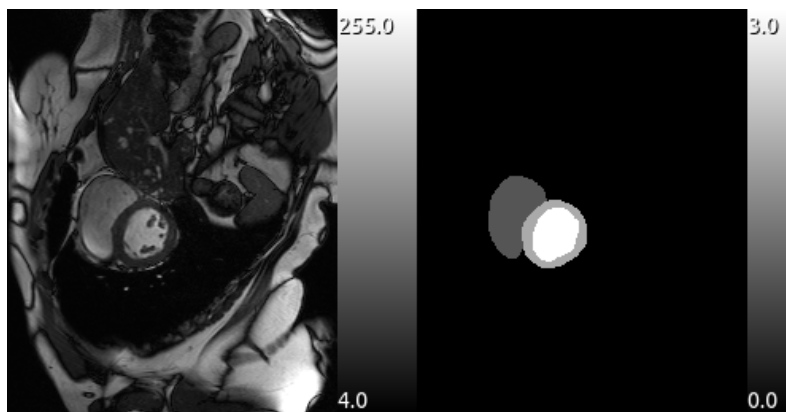


Figura 3.1: Exemplificando o par repetição temporal, à direita, e sua segmentação, à esquerda, disponibilizado pela base de dados fornecida pelo *Automated Cardiac Diagnosis Challenge* (ACDC) 2017 (Fonte: (BERNARD et al., 2018)).

Além disso, possui uma vasta comunidade de usuários nesta área, conforme elencado em IEEE Spectrum (2021).

Devido à popularidade da linguagem de programação selecionada há várias opções de bibliotecas e *frameworks* inseridos nos mais diversos contextos de aplicações. Por uma questão de familiaridade foi escolhido o *framework* *Tensorflow* (ABADI et al., 2015) para manipular o *pipeline* da AM. Além disso, como abstração da complexidade das partes e dos componentes básicos e comuns à AM foi utilizada a API *Keras* (CHOLLET et al., 2015). Enquanto que para a leitura das IRMs foi selecionada a biblioteca *NiBabel* (BRETT et al., 2020), na sua versão v3.2.1. Para expandir as possibilidades de aumento artificial dos dados (este procedimento é comumente nomeado em sua versão em inglês: *data augmentation*) à disposição fez-se uso da biblioteca *scikit-image* (WALT et al., 2014), na sua versão v0.24.2. O enriquecimento das métricas de desempenho e funções de perda foi possibilitado com o uso da biblioteca *Segmentation Models* (YAKUBOVSKIY, 2019), na sua versão v1.0.1. Outras bibliotecas foram utilizadas no ecossistema desenvolvido mas sem impacto sobre o desempenho ou sobre os resultados deste trabalho, por isso não foram explicitados.

Além do benefício de sua alta aplicabilidade às estratégias em AM e da vasta comunidade construída, a escolha da linguagem *Python*[™] proporciona uma abstração poderosa quando sistema operacional que está inserido. Dessa forma, este ecossistema pode ser executado em ambientes Linux (como é o contexto deste trabalho), Windows ou até MacOS. Tal seleção fica à cargo do usuário manipulador do mesmo.

Por fim, aplicou-se a abordagem baseada em cadernos *Jupyter* (KLUYVER et al., 2016). Tais cadernos podem armazenar os cálculos e as iterações realizadas em conjunto com o código desenvolvido (não limitado à linguagem *Python*[™]): possibilidade de explicações mais enriquecidas e diretas a terceiros; além de possuir uma grande variedade de

ferramentas para seu compartilhamento, conforme elucidado por Kluyver et al. (2016).

3.4 Hiperparametrização do Segmentador de Imagens Proposto

Conforme discutido na Seção 2.2, Mortazi e Bagci (2018) consideraram a estrutura-base ilustrada na Figura 2.10 como ponto de partida da busca automática por seus hiperparâmetros otimizados à segmentação semântica de imagens cardíacas. Procurou-se os melhores quantificadores e qualificadores para a quantidade de filtros, para a dimensões destes filtros e para as operações de *pooling* passíveis de serem realizadas. Este processo resultou na arquitetura ilustrada na Figura 3.2.

Tal processo fez uso da base de dados fornecida pelo Automated Cardiac Diagnosis Challenge (ACDC) 2017, a mesma utilizada neste trabalho - conforme discutido na Seção 3.2, portanto os hiperparâmetros por ele encontrados foram mantidos e utilizados pelo segmentador de imagens proposto. No que se refere aos pesos constituintes do modelo, considerou-se um treinamento desde o estágio inicial.



Figura 3.2: Arquitetura otimizada encontrada a partir da busca de hiperparâmetros por Mortazi e Bagci (2018). A abreviação *Avg.* corresponde a média (Adaptado: (MORTAZI; BAGCI, 2018)).

3.5 Escopos Evolutivos do Segmentador de Imagens Proposto

Tendo em vista o Capítulo 1, os treinamentos dos modelos são limitados em 50 épocas da mesma forma que Mortazi e Bagci (2018) o fizeram. Os escopos aqui descritos são amálgamas de detalhes das configurações gerais supracitadas, ecossistemas, que potencialmente impactam o desempenho e/ou os resultados do segmentador de imagens proposto. Ao final de tais descrições, a Tabela 3.2 traz o resumo do conjunto de configurações frente às adotadas por Mortazi e Bagci (2018).

3.5.1 Escopo 0: Primeiro contato

Neste escopo aconteceu o primeiro contato com a arquitetura proposta por Mortazi e Bagci (2018), ou seja, as informações contidas no artigo-base foram consideradas como definitivas, porém admitindo-se as limitações impostas pelo *hardware*. Com isso em mente, foram adotadas as seguintes configurações:

- **Configuração de *Hardware*:** foi utilizada a solução oferecida pelo Colab;
- **Base de Dados:** assim como descrito por Mortazi e Bagci (2018), as imagens manipuladas foram padronizadas às dimensões 200×200 , sendo obtidas por redimensionamento com interpolação por área a partir das imagens originais. As IRMs foram normalizadas individualmente e suas anotações foram codificadas na notação *one hot*; os conjuntos completos resultantes foram armazenados em arquivos .NPY (HARRIS et al., 2020);
- **Configuração de *Software*:** o *framework Tensorflow* (ABADI et al., 2015) foi utilizado na sua versão v2.4.1 e a API *Keras* (CHOLLET et al., 2015) na sua versão v2.4.3;
- **Hiperparametrização do Segmentador de Imagens Proposto:** as reduções das operações *pooling* corresponderam a $1/5$ do tamanho, diferente dos utilizado por Mortazi e Bagci (2018). Em contrapartida, foi usado o mesmo otimizador (*Adam*) com a mesma razão de aprendizagem ($1 \cdot 10^{-4}$) e a mesma função de perda (entropia cruzada).

O segmentador de imagens proposto neste escopo foi avaliado sob as métricas de desempenho DSC e *MeanIoU*, tendo à disposição $4/5$ dos dados para treinamento e $1/5$ para validação cruzada. A fim de otimizar o treinamento, admitiu-se uma estabilidade máxima de três épocas para uma interrupção prematura. Este ocorreu utilizando a Entropia cruzada como função de perda, por ter sido a mesma utilizada por Mortazi e Bagci (2018).

Para a renderização tanto da anotação quanto da predição adotou-se a Equação 3.1: a somas de suas camadas ponderadas pela sua ordem. Este método mantém a imagem-resultado como a composição de valores decimais, que simbolizam os pesos atribuídos às classes, com isso a incerteza acarretaria na falta de contornos definidos e, por conseguinte, na presença de borrões.

$$p = \sum_{i=0}^N i \cdot c_i \quad (3.1)$$

, onde p é o valor do *pixel* correspondente na imagem resultante, i é o índice da classe da segmentação semântica, N é o número de classes a serem segmentadas decrescido de uma unidade (em particular, $N = 3$) e c_i é o valor anotado na i -nêssima classe de segmentação.

3.5.2 Escopo 1: Aprendendo com os erros

Com o aprendizado do escopo anterior foram realizados os seguintes ajustes nas configurações disponíveis:

- **Configuração de *Hardware*:** foi utilizada a solução local;
- **Base de Dados:** a redução de capacidade computacional, elucidada na Tabela 3.1, impôs uma redução das dimensões das imagens a serem manipuladas, divergindo das adotadas por Mortazi e Bagci (2018). Não obstante a isso, a definição da nova configuração foi baseada na análise de regiões de interesse das fontes-verdade disponibilizadas, o que resultou em imagens de tamanho 100×100 . As IRMs e suas anotações foram recortadas nestas medidas em janelas deslizantes espaçadas entre si em 30 *pixels* vertical e horizontalmente. As primeiras foram normalizadas individualmente, ao passo que as segundas foram codificadas na notação *one hot*; ao final destas manipulações, arquivos .NPY (HARRIS et al., 2020) individuais foram gerados e armazenados em disco;
- **Configuração de *Software*:** o *framework Tensorflow* (ABADI et al., 2015) foi utilizado na sua versão v2.5.0 e a API *Keras* (CHOLLET et al., 2015) na sua versão internalizada pelo anterior;
- **Hiperparametrização do Segmentador de Imagens Proposto:** as reduções das operações *pooling* corresponderam a $1/5$ do tamanho, diferente dos $1/2$ utilizado por Mortazi e Bagci (2018). Além disso, houve ajuste na razão de aprendizagem inicial ($1 \cdot 10^{-3}$) com possibilidade de decréscimo até $1 \cdot 10^{-6}$, em casos de estabilidade de 3 épocas. Mantendo, portanto, a função de ativação (*Swish*) e o otimizador (*Adam*).

O segmentador de imagens proposto neste escopo foi avaliado sob as métricas de desempenho DSC, *Jaccard index*, *MeanIoU*, *IoU* e *F1-Score*. Quanto às funções de perda, um dos objetivos deste trabalho é a análise sobre o impacto destas no desempenho do agente segmentador, por isso foram adotadas a DiL, a FL (com $\gamma = 2$) e a combinação (C_{FP}) destas duas, determinada conforme a Equação 3.2. Por limitação de tempo de desenvolvimento, a tomada de decisão sobre estudo baseou-se em treinamentos limitado a 10 épocas.

$$C_{FP} = \text{DiL} + \text{FL} \quad (3.2)$$

Para a renderização adotou-se tanto a Equação 3.1 (a somas de suas camadas ponderadas pela sua ordem) quanto a Equação 3.3: o índice do maior valor disponível. Dessa forma, há uma otimização dos valores anotados entre as camadas por estarem em codificação *one hot*.

$$p = \underset{c}{\operatorname{argmax}}(c_i, \forall i \in N) \quad (3.3)$$

, onde p é o valor do *pixel* correspondente na imagem resultante, i é o índice da classe da segmentação semântica, N é o número de classes a serem segmentadas decrescido de uma unidade (em particular, $N = 3$) e c_i é o valor anotado na i -nêssima classe de segmentação.

3.5.3 Escopo 2: Firmando a aprendizagem

Com o aprendizado proporcionado pelo desenvolvimento do escopo anterior, a maioria das configurações foram mantidas para este novo contexto. Há de se distinguir, porém, que o agente aqui proposto foi avaliado sob as métricas DSC, *MeanIoU* e *F1-Score*, sendo treinado com a combinação C_{FP} (Equação 3.2) como função de perda num treinamento de até 50 épocas. Além disso, a renderização adotou a Equação 3.3 (o índice do maior valor disponível) como método.

³Informações sobre esta placa *NVIDIA*® estão disponíveis *online* sob o *link* <<https://www.nvidia.com/en-us/geforce/products/10series/titan-x-pascal/>>.

⁴Informação descoberta após o início do Escopo 1.

Tabela 3.2: Resumo das amálgamas de configurações adotadas tanto por Mortazi e Bagci (2018) quanto ao longo dos escopos evolutivos. As especificações na forma “-” sinalizam aquelas que são desconhecidas. As abreviações são elucidadas como: Red. para redimensionamento, R.A. para razão de aprendizagem e E.C. para Entropia cruzada.

Configurações	Experimentos			
	Mortazi e Bagci (2018)	Escopo 0	Escopo 1	Escopo 2
Hardware	Titan X ³	Colab	Notebook	Notebook
IRMs	200x200	200x200	100x100	100x100
Garantia dimensão	Recortes ⁴	Red.	Recortes	Recortes
Normalização	-	Individual	Individual	Individual
Codificação	-	One hot	One hot	One hot
Pooling	2	5	5	5
Ativação	Swish	Swish	Swish	Swish
Otimizador	Adam	Adam	Adam	Adam
R.A.	$1 \cdot 10^{-4}$	$1 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$
Função de perda	E.C.	E.C.	DiL	C_{FP}
			FL	
			C_{FP}	
Métrica	DSC	DSC	DSC	DSC
	HD	$MeanIoU$	$Jaccard\ index$	$MeanIoU$
			$MeanIoU$	
			IoU	

Capítulo 4

Resultados e Discussão

Os resultados encontrados em cada um dos escopos evolutivos serão descritos e detalhados nesta seção. A posteriori, a discussão sobre tais resultados será igualmente dissertada.

4.1 Escopo 0: Primeiro contato

Em decorrência das configurações descritas na Seção 3.5.1, os resultados aqui apresentados retratam o treinamento de 15 épocas com interrupção prematura frente as 50 que tinha à disposição. Em vista disso, a Figura 4.1 traz a evolução gráfica do treinamento do modelo sob a ótica das métricas DSC e *MeanIoU* e da função de perda Entropia cruzada. Ao considerar todas as classes de segmentação semântica, tal evolução pôde ser traduzida aos seguintes resultados numéricos: DSC de 0,9493, *MeanIoU* de 0,375 (constante desde o início) e, ainda menor, Entropia cruzada de 0,0907.

Além disso, no aspecto visual, o modelo obteve predições idênticas à exemplificada na Figura 4.2, isto é, independente do conjunto de entrada fornecido o mesmo conjunto de saída era fornecido. Para a renderização, tanto da anotação quanto da predição, foram adotadas as somas de suas camadas ponderadas pela sua ordem (Equação 3.1), ou seja, a primeira camada possui peso 0, a segunda peso 1 e assim em diante, para evidenciar quaisquer falha na atribuição de seus valores.

4.2 Escopo 1: Aprendendo com os erros

Em decorrência das configurações descritas na Seção 3.5.2, os resultados aqui apresentados retratam o treinamento de 10 épocas (todas possíveis) sob as diversas métricas utilizadas. Isto posto, as Figuras 4.3 a 4.7 trazem a evolução gráfica do treinamento do modelo sob a ótica das métricas DSC, *Jaccard index*, *MeanIoU*, *IoU* e *F1-Score* respectivamente. Nestas figuras, têm-se o desempenho progressivo comparando o emprego de funções de

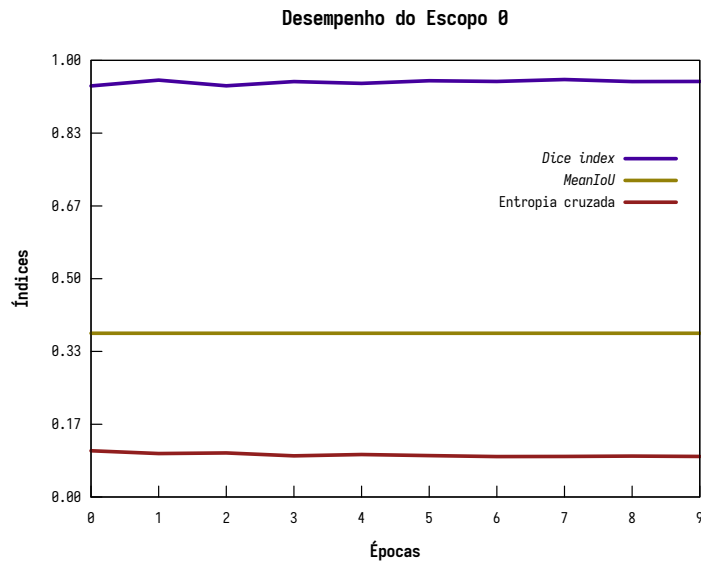


Figura 4.1: Desempenho do modelo treinado no Escopo 0 através das métricas DSC (em roxo) e *MeanIoU* (em ocre) e da função de perda Entropia cruzada (em bordô).



Figura 4.2: Comparativo visual de uma predição (à direita) do modelo treinado no Escopo 0 dadas as anotações (ao centro) e a IRM original (à esquerda) de referência.

perda distintas: DiL, FL (com $\gamma = 2$) e a combinação (C_{FP} , Equação 3.2). Ao considerar todas as classes de segmentação semântica, tais evoluções puderam ser traduzidas nos resultados numéricos apresentados na Tabela 4.1. Há o destaca dos melhores índices para cada uma das métricas avaliadas. Além de comparações indiretas, a Figura 4.8 ilustra a evolução sob a óticas das funções de perda.

No que o aspecto visual se propõe, a Figura 4.9 exemplifica uma predição do modelo treinado com a DiL como função de perda e a Figura 4.10 para o treinado com a FL, ambas interpretando o mesmo conjunto de imagem e anotações. Para a renderização, tanto da anotação quanto da predição, foram adotados os índices do maior valor disponível (Equação 3.3), ou seja, a existência de quatro classes de segmentação, logo as anotações e as predições são manipuladas com quatro extratos, para cada *pixel* de resultado considera-se a posição do maior valor encontrado neste mini-volume.

Tabela 4.1: Coletânea dos desempenhos do modelo treinado no Escopo 1 sob diversas métricas e considerando três funções de perda: DiL, FL e a Combinação C_{FP} (Equação 3.2). Os melhores índices estão em destaque para cada uma das métricas disponibilizadas.

Métrica de desempenho	Função de perda		
	DiL	FL	Combinação C_{FP}
DSC	0.9027	0.4679	0.6645
<i>Jaccard index</i>	0.6020	0.3994	0.5316
<i>MeanIoU</i>	0.9203	0.3750	0.4547
<i>IoU</i>	0.4384	0.6972	0.6230
<i>F1-Score</i>	0.5096	0.7816	0.7186

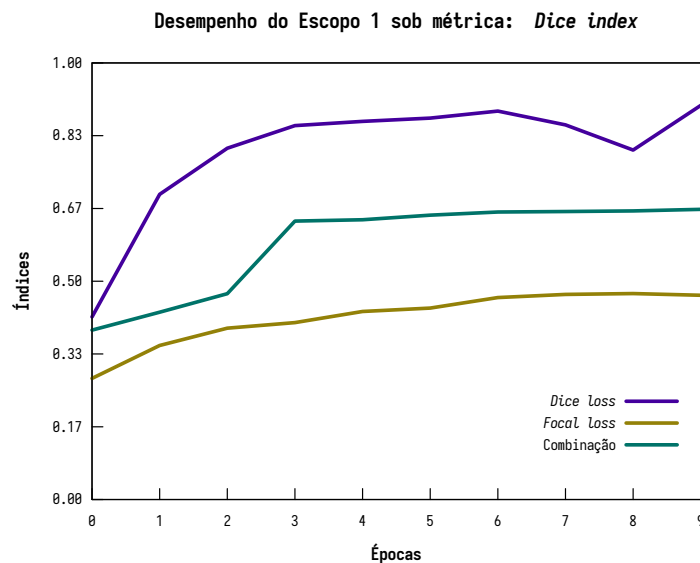


Figura 4.3: Desempenho do modelo treinado no Escopo 1 sob a métrica DSC ao considerar diferentes funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).

4.3 Escopo 2: Firmando a aprendizagem

Em decorrência das configurações descritas na Seção 3.5.3, os resultados aqui apresentados retratam o treinamento de 50 épocas (total disponibilizado). Deste modo, a Figura 4.11 traz a evolução gráfica do treinamento do modelo sob a ótica das métricas DSC, *MeanIoU* e *F1-Score* e da função de perda Combinação C_{FP} (Equação 3.2). Ao considerar todas as classes de segmentação semântica, tal evolução traduziu-se numericamente para: DSC de 1,0247, *MeanIoU* de 0,9326, *F1-Score* de 0,7570 e Combinação C_{FP} de 0,2665.

Ponderando visualmente, o modelo obteve previsões de qualidade semelhante às exemplificadas na Figura 4.12. Para a renderização, tanto das anotações quanto das previsões, foram adotados os índices do maior valor disponível (Equação 3.3).

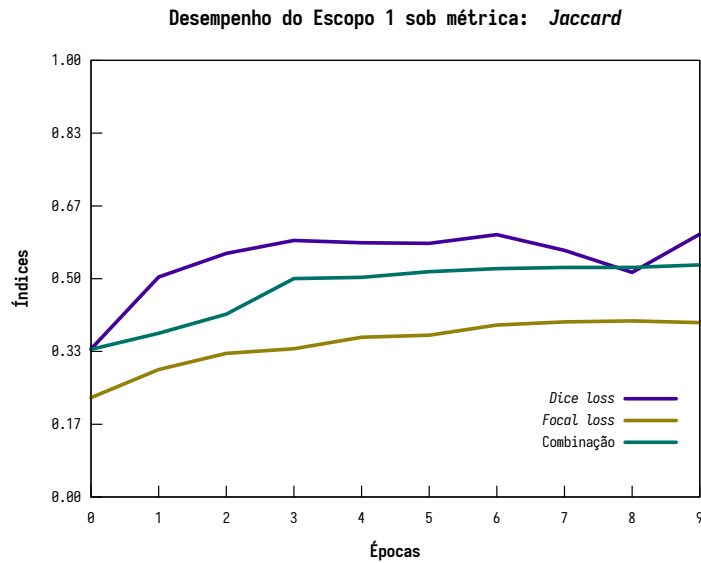


Figura 4.4: Desempenho do modelo treinado no Escopo 1 sob a métrica *Jaccard* ao considerar diferentes funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).

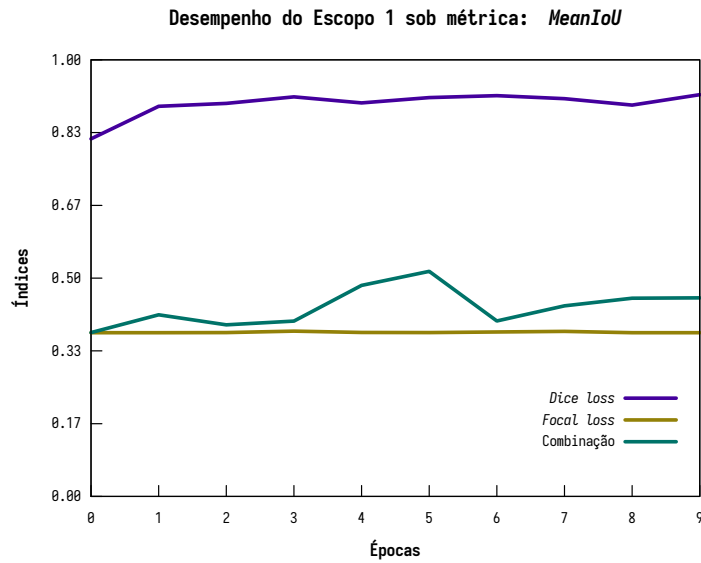


Figura 4.5: Desempenho do modelo treinado no Escopo 1 sob a métrica *MeanIoU* ao considerar diferentes funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).

4.4 O que pôde ser aprendido

Frente aos resultados do Escopo 0 foi possível entender um provável impacto do redimensionamento das imagens. As IRMs disponibilizadas pela base de dados não estão sob as

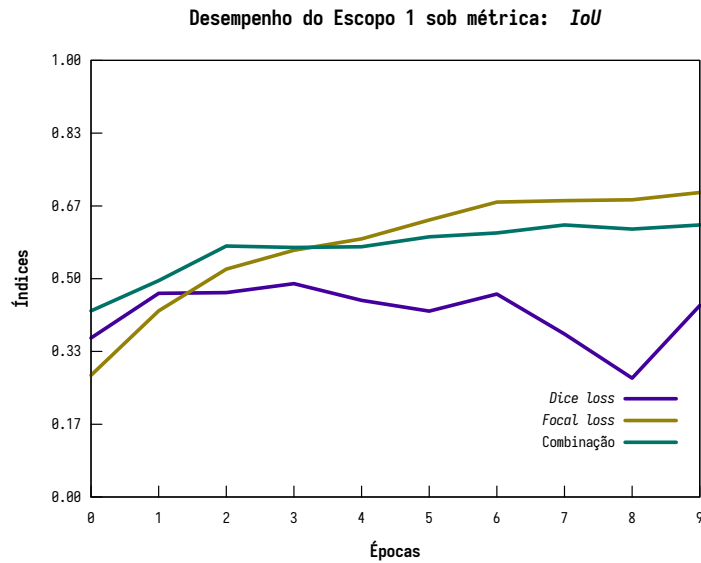


Figura 4.6: Desempenho do modelo treinado no Escopo 1 sob a métrica IoU ao considerar diferentes funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).

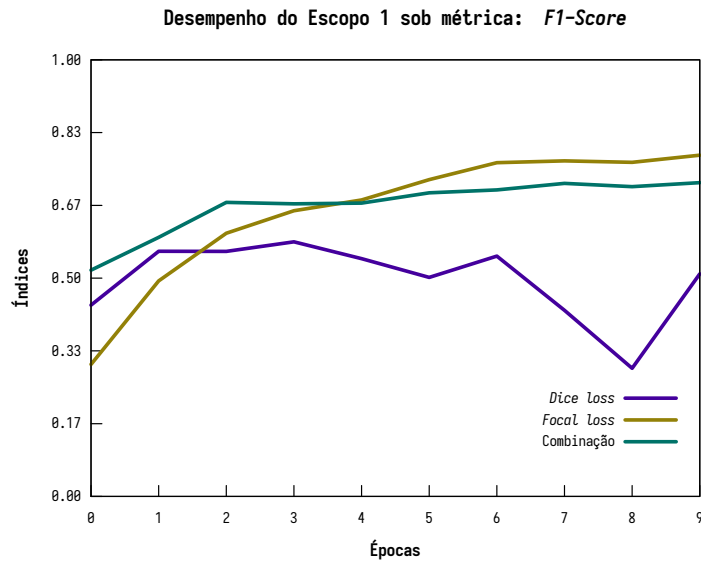


Figura 4.7: Desempenho do modelo treinado no Escopo 1 sob a métrica $F1-Score$ ao considerar diferentes funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).

mesmas dimensões. Uma estrutura de segmentador semântico de imagens pode manipular entradas de tamanhos diversos, mas esta estratégia não favorece seu aprendizado. Aplicou-se redimensionamento com interpolação de área sem tratamento a posteriori sobre

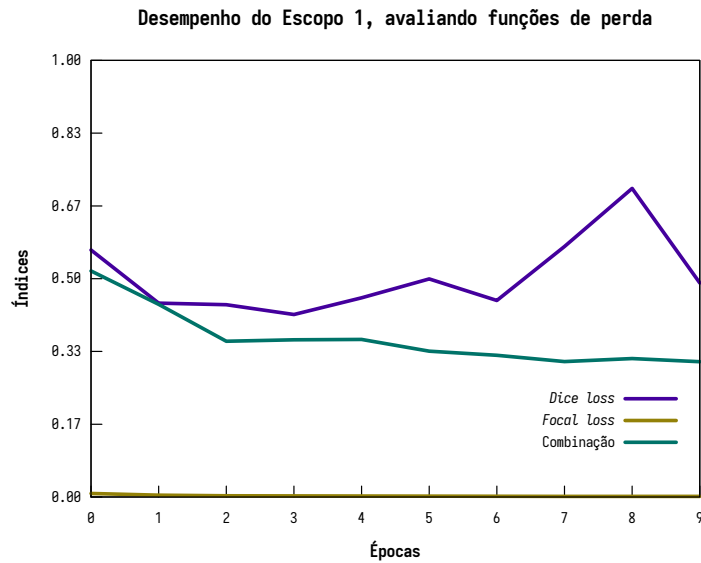


Figura 4.8: Desempenho do modelo treinado no Escopo 1 com avaliação direta entre as funções de perda: DiL (em roxo), FL (em ocre) e a Combinação C_{FP} (Equação 3.2, em verde).



Figura 4.9: Comparativo visual de uma predição (à direita) do modelo treinado no Escopo 1 dadas as anotações (ao centro) e a IRM original (à esquerda) de referência, tendo como função de perda a DiL.



Figura 4.10: Comparativo visual de uma predição (à direita) do modelo treinado no Escopo 1 dadas as anotações (ao centro) e a IRM original (à esquerda) de referência, tendo como função de perda a FL.

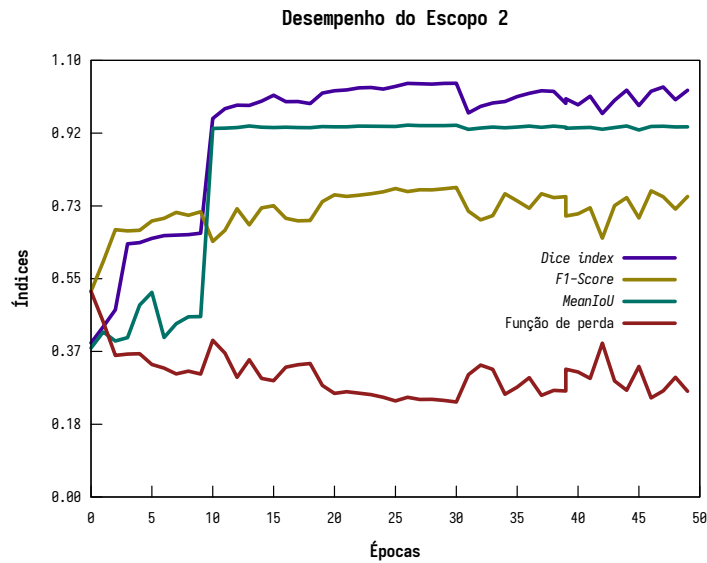


Figura 4.11: Desempenho do modelo treinado no Escopo 2 através das métricas DSC (em roxo), $F1$ -Score (em ocre) e $MeanIoU$ (em verde) e da função de perda Combinação C_{FP} (Equação 3.2, em bordô).

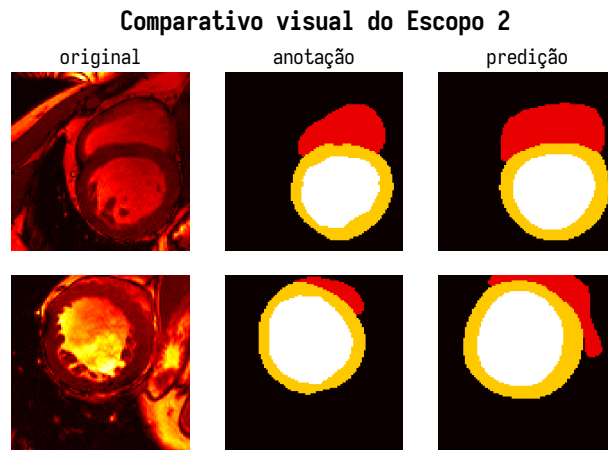


Figura 4.12: Comparativo visual de duas predições (à direita) do modelo treinado no Escopo 2 dadas as anotações (ao centro) e as IRMs originais (à esquerda) de referência.

imagens, a fim de padronizar o tamanho das IRMs e suas anotações às dimensões utilizadas por Mortazi e Bagci (2018) (200×200). Contudo artefatos devem ter sido criados e, portanto, acarretaram o distanciamento do entendimento do agente da realidade. A falta de contornos definidos presente na predição da Figura 4.2 demonstra a incerteza em seus valores, em virtude do método de renderização adotado (Equação 3.1). Em cenário de certeza, os contornos estariam definidos mesmo que errôneos, vide anotação da Figura 4.2. A soma da ponderação expõe extratos de valores inteiros como contornos bem definidos

e os decimais como contornos borrados.

Ainda em consequência disso, o agente demonstrou ter aprendido a caracterizar somente o tamanho médio e a localização média das classes anotadas por sempre disponibilizar a mesma predição, independentemente da imagem de entrada. O agente deve ter adotado esta estratégia justamente pela dificuldade em estabelecer as classes, utilizando valores decimais.

Entende-se que a Entropia cruzada sinaliza a diferença entre dois conjuntos de distribuição de probabilidades. Tais conjuntos devem dispor seus elementos sob as mesmas regras de ordenação e sob dimensões arbitrárias e compartilhadas. Com os elementos ordenados é realizada uma comparação entre um elemento-base (pertencente ao primeiro conjunto) e um elemento-objetivo (pertencente ao segundo conjunto). À vista disso, os conjuntos idênticos são avaliados a 0, enquanto que qualquer outro cenário apresentará uma avaliação não nula, podendo ser superior a 1,0. Quanto maior for esta avaliação, maior será a diferença detectada. No contexto das anotações e predições, os elementos caracterizam-se como o extrato das classes anotadas para cada *pixel* das mesmas.

Compreende-se, o oposto para a métrica DSC, uma vez que esta sinaliza a coincidência entre dois conjuntos binários. Tais conjuntos dispõem seus elementos sob as mesmas regras de ordenação e sob dimensões arbitrárias e compartilhadas. Com os elementos ordenados é realizada uma comparação entre a quantidade de correspondências verdadeiras individuais e a quantidade de elementos verdadeiros de ambos conjuntos separadamente. É uma métrica baseada em porções volumétricas. À vista disso, mesmo que os conjuntos compartilhem a parcela de elementos verdadeiros se estiverem dispostos em sequência disjunta são avaliados a 0; o completo oposto, avaliados a 1,0, acontece àqueles idênticos. Ademais, conjuntos estratificados (com duas ou mais classes anotadas) refletem as avaliações de suas camadas.

Uma sinalização similar é realizada pela métrica *MeanIoU*. Assemelha-se ao julgar as correspondências verdadeiras individuais, mas diverge ao examinar os elementos verdadeiros de ambos conjuntos concomitantemente. Correspondendo à parte *IoU*, também baseada em porções volumétricas. Para a análise completa, lida-se com conjuntos estratificados, mesmo que não o sejam originalmente, ao pesar as análises individuais de seus estratos.

Desta maneira, a observância de valores altos para a métrica DSC e de valores baixos para a função de perda, Entropia cruzada, vide Figura 4.1, sinalizaria predições realmente próximas às anotações originais. Como as predições não correspondiam à realidade e não se tem conhecido do método de ajuste dimensional adotado por Mortazi e Bagci (2018), interpreta-se o impacto como sendo reflexo do redimensionamento adotado sobre tal incongruência, justamente pela provável inserção de valores não reais às IRMs redi-

mencionadas. Somente a métrica *MeanIoU* traduziu equivalência entre seus valores e o resultado visual.

Relativo aos resultados do Escopo 1 foi possível entender as caracterizações análogas entre as diversas métricas postas em análise. As métricas DSC, *Jaccard index*, *MeanIoU*, *IoU* e *F1-Score* compartilham a sinalização de coincidência entre dois conjuntos a partir de manipulações de porções volumétricas. Em especial, a métrica DSC equivale à *F1-Score* e a *IoU* equivale à *Jaccard index*, logo espera-se significância difundida entre elas. Todavia, foi a métrica DSC que apresentou uma evolução semelhante à *Jaccard index* (agente treinado com DiL com índices superiores à Combinação C_{FP} da Equação 3.2 que, por sua vez, são superiores ao treinado com a FL), enquanto que a *F1-Score* assemelhou-se à *IoU* apresentou uma evolução diferente (agente treinado com DiL com índices inferiores à Combinação C_{FP} da Equação 3.2 que, por sua vez, são superiores ao treinado com a FL até a 4° época e invertendo para as demais). Somente a *MeanIoU* apresentou uma evolução distinta das demais.

Percebe-se que uma sinalização similar à Entropia cruzada é realizada pela função de perda FL, por ser o acréscimo de um fator de ponderação sobre a primeira. Com segmentações semânticas de multi-classes podem existir anotações balanceadas e desbalanceadas, onde: as balanceadas são aquelas em que cada classe detém aproximadamente $1/N$ de elementos verdadeiros, onde N é a quantidade de classes anotadas, enquanto que desbalanceadas fogem desta proporção. Tal ponderação introduzida pela FL reduz o impacto das classes mais fáceis de serem classificadas na composição de seu resultado, favorecendo, assim, as mais desbalanceadas.

Uma sinalização similar é realizada pela função de perda DiL. Seu comportamento é complementar à métrica DSC. Ao passo que a métrica se aproxima de 1,0, a função de perda se aproxima de 0.

Isto posto, a observância de valores médios para a função de perda DiL, vide Figura 4.8, sinalizaria predições próximas às anotações originais com algumas discrepâncias. Com ainda menos discrepância eram esperados os valores com a FL, por valores próximos a zero. Tais comportamentos foram comprovados nas Figuras 4.9 a 4.10. Além disso, a DiL apresentou bons resultados para as classes com maiores representações, visto que a classe **1** não foi anotada na predição exemplificada por ser a mais desbalanceada; o treinamento expôs um comportamento oscilante. Enquanto que a FL expôs um comportamento regular e apresentou uma predição praticamente idêntica à anotação original, justamente por ponderar sobre o desbalanceamento das classes. Por último mas não menos importante, tem-se a observância de um comportamento mais regular, mesmo que não seja tão próximo a zero quanto a FL, com a combinação das funções de perda (Equação 3.2); harmonizando os pontos positivos de suas partes.

Frente aos resultados do Escopo 2 foi possível observar tendências a bons resultados, mas com bastante oscilação. Conforme discutido na análise do escopo anterior, o comportamento oscilante expressado pela DiL foi replicado para este escopo, por ser componente da combinação C_{FP} (Equação 3.2). Este comportamento instável pode ser reflexo da alta taxa de aprendizado adotada ($1 \cdot 10^{-3}$) no treinamento.

A taxa de aprendizagem refere-se ao ajuste realizado sobre os pesos aprendidos durante o treinamento num mini conjunto de entradas. Em outras palavras, o agente não treina com todo o conjunto de treinamento de uma vez, seja por não ter espaço para manipular toda essa massa de dados seja para garantir passos dos aprendizagem, com isso, divide-se em mini grupos e estes são utilizados como uma rodada de treinamento. Ao final desta rodada, o agente compara sua predição com a anotação desejada e resulta na avaliação da sua atual aprendizagem, que por sua vez, é ajustada pela taxa de aprendizagem. Com este ajuste, inicia-se o processo de *backpropagation* do agente, isto é, a aprendizagem de fato. Percebe-se, portanto, que altas taxas de aprendizagem influenciam na convergência acelerada aos pesos otimizados, contudo, ao alcançar o cenário onde o real ajuste demandado deva ser inferior à taxa empregada tem-se o surgimento de oscilação ao redor deste ajuste demandado.

Posto isso, como a arquitetura foi preparada para reduzir a taxa de aprendizagem frente à estabilidade de 3 épocas no treinamento e isso não ocorreu, a taxa foi mantida até a última época. Corroborando com sua influência sobre a oscilação apresentada na evolução do agente, ilustrada na Figura 4.11.

Ainda a partir da Figura 4.11, contempla-se saltos abruptos nas métricas DSC e *MeanIoU* na 10ª época de treinamento, tendo como provável causa uma instabilidade no fornecimento de energia que ocorreu durante o treinamento, por ter sido o único sinistro ocorrido que se teve notícia. Ademais estas zonas de saltos, houve uma evolução típica de estratégias de AM: argumentando em defesa da interpretação a pouco realizada. Tal instabilidade ocorreu durante o processamento de duas épocas do treinamento: 10ª e 30ª. A quantidade de energia armazenada em baterias não foi suficiente para manter o treinamento ininterrupto.

Há de se manter atenção ao valor do DSC superior a 1,0, teoricamente impossível de acontecer. Porém não foram encontrados evidências de erro na manipulação dos conjuntos no cálculo desta métrica, tampouco houve tempo hábil para investigar uma possível correção, ainda mais ao ponderar sobre sua correção de seu cálculo sobre conjuntos menores de dados. Entretanto, ao analisar uma coletânea de dados diminutos e fora do contexto deste trabalho foi possível “replicar” tal comportamento. Assumindo os conjuntos $A = \{0, 1, 1, 0, 1, 1, 0, 0\}$ e $B = \{0, 1.2, 1.8, 0, 0.5, 0.7, 0, 0\}$ obtêm-se o DSC de 1,0244, também superior a 1,0. Este infortúnio aconteceu devido a valores maiores que 1 no con-

junto B , no entanto este cenário é impedido com a utilização da *Softmax* como função ativadora para disponibilização das predições por parte do agente; logo esta observância de valores continua sem explicação. Ressalta-se, também, que durante os cálculos internos do treinamento, as anotações só continham valores 0s e 1s e as predições mantinham seu espectro de valores entre esses dois.

Em suma, o agente proposto mostrou-se com bons resultados visuais também, conforme ilustrado na Figura 4.12. As classes anotadas de maior incidência (classes **2** e **3**) tiveram menos discrepância ao comparar com a classe de menor incidência (classe **1**). Isso estava dentro do esperado, uma vez que no escopo anterior (Figuras 4.9 a 4.10) tanto a DiL quanto a FL obtiveram bons resultados nas classes **2** e **3**, ao passo que para a classe **1** há, praticamente, o reflexo da FL na combinação C_{FP} (Equação 3.2). Reforçando a importância da combinação adotada.

Capítulo 5

Conclusões

A tecnologia aplicada a imagens médicas vem desempenhando um papel muito importante ao visar processos e resultados mais eficientes, precisos e confiáveis, particularmente para o processamento, a análise e o entendimento correto destas imagens (PANAYIDES et al., 2020b). Segundo Ronneberger, Fischer e Brox (2015), a utilização de *Deep Learning* (DL) na segmentação de tais imagens está em destaque, por proporcionarem resultados tão precisos quanto, ou melhores, aos de especialistas e numa janela de tempo menor.

Quando se trata de segmentação de imagens com uso de estratégias de Aprendizagem de Máquinas, em especial o *Deep Learning* (DL), já é difundido o emprego de variantes da estrutura *encoder-decoder*, conhecida como U-Net (RONNEBERGER; FISCHER; BROX, 2015). Tais variantes podem apresentar-se como uma tonificação sobre as *skip connections*, como realizado por Zhou et al. (2018), visto sua comprovação na melhora da recuperação do dados espaciais durante a reconstrução da imagem segmentada. Há ainda as que reforçam as estruturas internas tanto do *encoder* quanto do *decoder* em detrimento da primeira opção, porém com o mesmo intuito, como foi o caso de Mortazi e Bagci (2018).

Além da definição da estrutura do segmentador de imagens a ser manuseado, é preciso determinar os melhores hiperparâmetros para esta estrutura (MORTAZI; BAGCI, 2018). Isto é, escolher ou selecionar a quantidade de filtros presentes numa camada convolucional, a dimensionalidade de tais filtros, as funções de *pooling* a serem realizadas e até o tamanho de tais saltos são exemplos de hiperparâmetros para um segmentador Rede Neuronal Completamente Convolucional (RNCC). Esta escolha ou seleção repercute no desempenho do modelo.

Com isso em mente, Mortazi e Bagci (2018) realizaram uma busca automatizada pelos hiperparâmetros da sua estrutura-base proposta, ilustrada na Figura 2.10, otimizados à segmentação semântica de imagens médicas. Resultando, por fim, nos valores e funções ilustrados na Figura 3.2.

Isso posto, este trabalho objetivou o estudo sobre a arquitetura final sugerida por Mor-tazi e Bagci (2018) no contexto de Imagem de Ressonância Magnética cardíacas. Tal estudo relaciona-se inclusive com as configurações e condições de treinamento de um modelo que faça uso desta arquitetura. Por isso, admitiu-se alterações relacionadas à pesquisa inspiradora.

A progressão apresentada ao longo dos escopos explorados trouxeram o esclarecimento, o aperfeiçoamento e, inclusive, a surpresa de técnicas de treinamento de modelos focados em segmentação semântica de imagens. Podendo elencá-los como os seguintes aspectos:

- **Ajustes dimensionais dos dados de entrada:** ao manipular modelos com dimensões dos dados de entrada e de saída fixados e conhecidos é preciso garantir o respeito a estas restrições. Portanto, é comum recorrer à técnica de redimensionamento das imagens à disposição às dimensões desejadas, por figurar entre as manipulações básicas de imagens; tal abordagem cria artefatos que, se não tratados, afastam o modelo do entendimento mais adequado. Em contra partida, a estratégia de cortes deslizantes mantém os dados originais e intocáveis quanto seus valores. Este processo recorta um pedaço nas dimensões almeçadas das imagens e, então, desloca-se por esta imagem e recorta um outro pedaço, desde que as dimensões sejam respeitadas, repetindo este ciclo até toda a imagem tenha sido exaurida. Mesmo que haja trechos “repetidos” entre os recortes, estes nunca serão idênticos. Assim sendo, a utilização de cortes deslizantes é operação correta de ser feita;
- **Dimensionalidade dos dados:** o cuidado do item anterior é válido apenas para os casos onde os dados originais sejam iguais ou maiores às dimensões manipuladas pelo modelo, o contrário não deve existir: criação de artefatos. Seja por restrição de *hardware* seja por qualquer outro motivo, as menores dimensões passíveis de uso devem ser maiores que a maior área ocupada pelas classes anotadas combinadas, desconsiderando a de plano de fundo, para garantir que um recorte não esteja inserido em apenas uma das classe;
- **Adequação das métricas consideradas:** a avaliação de desempenho acontece em diversas áreas do conhecimento e, portanto, não seria diferente no contexto de segmentação de imagens. Entretanto, a métrica acurácia, aplicável em diversas estratégias de AM, não foi observada por este trabalho, uma vez que não é adequada à problemática enfrentada. Embora haja contexto de porções volumétricas, tal métrica é agnóstica aos desbalanceamento entre as classes anotadas, ou seja, um modelo que se torne especialista em segmentar plano de fundo apresentará bons resultados nesta ótica, o que não estaria próximo à realidade de fato. Por isso foram consideradas diversas métricas supostamente adequadas, mesmo que fossem

muito similares em alguns pontos, e reconhece-se a adequabilidade das métricas DSC e *MeanIoU* para esta aplicação;

- **Evolução das funções de perda:** é natural que o avanço do conhecimento sobre as estratégias de AM propicie a evolução das métricas e das funções de perda. No que diz respeito à segunda delas, a confiança sobre os valores numéricos surgiu ao utilizar a *Dice Loss* (DiL) (inverso e complementar à DSC) e a *Focal Loss* (FL) (evolução da Entropia cruzada), por ser derivada de uma métrica adequada à esta aplicabilidade e por ser ciente do desbalanceamento das classes anotadas respectivamente. Não obstante às suas avaliações isoladamente, a possibilidade de combinação entre funções de perda mostrou-se ser de grande utilidade e pode se tornar uma prática comum para o futuro. Dessa maneira, ao invés de investigar funções de perda que considerem os múltiplos e complexos aspectos conhecidos e desconhecidos das diversas aplicações, é possível investigar aquelas que considerem aspectos específicos mas que juntas consigam ponderar sobre a visão geral;
- **Atenção aos resultados numéricos:** corroborando com as duas conclusões anteriores (*Adequação das métricas consideradas* e *Evolução das funções de perda*), observou-se que os resultados numéricos apresentados no Escopo 0 (Seção 4.1) são suficientes para serem classificados como muito satisfatórios. Com métrica muito próxima de 1,0000 e a função de perda ainda mais próxima de 0,0000. Todavia, os resultados visuais estavam distantes de serem considerados usáveis, sendo assim talvez seja preciso estar atento a valores numéricos muito bons, mesmo que de métricas a priori adequadas, pela possibilidade de falsa correspondência com a realidade.

Com relação ao segmentador de imagens proposto (Seção 3.5.3), os ajustes realizados na arquitetura proposta por Mortazi e Bagci (2018), como o tamanho dos saltos das operações de *pooling* de 2 para 5 e a dimensão das imagens de entrada de 200×200 para 100×100 , embora tenham sido influenciados em grande parte pelo *hardware* disponível, as técnicas de treinamento supracitadas também foram observadas. Com isso, tem-se um modelo capaz de ser comparado com a sua inspiração. Ao passo que a inspiração manifestou um DSC de 0,872 para as predições cruas e 0,882 para as predições com tratamento pós-segmentação, o modelo treinado manifestou 1,0247. Com todos os resultados consideraram a média entre as classes de segmentações, o modelo proposto apresentou-se numericamente melhor que sua inspirações e apresentou bons resultados visuais; validando-se, assim, a evolução entre os modelos.

Tendo posto a evolução entre os modelos, este segmentador de imagens médias desencadeia outras investigações e explorações sobre o tema, a fim de garantir sua evolução

e aprimoramento. Os seguintes questionamentos foram amadurecidos frente o trabalho realizado:

- Como este modelo se comportaria frente outras bases de dados?
- Qual o impacto do esclarecimento do desbalanceamento das classes anotadas?
- Qual o impacto da utilização de outros fatores γ na *Focal Loss* (FL)?
- Qual o impacto de ponderações distintas na combinação C_{FP} ?
- Quais outras funções de perda possuem relevância nesta aplicação?

Referências

ABADI, M.; AGARWAL, A.; BARHAM, P.; BREVDO, E.; CHEN, Z.; CITRO, C.; CORRADO, G. S.; DAVIS, A.; DEAN, J.; DEVIN, M.; GHEMAWAT, S.; GOODFELLOW, I.; HARP, A.; IRVING, G.; ISARD, M.; JIA, Y.; JOZEFOWICZ, R.; KAISER, L.; KUDLUR, M.; LEVENBERG, J.; MANÉ, D.; MONGA, R.; MOORE, S.; MURRAY, D.; OLAH, C.; SCHUSTER, M.; SHLENS, J.; STEINER, B.; SUTSKEVER, I.; TALWAR, K.; TUCKER, P.; VANHOUCKE, V.; VASUDEVAN, V.; VIÉGAS, F.; VINYALS, O.; WARDEN, P.; WATTENBERG, M.; WICKE, M.; YU, Y.; ZHENG, X. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. Software available from tensorflow.org. Disponível em: <<https://www.tensorflow.org/>>.

ALFAKIH, K.; PLEIN, S.; THIELE, H.; JONES, T.; RIDGWAY, J.; SIVANANTHAN, M. Normal human left and right ventricular dimensions for MRI as assessed by turbo gradient echo and steady-state free precession imaging sequences. *JOURNAL OF MAGNETIC RESONANCE IMAGING*, 17, n. 3, p. 323–329, MAR 2003. ISSN 1053-1807.

BADRINARAYANAN, V.; KENDALL, A.; CIPOLLA, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 39, n. 12, p. 2481–2495, DEC 2017. ISSN 0162-8828.

BERNARD, O.; LALANDE, A.; ZOTTI, C.; CERVENANSKY, F.; YANG, X.; HENG, P.-A.; CETIN, I.; LEKADIR, K.; CAMARA, O.; BALLESTER, M. A. G.; SANROMA, G.; NAPEL, S.; PETERSEN, S.; TZIRITAS, G.; GRINIAS, E.; KHENED, M.; KOLLERATHU, V. A.; KRISHNAMURTHI, G.; ROHÉ, M.-M.; PENNEC, X.; SERMESANT, M.; ISENSEE, F.; JÄGER, P.; MAIER-HEIN, K. H.; FULL, P. M.; WOLF, I.; ENGELHARDT, S.; BAUMGARTNER, C. F.; KOCH, L. M.; WOLTERINK, J. M.; İSGUM, I.; JANG, Y.; HONG, Y.; PATRAVALI, J.; JAIN, S.; HUMBERT, O.; JODOIN, P.-M. Deep Learning Techniques for Automatic MRI Cardiac Multi-Structures Segmentation and Diagnosis: Is the Problem Solved? *IEEE Transactions on Medical Imaging*, v. 37, n. 11, p. 2514–2525, 2018. Disponível em: <<https://doi.org/10.1109/TMI.2018.2837502>>.

BRETT, M.; MARKIEWICZ, C. J.; HANKE, M.; CÔTÉ, M.-A.; CIPOLLINI, B.; MCCARTHY, P.; JARECKA, D.; CHENG, C. P.; HALCHENKO, Y. O.; COTTAAR, M.; LARSON, E.; GHOSH, S.; WASSERMANN, D.; GERHARD, S.; LEE, G. R.; WANG, H.-T.; KASTMAN, E.; KACZMARZYK, J.; GUIDOTTI, R.; DUEK, O.; DANIEL, J.; ROKEM, A.; MADISON, C.; MOLONEY, B.; MORENCY, F. C.;

GONCALVES, M.; MARKELLO, R.; RIDDELL, C.; BURNS, C.; MILLMAN, J.; GRAMFORT, A.; LEPPÄKANGAS, J.; SÓLON, A.; BOSCH, J. J. van den; VINCENT, R. D.; BRAUN, H.; SUBRAMANIAM, K.; GORGOLEWSKI, K. J.; RAAMANA, P. R.; KLUG, J.; NICHOLS, B. N.; BAKER, E. M.; HAYASHI, S.; PINSARD, B.; HASELGROVE, C.; HYMERS, M.; ESTEBAN, O.; KOUDORO, S.; PÉREZ-GARCÍA, F.; OOSTERHOF, N. N.; AMIRBEKIAN, B.; NIMMO-SMITH, I.; NGUYEN, L.; REDDIGARI, S.; ST-JEAN, S.; PANFILOV, E.; GARYFALLIDIS, E.; VAROQUAUX, G.; LEGARRETA, J. H.; HAHN, K. S.; HINDS, O. P.; FAUBER, B.; POLINE, J.-B.; STUTTERS, J.; JORDAN, K.; CIESLAK, M.; MORENO, M. E.; HAENEL, V.; SCHWARTZ, Y.; BARATZ, Z.; DARWIN, B. C.; THIRION, B.; GAUTHIER, C.; ORFANOS, D. P.; SOLOVEY, I.; GONZALEZ, I.; PALASUBRAMANIAM, J.; LECHER, J.; LEINWEBER, K.; RAKTIVAN, K.; CALÁBKOVÁ, M.; FISCHER, P.; GERVAIS, P.; GADDE, S.; BALLINGER, T.; ROOS, T.; REDDAM, V. R.; FREEC84. *nipy/nibabel: 3.2.1*. Zenodo, 2020. Disponível em: <<https://doi.org/10.5281/zenodo.4295521>>.

CARNEIRO, G.; NASCIMENTO, J.; FREITAS, A. Robust left ventricle segmentation from ultrasound data using deep neural networks and efficient search methods. In: *2010 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. [S.l.: s.n.], 2010. p. 1085–1088.

CARNEIRO, G.; NASCIMENTO, J.; FREITAS, A. The segmentation of the left ventricle of the heart from ultrasound data using deep learning architectures and derivative-based search methods. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, v. 21, p. 968–82, 09 2011.

CHEN, C.; QIN, C.; QIU, H.; TARRONI, G.; DUAN, J.; BAI, W.; RUECKERT, D. Deep learning for cardiac image segmentation: A review. *Frontiers in Cardiovascular Medicine*, v. 7, p. 25, 2020. ISSN 2297-055X. Disponível em: <<https://www.frontiersin.org/article/10.3389/fcvm.2020.00025>>.

CHEN, M.; FANG, L.; LIU, H. Fr-net: Focal loss constrained deep residual networks for segmentation of cardiac mri. In: *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. [S.l.: s.n.], 2019. p. 764–767.

CHOLLET, F. et al. *Keras*. 2015. Último acesso em 17 de setembro de 2021. Disponível em: <<https://keras.io>>.

COCOSCO, C. A.; NETSCH, T.; SÉNÉGAS, J.; BYSTROV, D.; NIESSEN, W. J.; VIERGEVER, M. A. Automatic cardiac region-of-interest computation in cine 3D structural MRI. *International Congress Series*, v. 1268, n. Complete, p. 1126–1131, 2004.

COLAB, G. *Colaboratory - Frequently Asked Questions*. Google, 2021. Último acesso em 17 de setembro de 2021. Disponível em: <<https://research.google.com/colaboratory/faq.html>>.

COVER, T. M.; THOMAS, J. A. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. USA: Wiley-Interscience, 2006. ISBN 0471241954.

DICE, L. R. Measures of the Amount of Ecologic Association Between Species. *Ecology*, 26, n. 3, p. 297–302, 1945. ISSN 0012-9658. Disponível em: <<https://doi.org/10.2307/1932409>>.

HARRIS, C. R.; MILLMAN, K. J.; WALT, S. J. van der; GOMMERS, R.; VIRTANEN, P.; COURNAPEAU, D.; WIESER, E.; TAYLOR, J.; BERG, S.; SMITH, N. J.; KERN, R.; PICUS, M.; HOYER, S.; KERKWIJK, M. H. van; BRETT, M.; HALDANE, A.; RÍO, J. F. del; WIEBE, M.; PETERSON, P.; GÉRARD-MARCHANT, P.; SHEPPARD, K.; REDDY, T.; WECKESSER, W.; ABBASI, H.; GOHLKE, C.; OLIPHANT, T. E. Array programming with NumPy. *Nature*, Springer Science and Business Media LLC, v. 585, n. 7825, p. 357–362, set. 2020. Disponível em: <<https://doi.org/10.1038/s41586-020-2649-2>>.

HUTTENLOCHER, D.; KLANDERMAN, G.; RUCKLIDGE, W. Comparing Images Using the Hausdorff Distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15, n. 9, p. 850–863, SEP 1993. ISSN 0162-8828. Disponível em: <<https://doi.org/10.1109/34.232073>>.

IEEE Spectrum. *Top Programming Languages 2021*. 2021. Último acesso em 17 de setembro de 2021. Disponível em: <<https://spectrum.ieee.org/top-programming-languages/#toggle-gdpr>>.

Interstellar Engine. *Intersection Over Union*. 2021. Último acesso em 30 de outubro de 2021. Disponível em: <<https://www.interstellarengine.com/ai/Intersection-Over-Union.html>>.

JANG, Y.; HONG, Y.; HA, S.; KIM, S.; CHANG, H.-J. Automatic segmentation of lv and rv in cardiac mri. In: _____. [S.l.: s.n.], 2018. p. 161–169. ISBN 978-3-319-75540-3.

JOLLY, M.-P. Automatic Recovery of the Left Ventricular Blood Pool in Cardiac Cine MR Images. In: Metaxas, D and Axel, L and Fichtinger, G and Szekely, G (Ed.). *MEDICAL IMAGE COMPUTING AND COMPUTER-ASSISTED INTERVENTION - MICCAI 2008, PT I, PROCEEDINGS*. [S.l.], 2008. (Lecture Notes in Computer Science, 5241), p. 110–118. ISBN 978-3-540-85987-1. ISSN 0302-9743. 11th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI2008), New York, NY, SEP 06-10, 2008.

KHENDED, M.; KOLLERATHU, V. A.; KRISHNAMURTHI, G. Fully convolutional multi-scale residual densenets for cardiac segmentation and automated cardiac diagnosis using ensemble of classifiers. *Medical Image Analysis*, v. 51, p. 21–45, 2019. ISSN 1361-8415. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S136184151830848X>>.

KLUYVER, T.; RAGAN-KELLEY, B.; PEREZ, F.; GRANGER, B.; BUSSONNIER, M.; FREDERIC, J.; KELLEY, K.; HAMRICK, J.; GROUT, J.; CORLAY, S.; IVANOV, P.; AVILA, D.; ABDALLA, S.; WILLING, C.; TEAM, J. D. Jupyter Notebooks—a publishing format for reproducible computational workflows. In: Loizides, F and Schmidt, B (Ed.). *POSITIONING AND POWER IN ACADEMIC PUBLISHING: PLAYERS, AGENTS AND AGENDAS*. 2016. p. 87–90. ISBN

978-1-61499-649-1; 978-1-61499-648-4. 20th International Conference on Electronic Publishing, Univ Gottingen, Gottingen, GERMANY, JUN 07-09, 2016. Disponível em: <<https://doi.org/10.3233/978-1-61499-649-1-87>>.

LAROBINA, M.; MURINO, L. Medical image file formats. *Journal of Digital Imaging*, v. 27, n. 2, p. 200–206, April 2014. ISSN 1618-727X. Disponível em: <<https://doi.org/10.1007/s10278-013-9657-9>>.

LIN, T.-Y.; GOYAL, P.; GIRSHICK, R.; HE, K.; DOLLÁR, P. Focal loss for dense object detection. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. [s.n.], 2017. p. 2999–3007. Disponível em: <<https://doi.org/10.1109/ICCV.2017.324>>.

MITCHELL, T. M. *Machine Learning*. New York: McGraw-Hill, 1997. ISBN 978-0-07-042807-2.

MONTEUX, A. *Metrics for semantic segmentation*. 2019. Último acesso em 30 de outubro de 2021. Disponível em: <<https://ilmonteux.github.io/2019/05/10/segmentation-metrics.html>>.

MORTAZI, A.; BAGCI, U. *Automatically Designing CNN Architectures for Medical Image Segmentation*. 2018. Disponível em: <<https://arxiv.org/abs/1807.07663>>.

Organização Pan-Americana da Saúde. *Doenças cardiovasculares*. 2021. Último acesso em 17 de outubro de 2021. Disponível em: <<https://www.paho.org/pt/topicos/doencas-cardiovasculares>>.

PANAYIDES, A. S.; AMINI, A.; FILIPOVIC, N. D.; SHARMA, A.; TSAFTARIS, S. A.; YOUNG, A.; FORAN, D.; DO, N.; GOLEMATI, S.; KURC, T.; HUANG, K.; NIKITA, K. S.; VEASEY, B. P.; ZERVAKIS, M.; SALTZ, J. H.; PATTICHIS, C. S. AI in Medical Imaging Informatics: Current Challenges and Future Directions. *IEEE Journal of Biomedical and Health Informatics*, v. 24, n. 7, p. 1837–1857, 2020. Disponível em: <<https://doi.org/10.1109/JBHI.2020.2991043>>.

PANAYIDES, A. S.; AMINI, A.; FILIPOVIC, N. D. et al. AI in Medical Imaging Informatics: Current Challenges and Future Directions. *IEEE Journal of Biomedical and Health Informatics*, v. 24, n. 7, p. 1837–1857, 2020.

PETITJEAN, C.; DACHER, J.-N. A review of segmentation methods in short axis cardiac mr images. *Medical Image Analysis*, v. 15, n. 2, p. 169–184, 2011. ISSN 1361-8415. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1361841510001349>>.

PETITJEAN, C.; ZULUAGA, M. A.; BAI, W.; DACHER, J.-N.; GROSGEORGE, D.; CAUDRON, J.; RUAN, S.; AYED, I. B.; CARDOSO, M. J.; CHEN, H.-C.; JIMENEZ-CARRETERO, D.; LEDESMA-CARBAYO, M. J.; DAVATZIKOS, C.; DOSHI, J.; ERUS, G.; MAIER, O. M.; NAMBAKSH, C. M.; OU, Y.; OURSELIN, S.; PENG, C.-W.; PETERS, N. S.; PETERS, T. M.; RAJCHL, M.; RUECKERT, D.; SANTOS, A.; SHI, W.; WANG, C.-W.; WANG, H.; YUAN, J. Right ventricle segmentation from cardiac MRI: A collation study. *Medical*

Image Analysis, v. 19, n. 1, p. 187–202, 2015. ISSN 1361-8415. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1361841514001509>>.

REZATOFIGHI, H.; TSOI, N.; GWAK, J.; SADEGHIAN, A.; REID, I.; SAVARESE, S. Generalized intersection over union. June 2019.

RONNEBERGER, O.; FISCHER, P.; BROX, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In: NAVAB, N.; HORNEGGER, J.; WELLS, W. M.; FRANGI, A. F. (Ed.). *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Cham: Springer International Publishing, 2015. p. 234–241. ISBN 978-3-319-24574-4.

ROSSUM, G. V.; DRAKE, F. L. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009. ISBN 1441412697.

RUSSELL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach*. 3. ed. [S.l.]: Prentice Hall, 2010.

TRAN, P. A fully convolutional neural network for cardiac segmentation in short-axis mri. 04 2016.

WALT, S. van der; SCHÖNBERGER, J. L.; Nunez-Iglesias, J.; BOULOGNE, F.; WARNER, J. D.; YAGER, N.; GOUILLART, E.; YU, T.; CONTRIBUTORS the scikit-image. scikit-image: image processing in Python. *PeerJ*, v. 2, p. e453, 6 2014. ISSN 2167-8359. Disponível em: <<https://doi.org/10.7717/peerj.453>>.

YAKUBOVSKIY, P. *Segmentation Models*. GitHub, 2019. Último acesso em 17 de setembro de 2021. Disponível em: <https://github.com/qubvel/segmentation_models>.

YANG, X.; BIAN, C.; YU, L.; NI, D.; HENG, P.-A. Class-balanced deep neural network for automatic ventricular structure segmentation. In: _____. [S.l.: s.n.], 2018. p. 152–160. ISBN 978-3-319-75540-3.

YE, C.; WANG, W.; ZHANG, S.; WANG, K. Multi-depth fusion network for whole-heart ct image segmentation. *IEEE Access*, v. 7, p. 23421–23429, 2019.

ZHOU, Z.; SIDDIQUEE, M. M. R.; TAJBAKHS, N.; LIANG, J. UNet++: A Nested U-Net Architecture for Medical Image Segmentation. In: *DEEP LEARNING IN MEDICAL IMAGE ANALYSIS AND MULTIMODAL LEARNING FOR CLINICAL DECISION SUPPORT, DLMI 2018*. [S.l.: s.n.], 2018. (Lecture Notes in Computer Science, 11045), p. 3–11.

ZREIK, M.; LEINER, T.; VOS, B. D.; HAMERSVELT, R. van; VIERGEVER, M.; ISGUM, I. Automatic segmentation of the left ventricle in cardiac ct angiography using convolutional neural networks. In: . [S.l.: s.n.], 2016. p. 40–43.