

**FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO**

# **Carotid Lumen Segmentation using a Neural Network Approach**

**Alexandre Saraiva Moreira**

VERSÃO DE TRABALHO



Mestrado Integrado em Engenharia Informática e Computação

Orientadora: Catarina de Brito Carvalho

Co-orientador: Jorge Alves da Silva

3 de Março de 2019



# **Carotid Lumen Segmentation using a Neural Network Approach**

**Alexandre Saraiva Moreira**

Mestrado Integrado em Engenharia Informática e Computação





# Resumo

Aterosclerose, uma das principais causas de morte e deficiência permanente em países desenvolvidos, refere-se à acumulação progressiva de placa no interior das paredes das artérias carótidas ao longo de várias décadas e à consequente diminuição do diâmetro do lúmen. Eventualmente, o endurecimento e estreitamento da artéria, também conhecido por estenose, irá colocar o fluxo do sangue em risco. Devido ao longo período de tempo necessário para que as artérias estreitem até níveis perigosos, a estenose das artérias carótidas, normalmente, não causa sintomas nos pacientes até se tornar realmente severo, o que faz com que torne difícil uma detecção antecipada e o respetivo diagnóstico [NU].

A forma principal de diagnosticar esta doença, ainda maioritariamente manual, passa pelo uso de imagens de ultra-som, às quais são aplicadas técnicas de segmentação de imagens tradicionais [RSC14], [RAC16], sendo assim possível a medição do diâmetro do lúmen. Contudo, apesar dos avanços feitos nos últimos anos em sistemas de reconhecimento visual [LKB<sup>+</sup>17] e dos trabalhos desenvolvidos na área [TSG<sup>+</sup>16], [LGB<sup>+</sup>17], técnicas de Redes Neurais ainda não foram exploradas, neste contexto, da mesma forma que as técnicas de segmentação de imagens tradicionais.

Como tal, esta tese propõe e descreve o desenvolvimento de uma técnica completamente automática para a segmentação do lúmen da artéria carótida em imagens de ultra-som modo-B longitudinais. O método desenvolvido baseia-se na *U-Net*, uma Rede Neuronal Convolutiva desenvolvida especificamente para resolver problemas de detecção e segmentação em imagens biomédicas. Os resultados obtidos permitem concluir que a solução proposta é capaz de lidar com o ruído, artefactos e variações de contraste características de imagens ultra-som e com outras estruturas semelhantes à artéria carótida presentes na imagem. Adicionalmente, foi feita uma comparação com uma técnica semi-automática [RCD<sup>+</sup>18], de forma a demonstrar que o método desenvolvido é capaz igualar e ultrapassar os métodos até agora desenvolvidos para a resolução do mesmo problema. Em 40 imagens de teste obteve-se uma pontuação DICE média de 97.57%.



# Abstract

Atherosclerosis, one of the main causes of death and permanent disability in developed countries, refers to the slow build up of plaque on the carotid artery walls over the decades and the consequent decrease of the lumen's diameter. Eventually, the hardening and narrowing of the artery, known as stenosis, will put the blood flow at risk. Due to the long period of time it takes for the artery to narrow down to dangerous levels, carotid artery stenosis does not, usually, present symptoms until it becomes severe, making it difficult for an early detection and diagnosis [NU].

The main method of diagnosis for this disease, which is still primarily manual, is the use of ultrasound images, through which the quantification of the lumen's diameter is commonly done using traditional image segmentation techniques [RSC14], [RAC16]. However, despite their breakthrough in visual recognition systems [LKB<sup>+</sup>17] and the developed work related to the subject [TSG<sup>+</sup>16], [LGB<sup>+</sup>17], Neural Network techniques have not yet been extensively explored in their application the same way traditional image segmentation techniques have.

As such, this thesis proposes and describes the development of a completely automatic technique with the goal of segmenting the carotid's lumen in longitudinal B-mode ultrasound images. The developed method is based on the U-Net, a Neural Network created, specifically, to solve detection and segmentation problems in biomedical images. The results lead to the conclusion that the proposed solution is able to deal with noise, artifacts and contrast variations typical of ultrasound images and with the presence of carotid-like structures. Additionally, the results were compared with a semi-automatic technique [RCD<sup>+</sup>18] to show that the developed method equals and surpasses the methods so far developed with the purpose of solving the same problem. The developed method obtained an average DICE score of 97.57% in a test set of 40 images.



# Agradecimentos

Gostava de começar por agradecer a todos os autores, quer de artigos, tutoriais, livros, aulas ou documentários que partilharam os seus trabalhos ou as suas interpretações dos trabalhos de outros. Tais documentos foram de grande ajuda para o desenvolvimento contínuo deste projeto.

Naturalmente não poderia deixar de agradecer a Catarina Carvalho e Jorge Silva, respetivamente, orientadora e co-orientador desta tese cuja ajuda e orientação contínua ao longo deste ano, permitiu-me alcançar o estado atual do projeto. Tal agradecimento estende-se, também, a todo o grupo do Centro de Investigação em Engenharia Biomédica (C-BER).

Para acabar, gostava de agradecer à minha família e amigos que me suportaram ao longo destes anos e que, de uma forma ou outra, me ajudaram a alcançar o fim desta etapa da minha vida. Agradecimentos especiais à minha irmã, Cristina Moreira, e aos meus colegas de curso, e grandes amigos, João Isaías, Pedro Carvalho, Pedro Pereira e Pedro Romano Barbosa.

Alexandre Saraiva Moreira



*“Never trust a computer you can’t throw out a window.”*

Steve Wozniak





# Conteúdo

|          |                                                                    |           |
|----------|--------------------------------------------------------------------|-----------|
| <b>1</b> | <b>Introdução</b>                                                  | <b>1</b>  |
| 1.1      | Contexto . . . . .                                                 | 1         |
| 1.2      | Motivação e Objetivos . . . . .                                    | 2         |
| 1.3      | Estrutura do Documento . . . . .                                   | 3         |
| <b>2</b> | <b>Segmentação em Imagens de Ultra-som Modo-B</b>                  | <b>5</b>  |
| 2.1      | Imagens de Ultra-som Modo-B . . . . .                              | 5         |
| 2.1.1    | Obtenção das Imagens . . . . .                                     | 5         |
| 2.1.2    | Características das Imagens . . . . .                              | 6         |
| 2.2      | Técnicas de Segmentação . . . . .                                  | 8         |
| 2.2.1    | Modelo de Contornos Ativos . . . . .                               | 9         |
| 2.2.2    | <i>Shape Priors</i> . . . . .                                      | 11        |
| 2.2.3    | <i>Super-pixel</i> ou <i>Patch-based</i> e Classificação . . . . . | 12        |
| 2.2.4    | Textura e Classificação . . . . .                                  | 13        |
| 2.2.5    | Programação Dinâmica . . . . .                                     | 14        |
| 2.2.6    | Redes Neurais Convolucionais . . . . .                             | 17        |
| 2.3      | Conclusão . . . . .                                                | 21        |
| <b>3</b> | <b>Redes Neurais Convolucionais</b>                                | <b>25</b> |
| 3.1      | Redes Neurais . . . . .                                            | 25        |
| 3.1.1    | Arquitetura de uma Rede Neuronal . . . . .                         | 25        |
| 3.1.2    | Neurónio . . . . .                                                 | 26        |
| 3.1.3    | Processo de Aprendizagem . . . . .                                 | 28        |
| 3.2      | Dados de Treino, Validação e Teste . . . . .                       | 30        |
| 3.3      | <i>Overfitting</i> , <i>Underfitting</i> e Capacidade . . . . .    | 31        |
| 3.4      | Tarefas . . . . .                                                  | 32        |
| 3.5      | Camadas . . . . .                                                  | 33        |
| 3.5.1    | Convolução . . . . .                                               | 34        |
| 3.5.2    | <i>Fully Connected</i> . . . . .                                   | 35        |
| 3.5.3    | <i>Pooling</i> . . . . .                                           | 36        |
| 3.5.4    | <i>Batch Normalization</i> . . . . .                               | 37        |
| 3.5.5    | <i>Dropout</i> . . . . .                                           | 37        |
| 3.6      | Arquiteturas . . . . .                                             | 37        |
| 3.6.1    | Classificação . . . . .                                            | 37        |
| 3.6.2    | Deteção . . . . .                                                  | 40        |
| 3.6.3    | Segmentação Semântica . . . . .                                    | 41        |

## CONTEÚDO

|          |                                                             |            |
|----------|-------------------------------------------------------------|------------|
| <b>4</b> | <b>Implementação da Rede Neuronal</b>                       | <b>43</b>  |
| 4.1      | Equipamento e Ferramentas . . . . .                         | 43         |
| 4.2      | Conjunto de Imagens . . . . .                               | 43         |
| 4.2.1    | Preparação das Imagens . . . . .                            | 45         |
| 4.2.2    | Seleção da Região de Interesse . . . . .                    | 46         |
| 4.2.3    | Divisão em Conjuntos de Treino, Validação e Teste . . . . . | 46         |
| 4.3      | Implementação da Solução Proposta . . . . .                 | 46         |
| 4.3.1    | Arquitetura da Rede . . . . .                               | 47         |
| 4.3.2    | Redimensionamento das Imagens . . . . .                     | 49         |
| 4.3.3    | Critério de Avaliação . . . . .                             | 50         |
| 4.3.4    | Otimizador . . . . .                                        | 51         |
| 4.4      | Configurações Experimentais . . . . .                       | 51         |
| 4.5      | Configuração Final . . . . .                                | 57         |
| <b>5</b> | <b>Resultados e Discussão</b>                               | <b>59</b>  |
| 5.1      | Resultados da Solução Implementada . . . . .                | 59         |
| 5.1.1    | Comparação Inter-Observador . . . . .                       | 60         |
| 5.1.2    | Comparação com Método Semi-Automático . . . . .             | 62         |
| <b>6</b> | <b>Conclusões e Trabalho Futuro</b>                         | <b>67</b>  |
| 6.1      | Satisfação dos Objetivos . . . . .                          | 67         |
| 6.2      | Trabalho Futuro . . . . .                                   | 67         |
|          | <b>Referências</b>                                          | <b>69</b>  |
| <b>A</b> | <b>Configurações de Treino</b>                              | <b>75</b>  |
| <b>B</b> | <b>Imagens de Teste</b>                                     | <b>101</b> |

# Lista de Figuras

|      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |    |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 1.1  | Imagens de ultra-som modo-B da artéria carótida comum (a) e (b) e respectivas delimitações do <i>ground truth</i> (c) e (d) . . . . .                                                                                                                                                                                                                                                                                                                                           | 3  |
| 2.1  | Ilustração de como uma imagem ultra-som modo-B é obtida. (A) Ondas ultrasónicas são emitidas para dentro do tecido e os ecos são refletidos de volta para o transdutor; (B) O equipamento determina a localização vertical e horizontal e o contraste de um pixel através da localização no transdutor onde o eco retornou, quanto tempo levou a retornar e a sua amplitude; (C) Estes passos repetem-se até a imagem estar completamente formada. Retirado de [WS11] . . . . . | 6  |
| 2.2  | Exemplos de artefactos de imagens de ultra-som modo-B. (a) Sombra acústica, (b) imagem espelhada, (c) reverberação (causada por um pedaço de vidro) e (d) reforço acústico. Retirado de [BB] . . . . .                                                                                                                                                                                                                                                                          | 7  |
| 2.3  | Exemplos de imagens de ultra-som modo-B da (a) artéria carótida com placa, (b) da bifurcação da artéria carótida e (c) da artéria carótida com a veia jugular interna. Retirado de [RL17] . . . . .                                                                                                                                                                                                                                                                             | 8  |
| 2.4  | Exemplo de um Modelo de Contornos Ativos. Retirado de [DDB11] . . . . .                                                                                                                                                                                                                                                                                                                                                                                                         | 10 |
| 2.5  | Fases de processamento até à segmentação final (pré-processamento, iterações finais da segmentação e segmentação final, por ordem, da esquerda para a direita). Retirado de [BSC <sup>+</sup> 17] . . . . .                                                                                                                                                                                                                                                                     | 11 |
| 2.6  | Exemplos de resultados obtidos ao longo de 6 frames. As linhas vermelhas representam a segmentação obtida e as verdes a segmentação manual. Retirado de [NHPY16] . . . . .                                                                                                                                                                                                                                                                                                      | 12 |
| 2.7  | Exemplo de resultados obtidos onde se vê são as imagens (a) originais, (b) após a redução do <i>speckle</i> , (c) após a divisão em fragmentos, (d) após os mapas de saliência, (e) seleção da região fonte e (f) segmentação final. Retirado de [WLJ <sup>+</sup> 15] . . . . .                                                                                                                                                                                                | 13 |
| 2.8  | Estrutura típica de um método de segmentação baseado em texturas. Retirado de [BP13] . . . . .                                                                                                                                                                                                                                                                                                                                                                                  | 14 |
| 2.9  | Estrutura do método proposto. <i>LDA</i> = <i>Linear Discriminant Analysis</i> , <i>ARD</i> = <i>Average Radial Derivative</i> , <i>BUS</i> = <i>Breast Ultrasound</i> . Retirado de [GFRO16] . . . . .                                                                                                                                                                                                                                                                         | 15 |
| 2.10 | (a) Filtro Gaussiano, (b) máscara binária, (c) <i>Laplacian of Gaussian</i> , (d) modelo sombra, (e) energia local, (f) fase local, (g) simetria de características, (h) <i>backscattering</i> integrado e (i) mapeamento da probabilidade de estrutura óssea. Retirado de [JMH <sup>+</sup> 16] . . . . .                                                                                                                                                                      | 16 |
| 2.11 | Exemplos de segmentações do (a) interior e (b) exterior das paredes da Artéria Carótida Comum. Retirado de [RSC11] . . . . .                                                                                                                                                                                                                                                                                                                                                    | 17 |
| 2.12 | Estrutura final da Rede Neuronal Convolucional implementada. Retirado de [SL16] . . . . .                                                                                                                                                                                                                                                                                                                                                                                       | 18 |
| 2.13 | Resultados obtidos em duas imagens de ultra-som de órgãos diferentes. Retirado de [SL16] . . . . .                                                                                                                                                                                                                                                                                                                                                                              | 18 |

## LISTA DE FIGURAS

|      |                                                                                                                                                                               |    |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 2.14 | Representação da arquitetura <i>FCN</i> proposta. Retirado de [SBIN17]                                                                                                        | 19 |
| 2.15 | Exemplos de resultados obtidos com as diferentes variações da arquitetura proposta. Retirado de [SBIN17]                                                                      | 19 |
| 2.16 | Representação da arquitetura <i>U-Net</i> . Retirado de [RFB15]                                                                                                               | 20 |
| 2.17 | Resultado exemplo obtido através da aplicação duma arquitetura <i>U-Net</i> para a segmentação de nervos em imagens de ultra-som do pescoço. Retirado de [Li16]               | 20 |
| 2.18 | Representação da arquitetura <i>VGG16-UNet</i> . Retirado de [BDS18]                                                                                                          | 21 |
| 2.19 | Exemplos de resultados obtidos a partir da <i>U-Net</i> base, a modificada e com Aumento de Dados. Retirado de [BDS18]                                                        | 22 |
| 3.1  | Ilustração de uma Rede <i>Feedforward</i> com uma camada de entrada, duas ocultas e uma de saída onde $X_i$ e $Y_i$ representam, respetivamente, as entradas e saídas da rede | 26 |
| 3.2  | Exemplo de um neurónio artificial. Retirado de [Wan17]                                                                                                                        | 27 |
| 3.3  | Representações gráficas das funções (a) <i>Sigmoid</i> , (b) Tangente Hiperbólica e (c) <i>ReLU</i>                                                                           | 28 |
| 3.4  | Exemplos de <i>Underfitting</i> , ótimo e <i>Overfitting</i> (da esquerda para a direita). Retirado de [Bha18]                                                                | 32 |
| 3.5  | Exemplos de tarefas de (a) classificação, (b) localização, (c) deteção e (d) segmentação. Retirados de [Des16]                                                                | 33 |
| 3.6  | Exemplo da arquitetura convencional de uma Rede Neuronal Convolutiva. Retirado de [Der17]                                                                                     | 34 |
| 3.7  | Exemplo de uma Convolução sem <i>padding</i>                                                                                                                                  | 35 |
| 3.8  | Exemplo de uma Convolução com <i>padding</i> de 1                                                                                                                             | 36 |
| 3.9  | Exemplo de <i>Max Pooling</i> com um filtro $2 \times 2$ e <i>stride</i> de 2                                                                                                 | 37 |
| 3.10 | Exemplo de uma rede convencional (a) sem e (b) com <i>dropout</i> . Retirados de [dS]                                                                                         | 38 |
| 3.11 | Estrutura da arquitetura <i>AlexNet</i> . Retirado de [KSH12]                                                                                                                 | 38 |
| 3.12 | As diferentes configurações de <i>VggNet</i> . Retirado de [SZ14]                                                                                                             | 39 |
| 3.13 | Estrutura da arquitetura <i>GoogLeNet</i> . Cada módulo <i>Inception</i> consiste em seis camadas de convoluções e uma de <i>pooling</i> . Retirado de [HCRG <sup>+</sup> 16] | 39 |
| 3.14 | Estrutura da arquitetura <i>ResNet</i> . Retirado de [Das17]                                                                                                                  | 40 |
| 3.15 | Representação de alto nível de <i>R-CNN</i> . Retirado de [GDDM14]                                                                                                            | 40 |
| 3.16 | Representação de alto nível de <i>Fast R-CNN</i> . Retirado de [Gir15]                                                                                                        | 41 |
| 3.17 | Representação de alto nível de <i>Faster R-CNN</i> . Retirado de [RHGS15]                                                                                                     | 41 |
| 3.18 | Estrutura da arquitetura <i>Deconvolution Network</i> . Retirado de [NHH15]                                                                                                   | 42 |
| 3.19 | Exemplo de uma <i>FCN</i> . Retirado de [LSD15]                                                                                                                               | 42 |
| 4.1  | Ilustrações de imagens com (a) ACC, (b) ACC e placa, (c) ACC e VJI e (d) ACC, VJI e placa                                                                                     | 44 |
| 4.2  | Exemplos das segmentações (a) A1, (b) A2 e (c) B1 da mesma imagem e exemplo de uma segmentação da (d) parede longínqua dos conjuntos respetivos                               | 45 |
| 4.3  | Exemplo da (a) imagem original e (b) com a máscara A1 sobreposta. (a) Máscara A1 original e (d) região de interesse selecionada                                               | 47 |
| 4.4  | Representação da arquitetura <i>U-Net</i> . Retirado de [RFB15]                                                                                                               | 48 |
| 4.5  | Arquitetura da rede desenvolvida                                                                                                                                              | 49 |
| 4.6  | Representações gráficas e visuais das interpolações <i>Nearest Neighbor</i> [(a) e (c)] e <i>Cubic Spline</i> [(b) e (d)]. Retirado de [Wik]                                  | 50 |
| 4.7  | Precisão de diferentes otimizadores durante (a) o treino e (b) validação. Retirado de [Sha]                                                                                   | 51 |

## LISTA DE FIGURAS

|      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |    |
|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 4.8  | Pior e melhor imagens do conjunto de teste. (a) Melhor imagem (b) com máscara de referência e (c) com máscara obtida [DICE = 97.15%]. (d) Pior imagem (e) com máscara de referência e (f) máscara obtida [DICE = 5.69%] . . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 52 |
| 4.9  | Representação gráfica da perda em função das épocas nas fases de treino. Não há convergência . . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 53 |
| 4.10 | Pior e melhor imagem do conjunto de teste. (a) Melhor imagem (b) com máscara de referência e (c) com máscara obtida [DICE = 99.43%]. (d) Pior imagem (e) com máscara de referência e (f) máscara obtida [DICE = 93.30%] . . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 54 |
| 4.11 | Representação gráfica da perda em função das épocas nas fases de treino. (a) Com TA constante não converge e (b) com TA reduzida a cada 20 épocas converge . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | 55 |
| 4.12 | Exemplo de uma imagem $8 \times 8$ com um filtro $3 \times 3$ . . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | 56 |
| 4.13 | (a) Imagem original (b) com segmentação $128 \times 128$ e (c) com segmentação $512 \times 512$ . . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 56 |
| 4.14 | Piores imagens com respectivas segmentações das soluções propostas. (a) Com segmentação $384 \times 384$ (solução <b>S2</b> ) e (b) com segmentação $480 \times 480$ (solução <b>S1</b> )                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 58 |
| 5.1  | Gráficos relacionados com o treino, validação e teste da solução implementada .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | 60 |
| 5.2  | (a) Imagem original com artefacto e (b) com segmentação obtida, (c) imagem original com placa e (d) com segmentação obtida, (e) imagem original com VJI e (f) com segmentação obtida (g) imagem original com lúmen ruidoso e (h) segmentação obtida . . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | 61 |
| 5.3  | Imagens originais 5.2 (a), (c), (e) e (g) com segmentações obtidas pelo método [RCD <sup>+</sup> 18] . . . . .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | 65 |
| 5.4  | Piores e melhores imagens do conjunto de teste do método proposto e [RCD <sup>+</sup> 18]. (a) Melhor imagem do método proposto com (b) máscara de referência, (c) máscara obtida pelo método proposto e (d) máscara obtida por [RCD <sup>+</sup> 18]. (e) Melhor imagem do método [RCD <sup>+</sup> 18] com (f) máscara de referência, (g) máscara obtida pelo método proposto e (h) máscara obtida por [RCD <sup>+</sup> 18]. (i) Pior imagem do método proposto com (j) máscara de referência, (k) máscara obtida pelo método proposto e (l) máscara obtida por [RCD <sup>+</sup> 18]. (m) Pior imagem do método [RCD <sup>+</sup> 18] com (n) máscara de referência, (o) máscara obtida pelo método proposto e (p) máscara obtida por [RCD <sup>+</sup> 18]. . . . . | 66 |

## LISTA DE FIGURAS

# Lista de Tabelas

|     |                                                                                                                                                                                                                                                                                                                                                                                                |     |
|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| 4.1 | Exemplos de dimensões de filtros e o <i>padding</i> correspondente. Calculados de acordo com a equação 3.13 definida na secção 3.5.1 . . . . .                                                                                                                                                                                                                                                 | 48  |
| 4.2 | Parâmetros do treino inicial. EP = Épocas, NC = Número de Classes, NN = Número de Níveis da rede, DF = Dimensões dos Filtros, TL = Tamanho dos Lotes, DI = Dimensões das Imagens, FP = Função de Perda, OT = Otimizador, TA = Taxa de Aprendizagem, <i>BCE</i> = <i>Binary Cross-Entropy</i> . A EP e a TA foram definidas como 100 e 0.1 pois estas não estavam definidas no artigo . . . . . | 52  |
| 4.3 | Parâmetros do treino experimental . . . . .                                                                                                                                                                                                                                                                                                                                                    | 53  |
| 4.4 | Parâmetros das melhores configurações das soluções propostas para obter segmentações de maior resolução . . . . .                                                                                                                                                                                                                                                                              | 57  |
| 4.5 | Parâmetros da configuração final selecionada . . . . .                                                                                                                                                                                                                                                                                                                                         | 58  |
| 5.1 | Resumo dos resultados obtidos com o método desenvolvido . . . . .                                                                                                                                                                                                                                                                                                                              | 59  |
| 5.2 | Pontuações DICE das 15 imagens originalmente do conjunto de 50. As primeiras 3 colunas apresentam as comparações inter-observador. As últimas 3 colunas apresentam as comparações das máscaras obtidas pelo método desenvolvido com as segmentações manuais A1, A2 e B1. SP = Solução Proposta . . . . .                                                                                       | 62  |
| 5.3 | Pontuações DICE das 15 imagens originalmente do conjunto de 50. As primeiras 3 colunas apresentam as comparações inter-observador. As últimas 3 colunas apresentam as comparações das máscaras obtidas pelo método [RCD <sup>+</sup> 18] com as segmentações manuais A1, A2 e B1. SA = Método Semi-Automático [RCD <sup>+</sup> 18] . . . . .                                                  | 63  |
| 5.4 | Pontuações DICE das 40 imagens do conjunto de teste. <i>GT</i> = <i>Ground Truth</i> . . . . .                                                                                                                                                                                                                                                                                                 | 64  |
| A.1 | Tabela de todas as configurações de treino executadas . . . . .                                                                                                                                                                                                                                                                                                                                | 100 |

## LISTA DE TABELAS



# Abreviaturas e Símbolos

|             |                                                |
|-------------|------------------------------------------------|
| C-BER       | Centro de Investigação em Engenharia Biomédica |
| ACC         | Artéria Carótida Comum                         |
| VJI         | Veia Jugular Interna                           |
| AVC         | Acidente Vascular Cerebral                     |
| PD          | Programação Dinâmica                           |
| RN          | Rede Neuronal                                  |
| RNC         | Rede Neuronal Convolucional                    |
| <i>FCN</i>  | <i>Fully Convolutional Network</i>             |
| <i>RPN</i>  | <i>Region Proposal Network</i>                 |
| <i>ML</i>   | <i>Machine Learning</i>                        |
| <i>FC</i>   | <i>Fully Connected</i>                         |
| <i>MLP</i>  | <i>Multi Layer Perceptron</i>                  |
| <i>MP</i>   | <i>Max Pooling</i>                             |
| <i>MuP</i>  | <i>Max UnPooling</i>                           |
| <i>ReLU</i> | <i>Rectifier Linear Unit</i>                   |
| <i>SVM</i>  | <i>Support Vector Machine</i>                  |
| <i>mAP</i>  | <i>Mean Average Precision</i>                  |
| <i>AUC</i>  | <i>Area Under Curve</i>                        |
| <i>GT</i>   | <i>Ground Truth</i>                            |



# Capítulo 1

## Introdução

Neste capítulo aborda-se o contexto em que esta dissertação se insere, o problema que se pretende resolver, a motivação para tal, os objetivos principais do projeto e, por fim, a estrutura geral deste documento. Adicionalmente, esta dissertação foi desenvolvida em associação com o grupo de investigação C-BER.

### 1.1 Contexto

No sistema circulatório humano, as artérias carótidas comuns são responsáveis pelo transporte de sangue oxigenado para a cabeça e pescoço. Qualquer tipo de obstáculo ao fluxo do sangue através destas artérias pode ser fatal. A estenose da carótida refere-se ao endurecimento e afunilamento progressivo do lúmen da artéria, o que faz com que o fluxo do sangue possa vir a ser abrandado, ou até mesmo interrompido, aumentando a probabilidade de provocar doenças potencialmente fatais, tais como, Aterosclerose e Acidentes Vasculares Cerebrais (AVCs) [RL17].

Os AVCs ocorrem quando o transporte de sangue oxigenado para o cérebro é bloqueado, devido à estenose das carótidas (Isquémico), ou quando um vaso no cérebro sofre uma rutura, provoca bloqueios (Hemorrágico), causando danos severos e permanentes ao cérebro, afetando as funções cognitivas [CUFb].

Aterosclerose refere-se à acumulação de placa nas paredes das artérias. A placa é uma substância viscosa, constituída de lípidos, colesterol, cálcio e outros materiais fibrosos que, ao longo do tempo, se acumulam dentro da parede interior das artérias e criam corpos que, quanto maiores, mais reduzem o diâmetro do lúmen, isto é, o diâmetro interior da artéria [CUFa].

Estas são apenas duas das principais causas de morte e deficiências permanentes em países desenvolvidos. Logo, torna-se óbvio porque é que a observação contínua das mudanças que ocorrem no lúmen das carótidas é crucial para o diagnóstico rápido e preciso destas mesmas doenças.

## 1.2 Motivação e Objetivos

Vários tipos de imagens têm sido usados ao longo dos anos para a avaliação precisa e o tratamento de doenças das artérias carótidas. Apesar disso, as imagens de ultra-som tornaram-se e continuam a ser o tipo de imagens principais usadas para tal, devido às suas seguintes vantagens [HMT10]:

- baixo custo;
- portabilidade de equipamento;
- ausência de radiação;
- processo não-invasivo;
- capacidade de obter as imagens em tempo real.

Destacam-se dois tipos de imagens de ultra-som: Modo-B e Doppler. Este projeto foca-se somente nas de Modo-B, visto que estas são as que são usadas mais frequentemente para a visualização e medição da estrutura das carótidas. As imagens Modo-B, ao contrário das Doppler, não dependem do fluxo do sangue, nem da velocidade deste, mas sim, apenas, da reflexão do som [HMT10].

Este projeto foi realizado devido às seguintes razões:

- a segmentação e medição do lúmen das carótidas é, normalmente, um processo manual realizado por médicos experientes, um processo moroso, propenso a erros e cujos resultados variam muito dependendo do observador;
- a quantidade de dados a ser analisada está em constante crescimento e, como tal, a possibilidade de processar grandes bases de dados multi-institucionais torna-se num recurso valioso para projetos e estudos clínicos e académicos.

Assim sendo, a crescente necessidade de métodos automáticos rápidos e precisos de segmentação e medição do lúmen das artérias carótidas torna-se óbvia. Como tal, este projeto tem em vista automatizar o processo de segmentação e medição do lúmen das artérias carótidas, de forma a melhorar a precisão dos resultados e a velocidade com que estes são obtidos.

Nesta dissertação propôs-se e concluiu-se os seguintes objetivos:

- **O1.** Detecção e segmentação do lúmen da artéria carótida em imagens de ultra-som modo-B, através de técnicas de Redes Neurais, mais especificamente, Redes Neurais Convolucionais;
- **O2.** Teste e validação do método desenvolvido;
- **O3.** Comparação dos resultados obtidos com os resultados de uma técnica de segmentação já implementada pelo grupo de investigação C-BER [RCD<sup>+</sup>18].

Logo, tendo as figuras 1.1 (a) e (b) como um exemplo, pretende-se segmentar toda a zona escura que se encontra dentro do limite definido pelas linhas verdes (figuras 1.1 (c) e (d)).

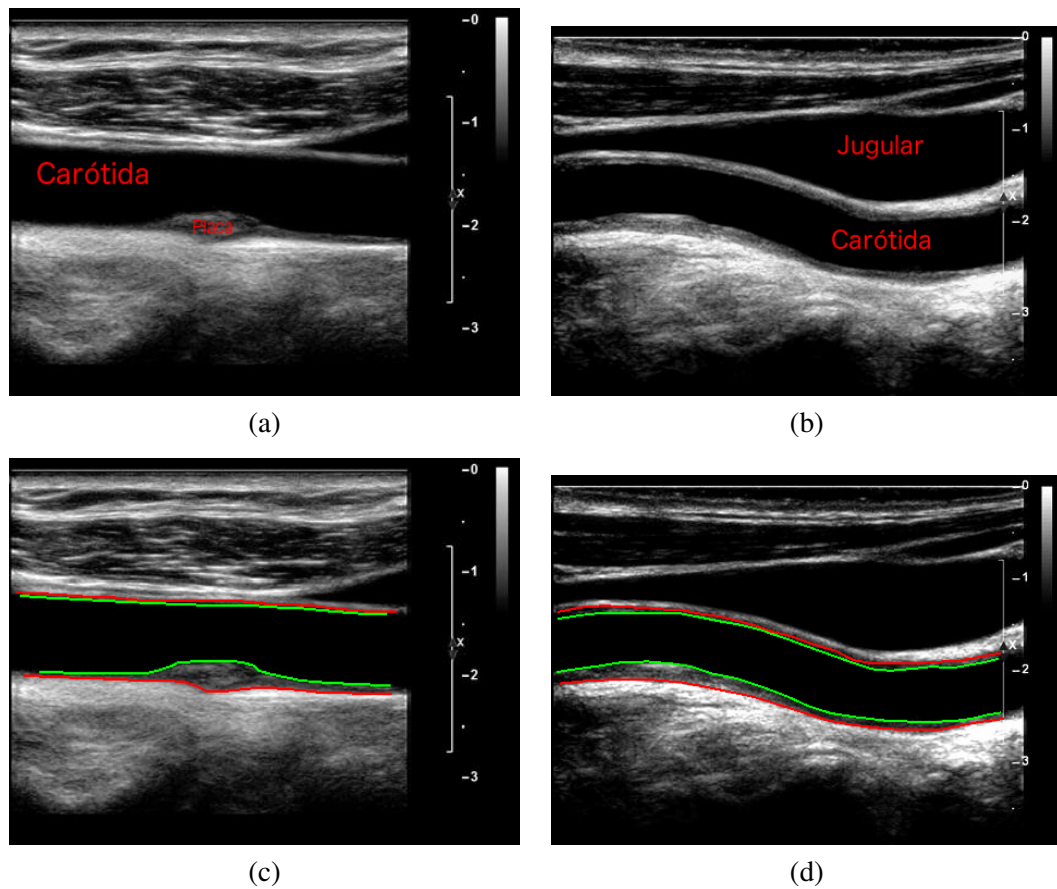


Figura 1.1: Imagens de ultra-som modo-B da artéria carótida comum (a) e (b) e respectivas delimitações do *ground truth* (c) e (d)

### 1.3 Estrutura do Documento

O resto deste documento será estruturado da seguinte forma:

- Capítulo 2, “Segmentação em Imagens de Ultra-som Modo-B”, fornece uma revisão da literatura atualizada sobre as técnicas usadas, em anos recentes, para segmentar órgãos e/ou tecido em imagens de ultra-som modo-B.
- Capítulo 3, “Redes Neurais Convolucionais”, introduz conceitos básicos sobre Redes Neurais Convolucionais.
- Capítulo 4, “Implementação da Rede Neuronal”, fornece uma descrição detalhada sobre as várias etapas de desenvolvimento do projeto e sobre a implementação da solução proposta.
- Capítulo 5, “Resultados e Discussão”, apresenta e discute os resultados da solução implementada e compara estes com as segmentações de médicos experientes e com os resultados de um método desenvolvido pelo grupo de investigação C-BER.

## Introdução

- Capítulo 6, “Conclusões e Trabalho Futuro”, fornece uma conclusão geral sobre a satisfação dos objetivos estipulados e sobre possíveis expansões do trabalho desenvolvido.
- Anexo A, “Configurações de Treino”, fornece uma tabela com todas as configurações de treino da rede executadas.
- Anexo B, “Imagens de Teste”, demonstra todas as imagens do conjunto de teste com as segmentações obtidas a partir do método desenvolvido.

## Capítulo 2

# Segmentação em Imagens de Ultra-som Modo-B

Este capítulo começa com uma introdução breve sobre imagens de ultra-som modo-B , como é que são obtidas, as suas principais características e os desafios que estas colocam para técnicas de segmentação do órgão e/ou tecido de interesse. De seguida, fornece uma revisão da literatura atualizada sobre as técnicas usadas, em anos recentes, para a segmentação em imagens de ultra-som modo-B.

### 2.1 Imagens de Ultra-som Modo-B

As imagens de ultra-som modo-B têm sido usadas cada vez mais frequentemente, em anos recentes, para o diagnóstico de doenças. Sistemas de ultra-som modernos são capazes de visualizar detalhadamente, o fluxo do sangue em vasos e tecidos e/ou estruturas 3D em movimento [WS11].

#### 2.1.1 Obtenção das Imagens

Uma imagem de ultra-som modo-B consiste numa representação de um corte transversal de tecidos e órgãos dentro do corpo. É criada a partir de vários ecos resultantes da reflexão de ondas ultrassónicas nos tecidos e órgãos observáveis. Cada eco é representado num ponto da imagem que corresponde à sua posição inicial do corte transversal do corpo. O contraste da imagem em cada ponto está diretamente relacionado com a amplitude do eco, daí o nome modo-B - *brightness mode* (“modo brilho”).

Para criar uma imagem de ultra-som modo-B, é colocado um transdutor em contacto com a pele que por sua vez transmite, repetidamente, ondas ultrassónicas para dentro do corpo do paciente. Estas são direcionadas ao longo de caminhos em forma de feixes. À medida que estas ondas viajam dentro dos órgãos e tecidos, elas são refletidas, em forma de ecos, alguns dos quais voltam para o transdutor e são, como tal, detetados pelo mesmo. De forma a representar cada eco

numa posição e contraste correspondente ao alvo que o refletiu o sistema necessita de dois dados: a distância do alvo ao transdutor e da posição e orientação do feixe ultrassónico [HMT10]. A figura 2.1 ilustra o processo de obtenção de uma imagem de ultra-som modo-B.

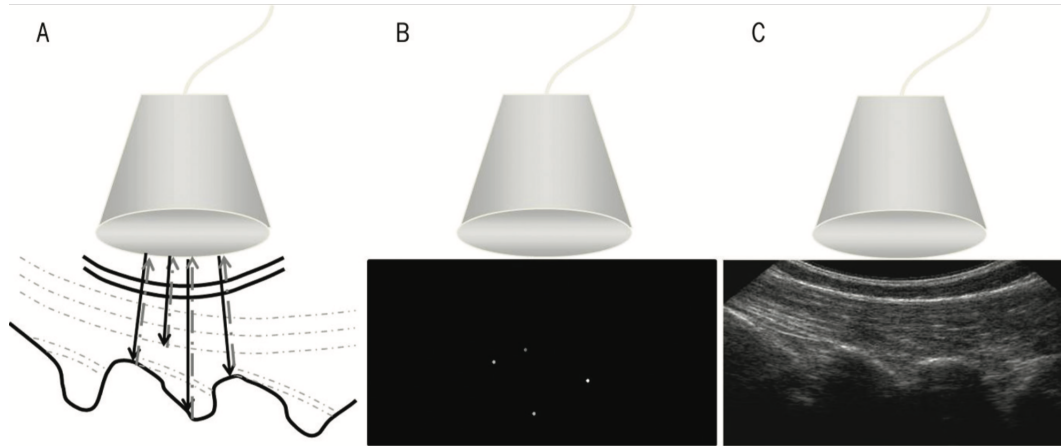


Figura 2.1: Ilustração de como uma imagem ultra-som modo-B é obtida. (A) Ondas ultrassónicas são emitidas para dentro do tecido e os ecos são refletidos de volta para o transdutor; (B) O equipamento determina a localização vertical e horizontal e o contraste de um pixel através da localização no transdutor onde o eco retornou, quanto tempo levou a retornar e a sua amplitude; (C) Estes passos repetem-se até a imagem estar completamente formada. Retirado de [WS11]

Nas artérias, o lúmen, a zona das imagens que se pretende segmentar, refere-se ao tubo interior responsável pelo transporte do sangue. Considerando que o sangue é um fluido e que os fluidos não refletem som, o lúmen é sempre representado por uma zona quase ou completamente escura.

### 2.1.2 Características das Imagens

Apesar do facto de que o equipamento utilizado para obter imagens de ultra-som tem uma aplicação relativamente simples, a qualidade da imagem final dependerá sempre, em parte, do utilizador que a obteve. Como tal, qualquer uso impróprio do equipamento poderá provocar a presença de artefactos de imagem, um dos principais desafios quer para a segmentação manual por parte de médicos experientes, quer para a segmentação automática através de técnicas de computador.

Artefactos de imagem referem-se a qualquer alteração na imagem que não corresponde a uma representação fidedigna da estrutura observada. Os artefactos de imagem podem indicar a presença de estruturas que não existem ou, no pior caso, obstruir a observação de estruturas importantes. Ambos os casos podem levar a erros de diagnóstico. Na figura 2.2 pode-se observar exemplos dos artefactos de imagem mais comuns [LAS13]:

- sombra acústica;
- imagem espelhada;
- reverberações;



## Segmentação em Imagens de Ultra-som Modo-B

- reforço acústico.

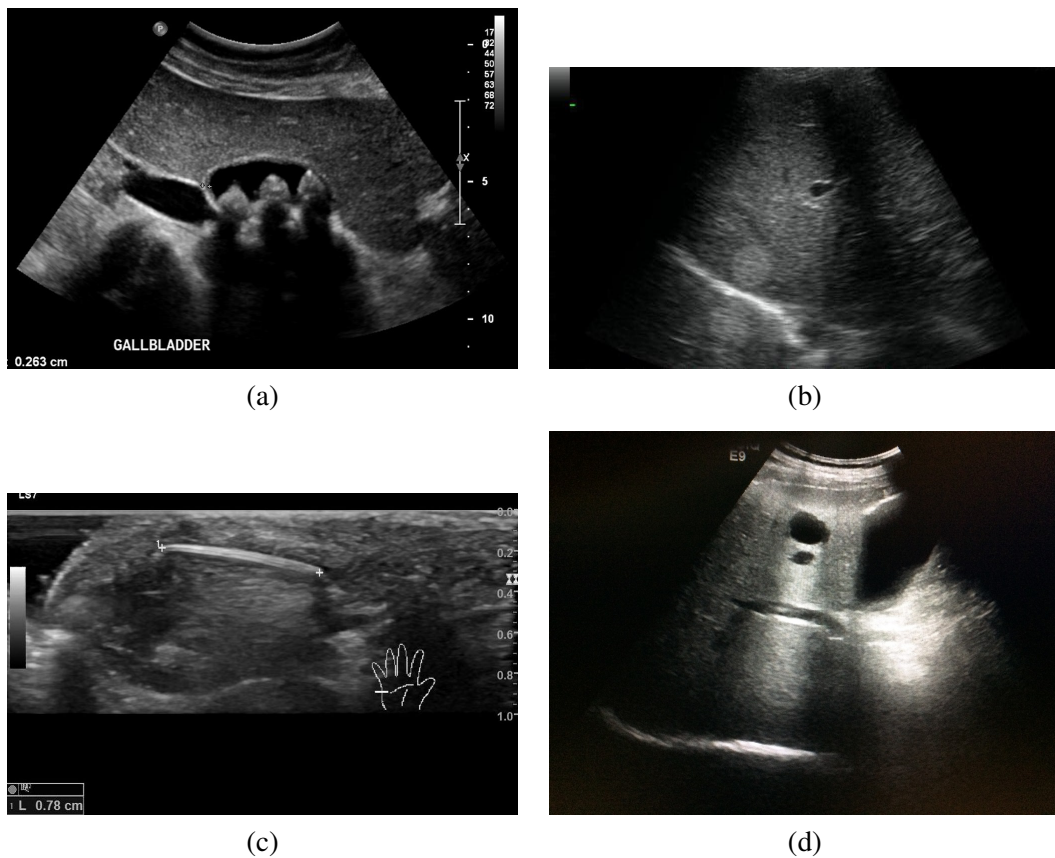


Figura 2.2: Exemplos de artefactos de imagens de ultra-som modo-B. (a) Sombra acústica, (b) imagem espelhada, (c) reverberação (causada por um pedaço de vidro) e (d) reforço acústico. Retirado de [BB]

Uma forma de mitigar a presença de artefactos e a discrepância entre imagens trata-se de usar a mesma configuração do equipamento sempre que possível.

Por vezes, as dificuldades não advêm da obtenção duma imagem imperfeita, mas sim através das estruturas representadas corretamente nas imagens. No caso de imagens longitudinais da carótida, para além da presença da mesma, também é possível, por vezes, observar a veia jugular interna (uma estrutura visualmente semelhante), a bifurcação da carótida e a placa presente no interior das paredes. As figuras 2.3 (a), (b) e (c) demonstram exemplos dos casos especificados.

Para um médico experiente, estas nem sempre apresentam grandes desafios para a segmentação manual, mas o mesmo nem sempre se aplica a técnicas de computador. Qualquer método de processamento de imagem deve ser capaz de lidar de forma consistente e robusta com qualquer uma das estruturas presentes.

Uma característica relevante de todo o tipo de imagens é a sua resolução. A resolução de imagens de ultra-som depende da frequência do transdutor. Ondas de alta frequência geram imagens de maior resolução, no entanto, dependendo da distância do transdutor ao alvo, ondas de

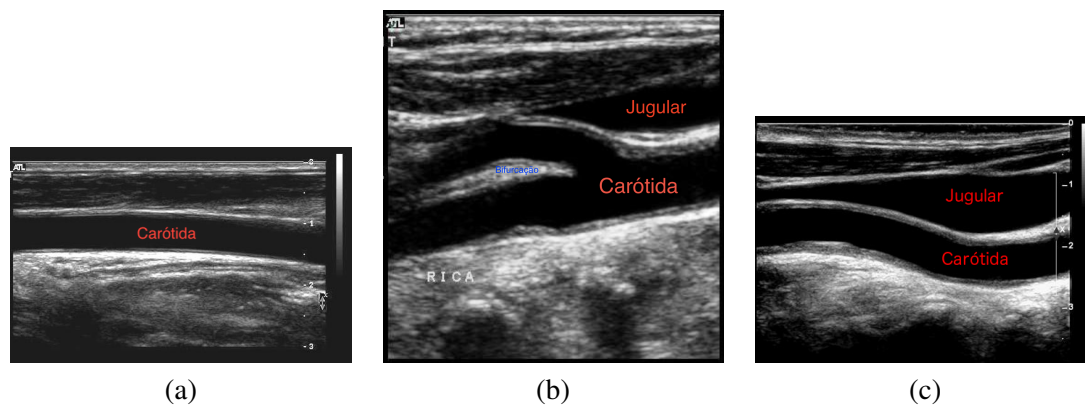


Figura 2.3: Exemplos de imagens de ultra-som modo-B da (a) artéria carótida com placa, (b) da bifurcação da artéria carótida e (c) da artéria carótida com a veia jugular interna. Retirado de [RL17]

baixa frequência são atenuadas menos do que ondas de alta frequência. Ondas de alta (15MHz) e baixa (2 a 5MHz) frequência são ideais para observar, respetivamente, estruturas superficiais e profundas.

Por fim, e mais importante, uma das principais características de imagens de ultra-som trata-se da presença de *speckle* (um efeito que dá uma aparência granular à imagem). Embora *speckle* contenha informação relevante para muitas técnicas que foram desenvolvidas para o seguimento em tempo real de determinadas estruturas (através do seguimento de um padrão específico de *speckle*), este efeito é muitas vezes considerado um artefacto de imagem para técnicas automáticas de localização e segmentação em imagens de ultra-som [MAM17].

## 2.2 Técnicas de Segmentação

Na secção que se segue introduz-se o pré-processamento de imagens, um passo crucial de muitas técnicas de segmentação, e lista-se e descreve-se as seguintes técnicas usadas em anos recentes para a segmentação em imagens de ultra-som modo-B: Modelo de Contornos Ativos, *Shape Priors*, *Super-pixel* ou *Patch-based* e Classificação, Textura e Classificação, Programação Dinâmica e Redes Neurais.

**Pré-processamento das Imagens** Como visto na secção 2.1.2, existem uma série de desafios que têm de ser ultrapassados para se poder localizar e segmentar o tecido ou órgão objetivo com sucesso. Embora nem todos os problemas semelhantes tenham a mesma solução, existem várias técnicas de pré-processamento de imagem que podem mitigar alguns deles e que devem ser usadas sempre que possível.

Por exemplo, de forma a ultrapassar a dependência do utilizador e do equipamento que obteve as imagens, pode-se normalizar a intensidade da imagem fazendo com que o tecido ou órgão de interesse assumam sempre o mesmo valor médio de intensidade de pixel.

*Despeckling* (diminuição do efeito de *speckle* nas imagens), trata-se, também, de uma das técnicas mais importantes de pré-processamento de imagens de ultra-som, devido às razões expostas na secção 2.1.2. No entanto, deve-se ter em conta o tecido ou órgão que se pretende segmentar quando se procura o filtro de *despeckling* mais eficiente para a aplicação em particular [GSP05].

Adicionalmente, quando a segmentação final desejada é demasiado detalhada ou complexa para ser obtida a partir da imagem original completa, pode-se implementar um processo de segmentação de dois-passos. Muitas vezes o primeiro passo consiste na localização da região de interesse, dentro da imagem original, de forma a limitar o processo de segmentação somente à região da imagem que contém o tecido ou órgão em interesse.

É importante ter em conta que algumas das técnicas que se seguem aplicam algum tipo de pré-processamento de imagem aos conjuntos de dados respetivos e que, embora cada uma das técnicas que se segue seja listada e descrita individualmente, uma combinação destas também é por vezes utilizada de forma a desenvolver um algoritmo com um objetivo específico.

### 2.2.1 Modelo de Contornos Ativos

Contornos Ativos trata-se de uma técnica de visão de computador dedicada à delimitação do objeto de interesse numa imagem 2D. Trata-se de uma técnica bastante popular dentro do âmbito correspondente é já foram extensivamente exploradas na sua aplicação em tarefas de seguimento de objetos, reconhecimento de formas, segmentação, deteção de bordas e correspondência estéreo [MAM17].

Contornos Ativos dividem-se em duas sub-classes: paramétricos ou geométricos. Modelos de Contornos Ativos dependem da definição das energias interna e externa e da evolução do contorno inicial até que estas energias atinjam um equilíbrio. Modelos geométricos também podem ser aplicados em imagens 3D. Tanto os modelos paramétricos, como os geométricos, exigem algum tipo de contorno inicial que irá evoluir ao longo do tempo. Ou seja, o contorno inicial, também conhecido como “cobra”, trata-se de uma *spline* deformável que é influenciada pela energia externa, a da estrutura, que a puxa na direção da mesma e pela energia interna, a da “cobra”, que força esta a resistir, até certo ponto, à deformação. A figura 2.4 ilustra um exemplo de um Modelo de Contornos Ativos.

Contornos Ativos paramétricos já foram explorados na sua aplicação em problemas de segmentação das artérias carótidas em imagens de ultra-som [LPP<sup>+</sup>07]. Tendo em conta esta aplicação em específico, uma “cobra” que seja, inicialmente, reta deve ser capaz de se adaptar devidamente às paredes externas ou internas das artérias carótidas. A maior parte dos estudos utilizam a formulação tradicional de uma “cobra” de Williams e Shah [WS92]. Uma “cobra” 2D pode ser descrita como:

$$v(s) = [x(s), y(s)] \quad (2.1)$$

onde  $(x, y)$  representam as coordenadas espaciais da imagem e  $s \in [0, 1]$  denota o domínio paramétrico. De seguida, a evolução dinâmica da “cobra” ocorre segundo a seguinte função global de

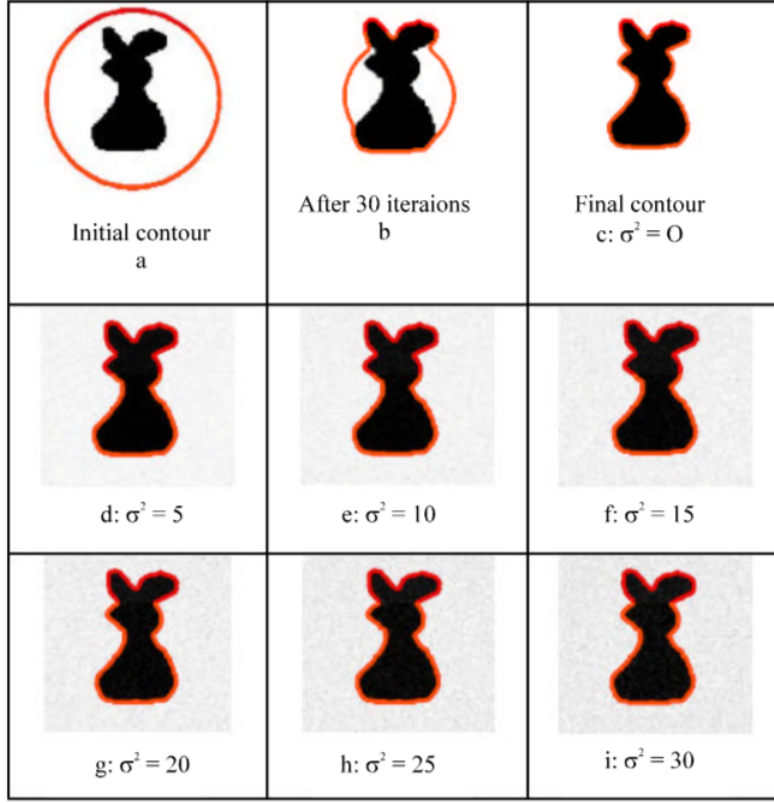


Figura 2.4: Exemplo de um Modelo de Contornos Ativos. Retirado de [DDB11]

energia:

$$E_{snake} = \int_0^1 (E_i(v(s)) + E_e(v(s))) ds \quad (2.2)$$

$$E_i(v(s)) = \int_0^1 \alpha |v'(s)|^2 ds \quad (2.3)$$

$$E_e(v(s)) = \int_0^1 \beta e(v(s)) + \gamma |v(s) - e(v(s))| ds \quad (2.4)$$

onde  $E_i(v(s))$  e  $E_e(v(s))$  representam a energia interna e externa, respetivamente.

Em [BSC<sup>+</sup>17], por exemplo, Contornos Ativos paramétricos foram utilizados para segmentar a placa presente no interior das paredes das artérias carótidas em imagens de ultra-som modo-B. Começaram com uma fase de pré-processamento onde normalizaram as imagens, reduziram o *speckle* através de um filtro gradiente e, por fim, selecionaram a região de interesse. De seguida, e por último, é aplicado um modelo de contornos ativos paramétricos, baseado nas fórmulas tradicionais 2.2, 2.3, 2.4, para a segmentação da placa nas regiões de interesse previamente extraídas. Apenas as curvas fechadas foram consideradas placa e o método foi avaliado em imagens com a presença de uma única placa. A figura 2.5 demonstra o funcionamento do método proposto. Obteve uma precisão de diagnóstico de 82%.



Figura 2.5: Fases de processamento até à segmentação final (pré-processamento, iterações finais da segmentação e segmentação final, por ordem, da esquerda para a direita). Retirado de [BSC<sup>+</sup>17]

### 2.2.2 Shape Priors

Métodos de *Shape Priors* para a segmentação de imagens 2D são baseados no facto de que a forma do objeto desejado já é conhecida, o que lhes oferece uma grande vantagem para a segmentação em imagens de ultra-som. A maior parte dos estudos define um modelo de *shape prior* como [CCR12]:

$$E(C, T) = E_{data}(C, I) + \lambda D[C, T(C_{ref})] \quad (2.5)$$

onde  $I$  representa a imagem de entrada  $I : \Omega \rightarrow \mathbb{R}$ ,  $C$  representa a forma previamente conhecida,  $T$  representa a transformação da forma e  $D$  representa uma métrica de distância da forma.

Este tipo de método é o ideal para lidar com imagens que contenham artefactos, como por exemplo, sombras ou contornos desfocados. Idealmente os modelos são construídos a partir de conjuntos de dados significativos de forma a determinar o *shape prior* final. No entanto, o modelo adquirido só será tão bom quanto o conjunto de dados utilizado para a sua construção. Também se deve ter em consideração que a forma das estruturas anatómicas varia em forma e tamanho, o que dificulta a capacidade da adaptação do modelo à variedade de estruturas que lhe possam ser apresentadas.

Estes métodos podem ser bastante versáteis, dependendo da implementação e aplicação, e podem ser utilizados para a segmentação direta da estrutura de interesse ou como um passo num método de segmentação de dois-passos.

No caso de [NHPY16], foi demonstrada uma melhoria no performance de modelos de Contornos Ativos na segmentação de miomas uterinos em vídeos de ultra-som do útero. As mudanças da forma do objeto de interesse nos frames do vídeo são sucessivas e, como tal, correlacionadas. O método utilizado propõe utilizar esta correlação entre as formas do objeto nos diferentes frames como o conjunto de dados para a construção do modelo *shape prior* que será utilizado para a segmentação final. Ou seja, o modelo *shape prior* é construído com dados específicos ao paciente e, como tal, não só não necessita de dados adicionais, como aumentará a precisão da segmentação. Um algoritmo de Contornos Ativos é aplicado a cada frame do vídeo de forma a delimitar os contornos do objeto que, de seguida, são utilizados para construir o modelo de *shape prior*. Na figura

2.6 pode-se observar os resultados obtidos ao longo de vários frames.

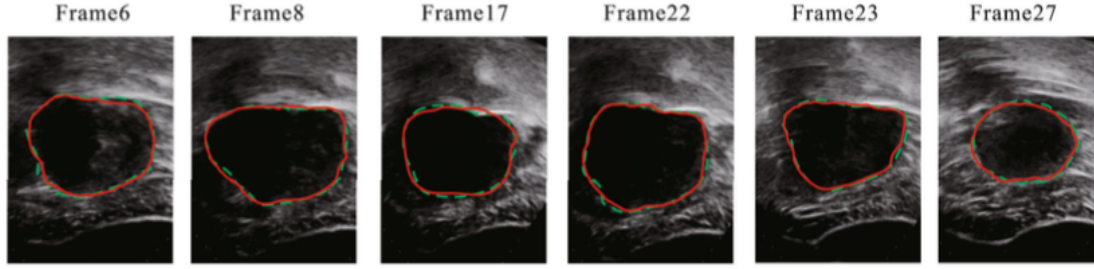


Figura 2.6: Exemplos de resultados obtidos ao longo de 6 frames. As linhas vermelhas representam a segmentação obtida e as verdes a segmentação manual. Retirado de [NHPY16]

### 2.2.3 Super-pixel ou Patch-based e Classificação

Técnicas *Super-pixel* ou *Patch-based* para segmentação de imagens 2D baseiam-se na divisão da imagem original em fragmentos que são, de seguida, classificados como objeto ou *background* através de cortes de grafos. Um corte  $C$  de uma grafo  $G$ , trata-se de uma partição  $V$ , desse mesmo grafo, nos sub-conjuntos  $S$  e  $T$ , onde  $CS(C)$  representa o sub-conjunto do corte  $C$  [Die17].

$$C = (S, T) \quad (2.6)$$

$$G = (V, E) \quad (2.7)$$

$$CS(C) = \{(u, v) \in E | u \in S, v \in T\} \quad (2.8)$$

A divisão da imagem original em fragmentos reduz a complexidade associada com o processamento de milhares de pixels, visto que passa a processar *super-pixels* (fragmentos). Este tipo de métodos normalmente resultam numa segmentação muito grosseira que depois necessita de ser refinada, sendo, muitas vezes, utilizado para determinar a região do tecido ou órgão de interesse em métodos de segmentação de dois-passos.

Um exemplo deste tipo de segmentação pode ser encontrado em [WLJ<sup>+</sup>15]. Neste é proposto um método automático de *region growing* baseado em energia para a segmentação de lesões do fígado em imagens de ultra-som. Após a redução do *speckle*, é gerado um mapa de fragmentos através do algoritmo de *clustering* linear, de forma a separar a imagem original em várias imagens mais pequenas e uniformes (200 neste caso). De seguida, é implementado um método automático de *region growing* baseado em energia para obter a segmentação final. O processo é demonstrado nas figuras 2.7.

Os autores compararam o resultado obtido do método proposto com os resultados de segmentações manuais e os resultados obtidos de outras técnicas de segmentação populares, em 40



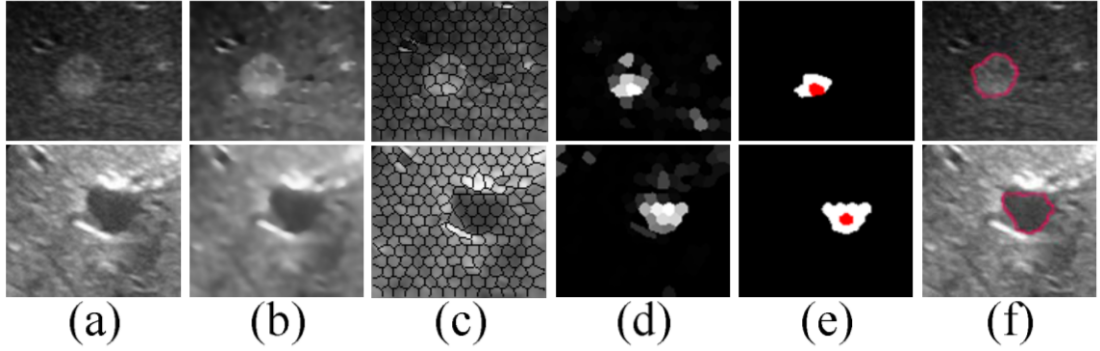


Figura 2.7: Exemplo de resultados obtidos onde se vê são as imagens (a) originais, (b) após a redução do *speckle*, (c) após a divisão em fragmentos, (d) após os mapas de saliência, (e) seleção da região fonte e (f) segmentação final. Retirado de [WLJ<sup>+</sup>15]

imagens de ultra-som do fígado. Observou-se uma melhoria em relação a todas as outras técnicas, com o melhor resultado, na segmentação de quistos hepáticos, de 94.6%.

#### 2.2.4 Textura e Classificação

De acordo com [BP13], uma textura pode ser definida como a repetição regular de um elemento, ou padrão, numa superfície e pode ser utilizada para identificar diferentes regiões da imagem que têm ou não textura. Ou seja, métodos de textura e classificação baseiam-se em encontrar características na estrutura do tecido ou órgão de interesse e, de seguida, classificar os pixels ou secções da imagem, como regiões diferentes, baseando-se nestas características. Tipicamente, métodos baseados em textura seguem a estrutura definida na figura 2.8.

Muitas das técnicas de textura e classificação são baseadas no cálculo das matrizes de ocorrência de Haralick [V. 99], que apesar de serem computacionalmente intensivas, demonstraram-se capazes de devolver bons resultados.

$$\begin{bmatrix} p(1,1) & p(1,2) & p(1,3) & \dots & p(1,N_g) \\ p(2,1) & p(2,2) & p(2,3) & \dots & p(2,N_g) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p(N_g,1) & p(N_g,2) & p(N_g,3) & \dots & p(N_g,N_g) \end{bmatrix} \quad (2.9)$$

onde  $N_g$  representa o número de níveis de cinzento da imagem. Cada elemento  $(i, j)$  da matriz é gerado contando o número de vezes que um pixel de valor  $i$  é encontrado adjacente a um pixel de valor  $j$  seguido da divisão da matriz inteira pelo número de vezes total que tais comparações são feitas. Como tal, cada elemento da matriz trata-se da probabilidade de encontrar um pixel de valor  $i$  adjacente a um pixel de valor  $j$ .

Tal como no método descrito anteriormente, este tipo de métodos normalmente resultam numa segmentação muito grosseira que depois necessita de ser refinada, sendo, também, muitas vezes,

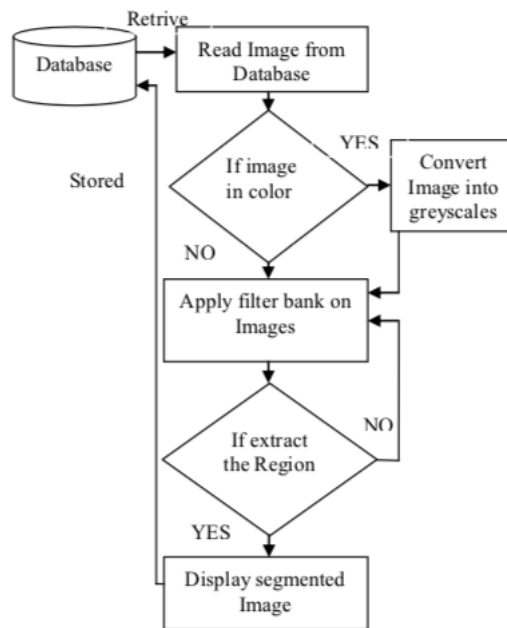


Figura 2.8: Estrutura típica de um método de segmentação baseado em texturas. Retirado de [BP13]

utilizado para determinar a região do tecido ou órgão de interesse em métodos de segmentação de dois-passos.

Em [GFRO16] é proposto um método para a segmentação de lesões em imagens de ultra-som da mama. A imagem de ultra-som original é decomposta num conjunto de canais de texturas, através de filtros *log-Gabor* [FRC09], que por sua vez são divididos em malhas sobrepostas de forma a descrever as texturas locais. De seguida, um classificador de análise discriminante linear associa uma pontuação a cada malha de forma a gerar regiões que, provavelmente, contêm um lesão. Depois, uma função avalia os contornos de potenciais lesões dessas mesmas regiões, de forma a revelar a região com maior probabilidade de conter uma lesão. Por fim, a delimitação é realizada por um processo baseado numa função derivada radial média aplicada à região detetada, detetando assim o contorno final da lesão. A estrutura do método proposto encontra-se definida na figura 2.9. Obteve uma precisão *AUC* média de 92%.

### 2.2.5 Programação Dinâmica

Programação Dinâmica (PD) refere-se aos algoritmos que decompõem um problema complexo em sub-problemas, preferencialmente menos complexos, e armazenam os resultados destes últimos de forma a evitar que estes sejam calculados repetidamente. Ou seja, tratam-se de técnicas usadas para procurar o caminho ótimo para uma determinada solução minimizando ou maximizando uma função custo específica. PD é um método comum de algoritmos de otimização que foi implementado em problemas de visão de computador com sucesso, pela primeira vez, no início do



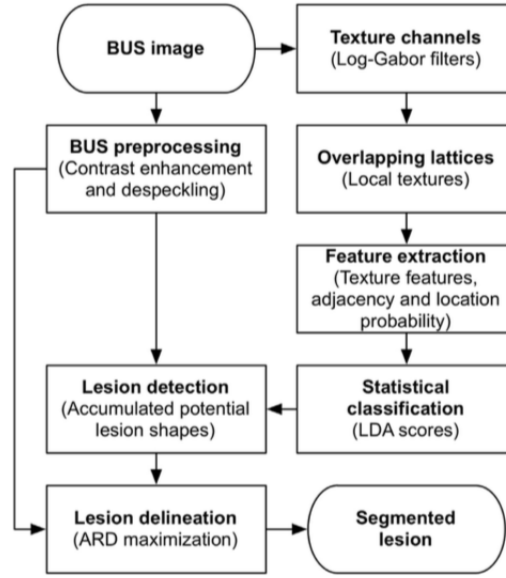


Figura 2.9: Estrutura do método proposto. *LDA* = *Linear Discriminant Analysis*, *ARD* = *Average Radial Derivative*, *BUS* = *Breast Ultrasound*. Retirado de [GFRO16]

anos 90 de forma a minimizar o custo computacional fazendo com que não fosse necessário inspecionar todos os pontos duma imagem. [WLGW97] apresenta uma das primeiras implementações desenvolvidas para a segmentação em imagens de ultra-som.

Tendo em conta o modo de funcionamento destes algoritmos, eles são, normalmente, utilizados como um primeiro ou segundo passo num método de segmentação de dois-passos de forma a refinar o processo de segmentação, por exemplo, a seguir à seleção da região de interesse da imagem de ultra-som original.

No exemplo de [JMH<sup>+</sup>16], é proposto um método automático para a segmentação de ossos em imagens de ultra-som. O método desenvolvido divide-se em dois grandes passos: mapeamento da probabilidade de estrutura óssea e otimização através de PD. O processo de mapeamento encontra-se resumido e exemplificado na figura 2.10.

O segundo passo começa assumindo que existe apenas um pixel do contorno da estrutura óssea em cada coluna da imagem ultra-som. Esses pixels são agrupados em uma das três classes: com osso, sem osso ou com salto. Em cada coluna  $s$  minimiza-se uma função custo composta de energia interna  $E_{int}$  e externa  $E_{ext}$  de forma a localizar a estrutura óssea dentro dessa mesma coluna  $Seg(s)$ .

$$E_{ext} = 1 - BP \quad (2.10)$$

$$E_{int}(i, j) = \begin{cases} \alpha \left\| \frac{dSeg}{ds} \right\|^2 + \beta \left\| \frac{d^2Seg}{ds^2} \right\|^2 + \gamma & \text{Com osso} \\ \text{CustoSalto} & \text{Com salto} \\ \alpha D_1^2 + \beta D_2^2 & \text{Sem osso} \end{cases} \quad (2.11)$$

onde  $BP$  representa a probabilidade de estrutura óssea,  $\alpha$  e  $\beta$  são os parâmetros para a suavização

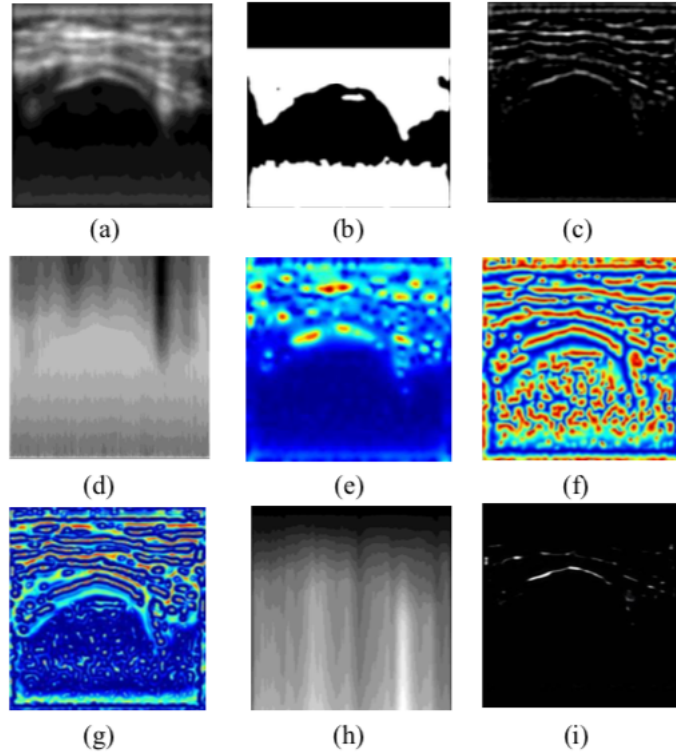


Figura 2.10: (a) Filtro Gaussiano, (b) máscara binária, (c) *Laplacian of Gaussian*, (d) modelo sombra, (e) energia local, (f) fase local, (g) simetria de características, (h) *backscattering* integrado e (i) mapeamento da probabilidade de estrutura óssea. Retirado de [JMH<sup>+</sup>16]

(a primeira derivada de *Seg*) e curvatura (a segunda derivada de *Seg*),  $\gamma$  representa um escalar negativo pequeno e  $s$  representa o índice do pixel atual. O *CustoSalto* penaliza o salto dado entre as regiões com e sem osso. Obtiveram uma precisão de 62.5%.

Em [RSC11], é proposto um método automático de detecção e segmentação das paredes da Artéria Carótida Comum. A abordagem proposta consiste em 4 passos: seleção da região de interesse, estimativa dos mapas *fuzzy edge* e *fuzzy valley edge*, detecção do interior das paredes e detecção do exterior das paredes. A estimativa do mapa *fuzzy edge* é dado por  $f(x, y) = 1 - \tau(x, y)$ , sendo  $\tau(x, y)$  definido por:

$$\tau(x, y) = \begin{cases} \left[ 1 - \left( \frac{ICOV(x, y)}{\sigma_s} \right)^2 \right]^2 & ICOV < \sigma_s \\ 0 & ICOV \geq \sigma_s \end{cases} \quad (2.12)$$

onde  $\sigma_s = \sqrt{5}\sigma_e$  e  $\sigma_e$  representam as escalas dos contornos das imagens (*image edge scale*). Por outro lado, a estimativa do mapa *fuzzy valley edge* é dado por  $f^*(x, y) = 1 - \tau^*(x, y)$ , sendo  $\tau^*(x, y)$  definido por:

$$\tau^*(x, y) = \begin{cases} \left[ 1 - \left( \frac{\Delta I(x, y)}{\sigma_s^*} \right)^2 \right]^2 & \Delta I < \sigma_s^* \\ 0 & \Delta I \geq \sigma_s^* \end{cases} \quad (2.13)$$

onde  $\sigma_s^* = \sqrt{5}\sigma^*$  e  $\sigma^*$  representam as escalas da amplitude normalizada da máxima inferior  $\Delta I = (I_p - I_v)/(I_v + \varepsilon)$  onde  $\varepsilon$  é um valor numérico, perto de zero, para impedir divisões por zero. Tendo a região de interesse e os mapas estimados, estes passam por um algoritmo de PD que deteta as paredes da artéria carótida. Exemplos das segmentações obtidas podem ser observados na figura 2.11.

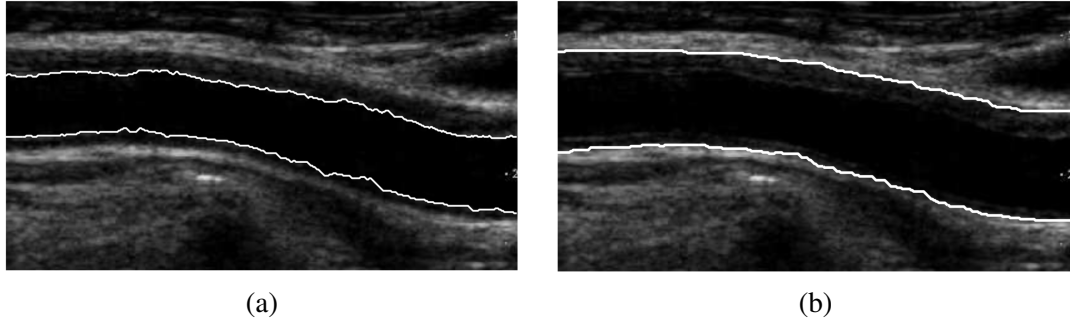


Figura 2.11: Exemplos de segmentações do (a) interior e (b) exterior das paredes da Artéria Carótida Comum. Retirado de [RSC11]

## 2.2.6 Redes Neurais Convolucionais

O Capítulo 3 fornece uma descrição profunda e detalhada da estrutura e funcionamento de Redes Neurais Convolucionais. Vale a pena acrescentar, que estas podem ser aplicadas na segmentação do tecido ou órgão de interesse na imagem de ultra-som inteira ou numa região de interesse previamente extraída. Isto é, podem ser usadas para uma segmentação grosseira ou detalhada de forma direta ou como um segundo passo num método de segmentação de dois-passos.

Em [SL16] é proposto um método automático para deteção e classificação de vasos sanguíneos em imagens de ultra-som em tempo real. A solução proposta divide-se em dois passos principais: localização de regiões candidatas<sup>1</sup> e classificação de cada uma dessas regiões extraídas com uma Rede Neuronal Convolucional, ou seja, o segundo passo determina se as regiões extraídas da imagem original contêm vasos ou não. A arquitetura proposta é, inicialmente, a *AlexNet* que foi gradualmente simplificada através da remoção dos blocos *convolução-pooling* e da redução do número de convoluções (com o objetivo principal de melhorar o tempo de execução).

A arquitetura final encontra-se na figura 2.12 e consiste em: duas convoluções, duas camadas *Max Pooling*, três camadas *Fully Connected* e uma função de ativação *Rectifier Linear Unit (ReLU)* por cada camada convolucional e *Fully Connected*, consistindo, assim, num total de 12 camadas. O método foi testado em imagens transversais da artéria carótida com uma precisão de 96%. Exemplos de segmentação deste método podem ser observados na figura 2.13.

Noutro exemplo, [SBIN17], é proposto um método automático para a segmentação do coração fetal em imagens de ultra-som. A solução proposta aborda o problema como um problema de segmentação semântica com quatro classes de saída: 4C, LVOT, 3V e NH (*background*). A

<sup>1</sup>regiões que podem conter vasos sanguíneos

## Segmentação em Imagens de Ultra-som Modo-B

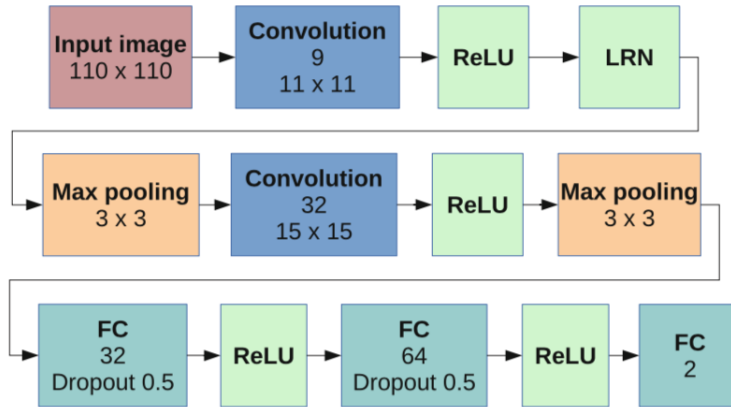


Figura 2.12: Estrutura final da Rede Neuronal Convolutiva implementada. Retirado de [SL16]

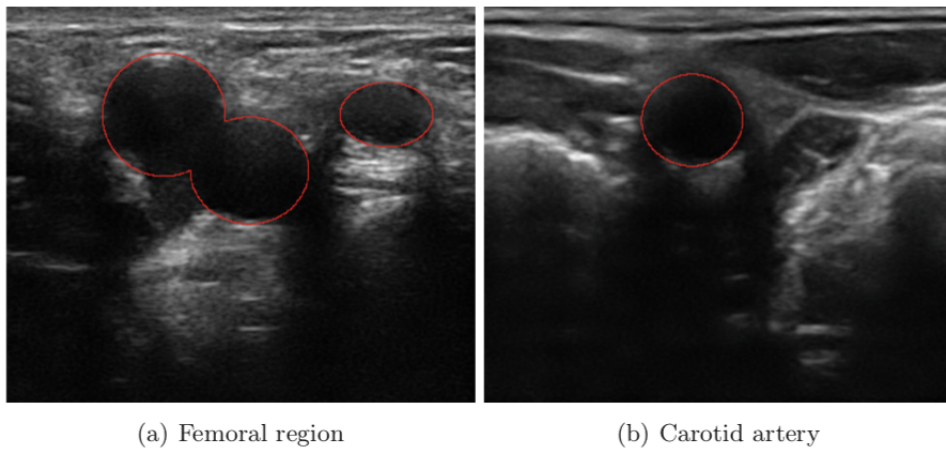


Figura 2.13: Resultados obtidos em duas imagens de ultra-som de órgãos diferentes. Retirado de [SL16]

arquitetura proposta, uma *Fully Convolutional Network*, é baseada na *VggNet* de 16 camadas e encontra-se especificada na figura 2.14.

Todas as imagens são redimensionadas, para  $227 \times 227$ , de forma a serem compatíveis com a arquitetura e as dimensões dos filtros das convoluções. Cada convolução é seguida de uma *ReLU* cuja saída é usada como entrada numa camada de *Max Pooling* com um *stride* de 2. Tendo em conta que o objetivo é a localização e segmentação precisa do coração fetal, apenas as saídas das camadas *Max Pooling* mais profundas são necessárias, no entanto, de forma a comparar o performance de modelos diferentes, as predições após as camadas Pool5, Pool4 e Pool3 passam por um processo de *upsample* para obter os resultados finais das arquiteturas de *stride* 32, 16 e 8, respetivamente, cuja especificação, também, se encontra na figura 2.14. Este método obteve erros de classificação de 26.81%, 23.48% e 68.67% para os métodos *FCN32-stride*, *FCN16-stride* e *FCN8-stride*, respetivamente. Exemplos de resultados encontram-se especificados na figura 2.15.

Adicionalmente, em [RFB15], é proposto um método de segmentação de imagens biomédicas baseado em Redes Neurais Convolucionais, cujo nome advém da sua forma em U (*U-Net*). A ar-

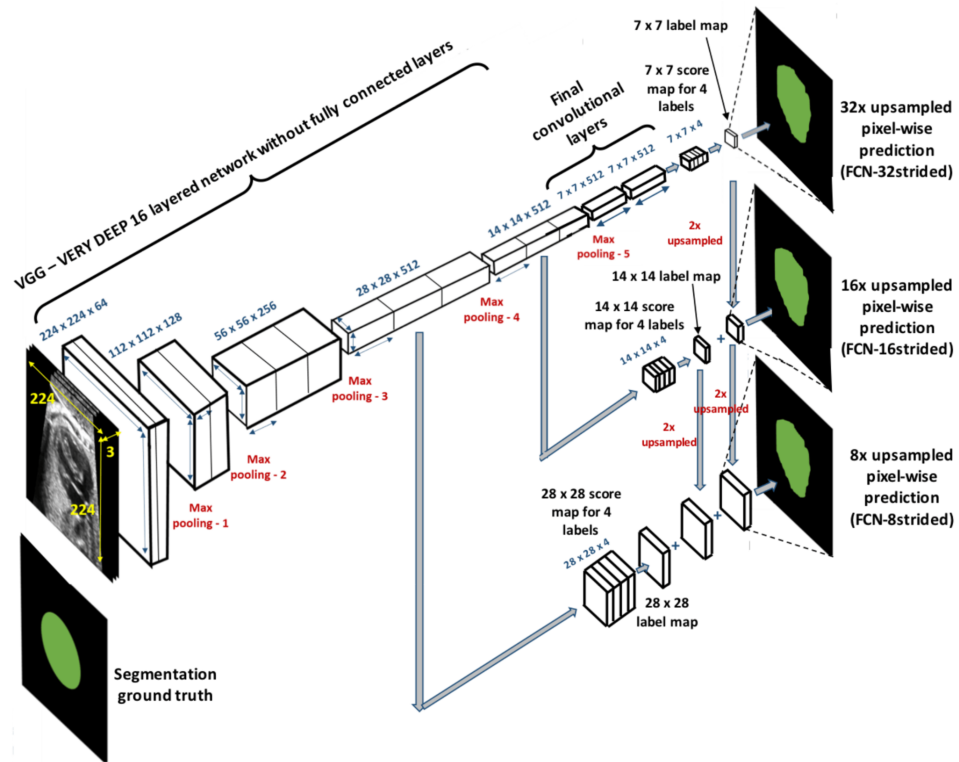


Figura 2.14: Representação da arquitetura *FCN* proposta. Retirado de [SBIN17]

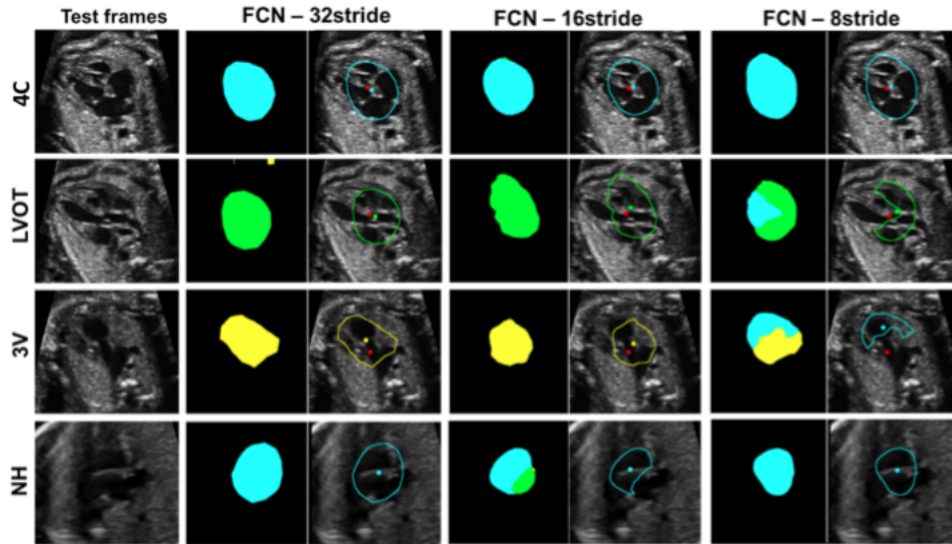


Figura 2.15: Exemplos de resultados obtidos com as diferentes variações da arquitetura proposta. Retirado de [SBIN17]

a arquitetura proposta consiste em dois processos principais: *downsampling* e *upsampling*, isto é, um caminho de “contração” e “expansão”, respectivamente. A sua estrutura encontra-se especificada na figura 2.16.

## Segmentação em Imagens de Ultra-som Modo-B

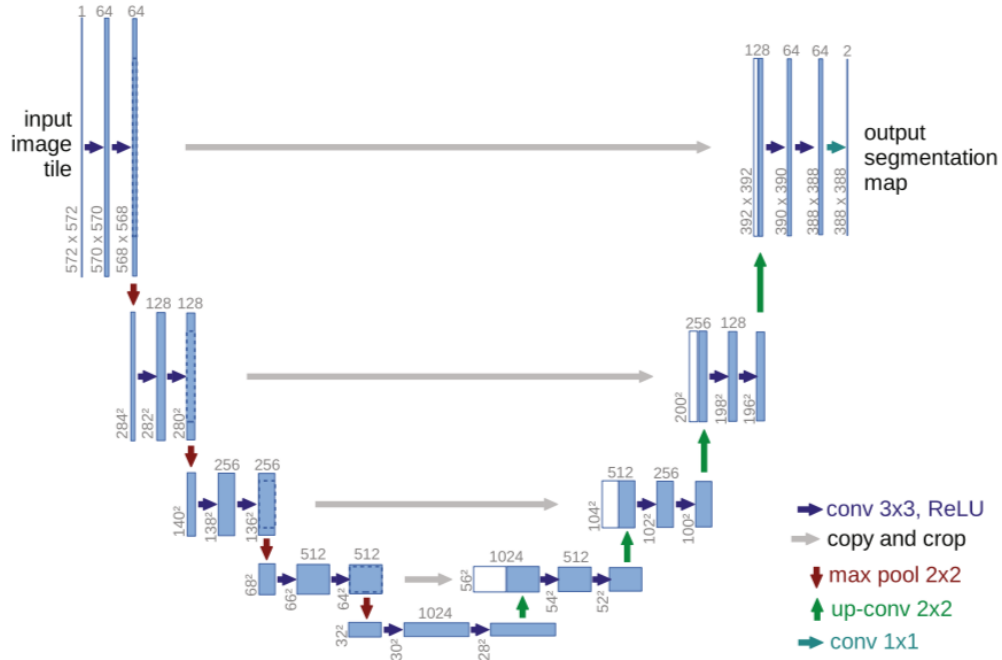


Figura 2.16: Representação da arquitetura *U-Net*. Retirado de [RFB15]

O caminho de “contração” segue uma arquitetura de Rede Neuronal Convolutacional convencional e consiste na aplicação repetida de duas convoluções  $3 \times 3$ , sem *padding*, seguidas por uma *ReLU* e uma camada *Max Pooling*  $2 \times 2$  com um *stride* de 2. Cada passo do caminho de “expansão” consiste num *upsampling* do mapa de ativações, seguido de uma convolução transposta  $2 \times 2$ , uma concatenação com o mapa de ativações correspondente, obtido no caminho anterior, e duas convoluções  $3 \times 3$  seguidas por uma *ReLU*. Na camada final é usado uma convolução  $1 \times 1$  para mapear cada pixel para a sua classe correspondente. No total, a arquitetura consiste de 23 camadas. O exemplo de uma segmentação obtida com uma *U-Net* encontra-se na figura 2.17.

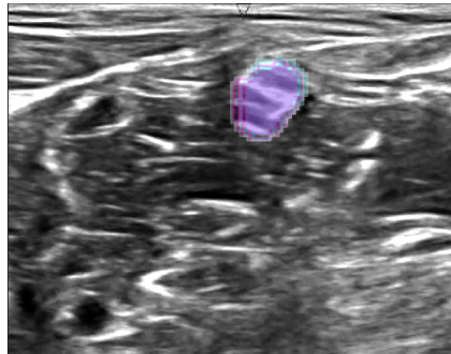


Figura 2.17: Resultado exemplo obtido através da aplicação duma arquitetura *U-Net* para a segmentação de nervos em imagens de ultra-som do pescoço. Retirado de [L16]

Para terminar, em [BDS18], é proposto um método para a segmentação do lúmen em imagens de ultra-som modo-B intra-vasculares. A arquitetura implementada tem como base a *U-Net*, refe-



rida no exemplo anterior, e combina esta com a *VGG-16*, criando assim o que eles designam de *VGG16-UNet*. Ou seja, a *VGG-16* é usada como a parte de “contração” da *U-Net*. A arquitetura proposta encontra-se especificada na figura 2.18.

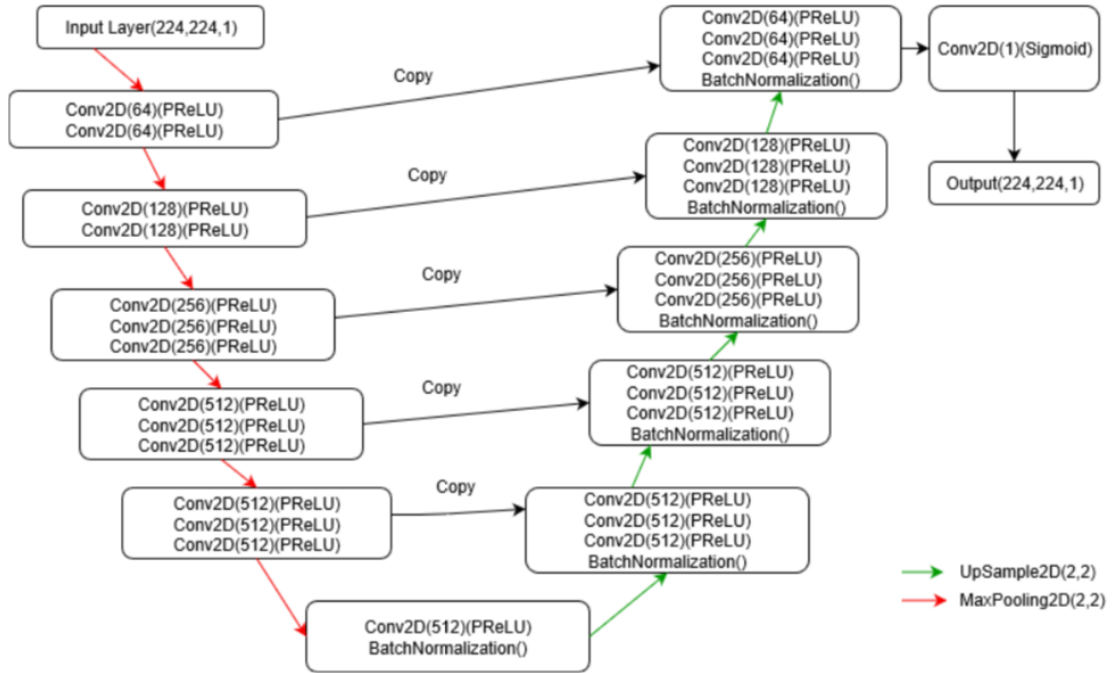


Figura 2.18: Representação da arquitetura *VGG16-UNet*. Retirado de [BDS18]

Apesar do facto de que a *U-Net* é capaz de aprender a partir de relativamente poucos dados, os autores decidiram também aplicar técnicas de Aumento de Dados ao seu conjunto de 109 imagens. Como tal, exemplos de resultados obtidos encontram-se especificados na figura 2.19.

## 2.3 Conclusão

No total foram apresentados e explorados, nas suas implementações e aplicações, seis campos onde técnicas foram, e são, desenvolvidas para a segmentação em imagens de ultra-som modo-B.

Modelos de Contornos Ativos, de certa forma, podem ser considerados técnicas de Programação Dinâmica porque ambos têm como objetivo encontrar um balanço entre as forças de energias internas e externas, no entanto, devido ao número de casos em que estes modelos são aplicados, estas técnicas foram listadas e exploradas separadamente. Também se deve ter em consideração que os Modelos de Contornos Ativos necessitam que seja definido algum contorno inicial (manual ou automaticamente), ao contrário da maioria das técnicas de Programação Dinâmica. Por outro lado, técnicas de Programação Dinâmica adaptam-se com mais facilidade à forma da estrutura de interesse mas dependem muito de uma função custo apropriada. Adicionalmente, estas técnicas

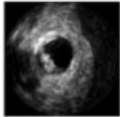
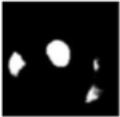
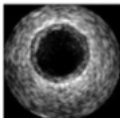
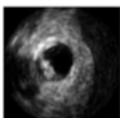
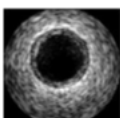
| Test Images        | Original Image                                                                    | Grand Truth                                                                       | Simple U-Net                                                                      | VGG16-UNet without DA                                                              | VGG16-UNet with DA                                                                  |
|--------------------|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| Noisy Lumen Sample |  |  |  |  |  |
| Lumen Sample       |  |  |  |  |  |
| Noisy Media Sample |  |  |  |  |  |
| Media Sample       |  |  |  |  |  |

Figura 2.19: Exemplos de resultados obtidos a partir da *U-Net* base, a modificada e com Aumento de Dados. Retirado de [BDS18]

são testadas e otimizadas com um conjunto de dados específico e, muitas vezes, têm de ser alteradas antes de serem aplicadas a um conjunto diferente. No entanto, uma grande vantagem destas técnicas é que têm em consideração, tanto o contorno, como a forma da estrutura de interesse.

*Shape Priors*, por outro lado, são muito tolerantes à presença de artefactos de imagem. A presença destas características degradantes das imagens, normalmente, não têm grande influência na segmentação final, visto que se procura uma forma bem pré-estabelecida dentro dos limites da imagem. No entanto, são técnicas computacionalmente intensivas, precisam de um conjunto de dados apropriado para determinar a forma a ser detetada e, qualquer tipo de variação da forma desejada, pode levar a que esta não seja detetada corretamente.

Das técnicas restantes, todas necessitam, de alguma forma, que algumas características específicas ou regiões de interesse da imagem sejam classificadas, mais especificamente, as técnicas de *Super-pixel*, Texturas e Redes Neurais Convolucionais. Todas estas técnicas requerem algum tipo de conhecimento prévio de forma a classificar os pixels ou regiões da imagem corretamente.

Mais concretamente, métodos de Redes Neurais Convolucionais têm, recentemente, sido alvo de muito interesse dentro âmbito de visão de computador, não só, mas também, devido ao foco crescente em *Big Data*. De facto, estas técnicas têm como base o treino de um modelo computacional, a partir de um conjunto, preferencialmente massivo, de dados de forma a aprender a classificar um pixel ou região de interesse corretamente. Vários padrões podem ser capturados, através das várias camadas presentes em Redes Neurais Convolucionais, e, no momento em que esteja treinado apropriadamente, o modelo não necessita de intervenção manual.



## Segmentação em Imagens de Ultra-som Modo-B

Todas estas técnicas foram aplicadas, de alguma forma, para auxiliar, ou automatizar completamente, o diagnóstico de doenças e/ou o seguimento de lesões em imagens de ultra-som modo-B, demonstrando, assim, quão importante é ter a possibilidade de, se aplicável, localizar corretamente a região de interesse e padronizar a medição da informação quantitativa através de algoritmos automáticos.

Em suma, a utilização de imagens de ultra-som modo-B irá, muito provavelmente, continuar a crescer, logo ter um entendimento claro das técnicas de segmentação disponíveis atualmente ajudará ao desenvolvimento e/ou otimização das tecnologias no campo respectivo.



## Capítulo 3

# Redes Neurais Convolucionais

Neste capítulo introduz-se conceitos básicos sobre Redes Neurais Convolucionais, o que são, em que consistem, como funcionam e algumas das tarefas e arquiteturas mais populares em anos recentes.

### 3.1 Redes Neurais

Redes Neurais tratam-se de sistemas computacionais inspirados nas redes neuronais biológicas do cérebro humano. Uma Rede Neuronal, na sua definição mais básica, consiste num grafo dirigido, onde cada nó produz uma saída composta de acordo com as suas entradas. Redes Neurais foram exploradas em várias áreas de estudo, como por exemplo, reconhecimento de padrões e processamento de sinais.

Redes Neurais enquadram-se no âmbito de *Machine Learning* que, segundo [M<sup>+</sup>97], pode ser definido por:

*A computer program is said to learn from experience  $E$  with respect to some class of tasks  $T$  and performance measure  $P$ , if its performance at tasks in  $T$ , as measured by  $P$ , improves with experience  $E$ .*

Logo, uma rede neuronal tem como objetivo aprender a realizar uma determinada tarefa, através de um conjunto de dados que lhe é fornecida, da forma mais eficiente possível (de acordo com algum tipo de métrica estabelecida).

#### 3.1.1 Arquitetura de uma Rede Neuronal

A arquitetura de uma rede neuronal consiste, tipicamente, em camadas, que por sua vez consistem em nós interligados. Cada ligação entre os nós está associada a um valor, denominado de peso, que contém informação sobre o sinal de entrada. Cada peso tem como responsabilidade processar o sinal que está a ser comunicado, através da sua excitação ou inibição.

Uma rede neuronal tradicional contém três tipos de camadas: a camada de entrada, camadas ocultas e a camada de saída. Uma rede recebe um determinado conjunto de dados na sua camada de entrada que, por sua vez, comunica com as camadas ocultas. Estas estão ligadas entre si, através de ligações associadas a um determinado peso que, referido anteriormente, são responsáveis por processar os dados. No fim, as camadas ocultas enviam os dados processados para a camada de saída, onde estes são, finalmente, caracterizados ou classificados.

Existem dois tipos de redes neurais: *feedforward* e *feedback*. Uma rede *feedforward* é unidirecional, isto é, transmite dados da entrada para a saída. Não existe nenhum tipo de conexão cíclica que permita a transmissão de dados no sentido contrário, impedindo, assim, que a saída de cada camada afete a entrada de camadas anteriores (camadas mais perto da camada de entrada). Por outro lado, redes *feedback* contêm um ou mais ciclos que permitem a transmissão de informação em ambos os sentidos. A figura 3.1 mostra um exemplo de uma rede *feedforward*.

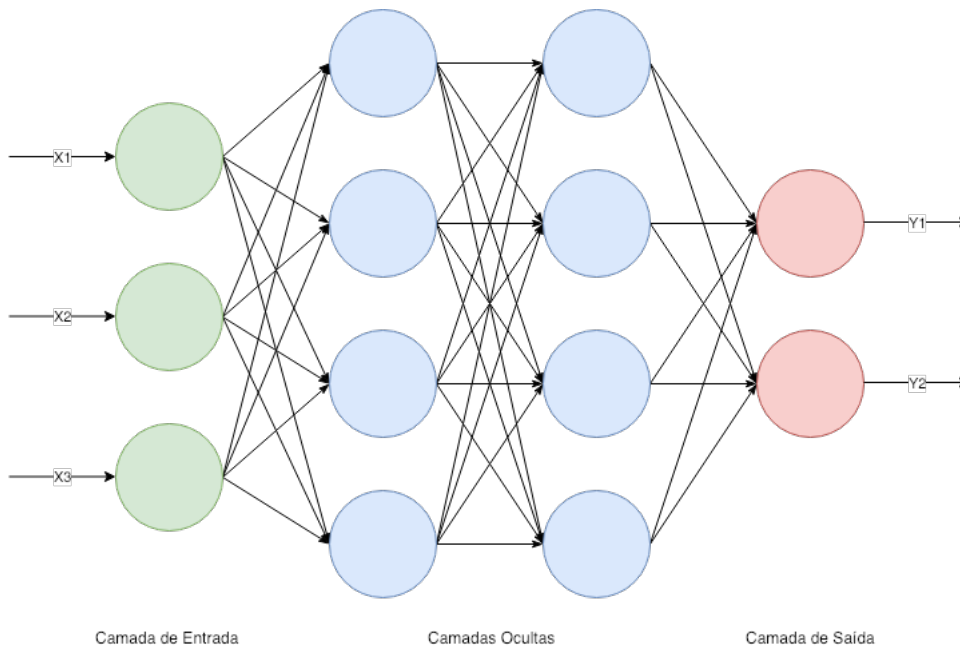


Figura 3.1: Ilustração de uma Rede *Feedforward* com uma camada de entrada, duas ocultas e uma de saída onde  $X_i$  e  $Y_i$  representam, respetivamente, as entradas e saídas da rede

### 3.1.2 Neurónio

Tipicamente, refere-se aos nós que constituem as camadas das redes neurais como neurónios. Um neurónio consiste num conjunto de dados de entrada, pesos e uma função de ativação. Em redes *feedforward*, cada neurónio calcula o produto escalar dos dados de entrada com os pesos (um valor designado de *bias* também pode ser adicionado). Logo, um neurónio  $i$  numa camada  $l$  pode ser definido como:

$$u_i^{(l)} = \sum_i w_i^{(l)} x_i^{(l-1)} + b^{(l)} \quad (3.1)$$

onde  $i$  e  $l$  representam, respetivamente, os índices do neurónio e da camada,  $w$  representa os pesos,  $x$  os dados de entrada e  $b$  o *bias*. É norma referir-se aos pesos e ao *bias* como parâmetros.

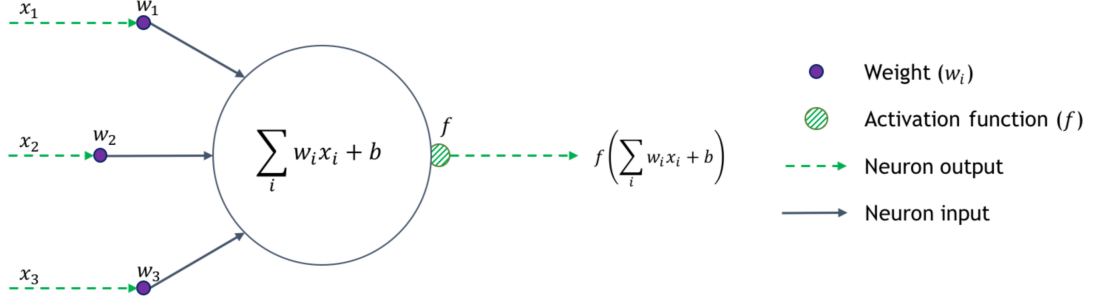


Figura 3.2: Exemplo de um neurónio artificial. Retirado de [Wan17]

De seguida, é aplicada uma função de ativação,  $f(u)$ , à soma resultante de forma a introduzir não-linearidade e limitar o intervalo dos valores de saída. Finalmente, o resultado de cada neurónio é propagado para a próxima camada.

**Função de Ativação** Como referido na secção 3.1.2, aplica-se, tipicamente, uma função de ativação não-linear ao resultado de cada neurónio. Estas funções são necessárias porque impedem que os resultados finais devolvidos pela camada de saída sejam, simplesmente, uma representação linear dos dados de entrada da rede. Existem vários tipos de funções de ativação:

- *Sigmoid* - traduz os valores de entrada para valores dentro do intervalo  $[0, 1]$  (figura 3.3 (a));

$$y = \sigma(x) = \frac{1}{1 + e^{-x}} \quad (3.2)$$

- *Softmax* - transforma os valores de entrada em probabilidades;

$$y_i = \frac{e^{I_i}}{\sum_{j=1}^n e^{I_j}} \quad (3.3)$$

- Tangente Hiperbólica (TanH) - traduz os valores de entrada para valores dentro do intervalo  $[-1, 1]$  (figura 3.3 (b));

$$y = 2\sigma(2x) - 1 \quad (3.4)$$

- *Rectifier Linear Unit (ReLU)* - transforma os valores de entrada negativos em zero (figura 3.3 (c)).

$$y = \max(0, x) \quad (3.5)$$

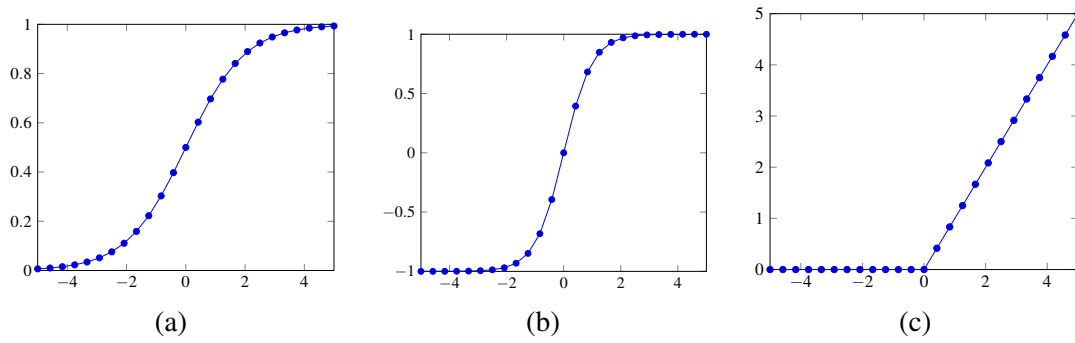


Figura 3.3: Representações gráficas das funções (a) *Sigmoid*, (b) *Tangente Hiperbólica* e (c) *ReLU*

### 3.1.3 Processo de Aprendizagem

Aprendizagem, em Redes Neurais, refere-se à modificação ou ajustamento dos pesos das ligações entre os neurónios de uma rede. Aprendizagem classifica-se em uma de três categorias:

- **Supervisionada** - Processo dependente. Neste tipo de aprendizagem, a rede recebe um determinado conjunto de dados juntamente com o conjunto de resultados desejados respetivo. Desta forma, redes deste tipo comparam os resultados obtidos com os desejados e a diferença é utilizada para ajustar os pesos das ligações até a rede devolver os resultados desejados;
- **Não Supervisionada** - Processo independente. Redes deste tipo recebem dados sem indicações do que é desejados desses dados. Desta forma, estas redes tentam agrupar automaticamente os dados de acordo com a sua semelhança;
- **Reforçada** - Este tipo de redes tentam reforçar a aprendizagem sobre um determinado conjunto de dados considerado crítico. Processo semelhante à aprendizagem supervisionada mas com menos informação. Da mesma forma, utilizam os dados para ajustar os pesos das ligações mas para receber mais informação crítica.

A aprendizagem divide-se em três fases: treino, validação e teste. O treino é a principal. Durante esta fase, a rede tem como objetivo principal reduzir a função de perda. Esta mede o erro entre os resultados obtidos e os de referência.

**Função de Perda** Como referido anteriormente, uma função de perda  $L$  é dada pela relação entre os resultados obtidos  $y$  e os de referência  $o$ . Esta tem como responsabilidade avaliar a precisão da rede desenvolvida de acordo com o conjunto de dados fornecido.

Existem uma variedade de funções de perda e cada rede deve usar a mais apropriada para a tarefa que pretende concluir. Uma das mais comuns é o Erro Quadrático Médio (*Mean Squared Error (MSE)*), que pode ser definida como:

$$L = \frac{1}{K} \sum_{i=1}^K (y_i - o_i)^2 \quad (3.6)$$

onde  $K$  representa o número total de classes e  $y_i$  e  $o_i$  representam, respetivamente, os resultados de referência e obtido para a classe de índice  $i$ .

Outra função popular é a Perda Logarítmica (*Log Loss* ou *Cross-Entropy Loss*), que pode ser definida como:

$$L = - \sum_i y_i \log(s_i) \quad (3.7)$$

onde  $s$  representa a distribuição de probabilidades obtida.

**Backpropagation** Tipicamente, redes neuronais seguem o esquema de uma *Backpropagation Neural Network (BPN)*, que consiste numa camada de entrada, pelo menos uma oculta e uma de saída. Como o nome sugere, ocorre um processo de propagação no sentido das camadas mais profundas para as menos profundas (mais perto da camada de entrada). Mais especificamente, é calculado a diferença entre o resultado obtido e o de referência que depois é propagado de volta para a camada inicial. *Backpropagation* funciona em três passos:

- *Forward Pass* - Passo inicial onde os valores de saída da cada camada são calculados de acordo com os valores de entrada. Segue o fluxo natural de uma rede e atravessa todos os neurónios desde a primeira até à última camada;
- Cálculo do erro - Segundo passo onde é calculada a diferença entre os resultados obtidos e os de referência, através da função de perda;
- Atualização dos pesos - Último passo onde se utiliza a diferença calculada para atualizar os pesos, a começar pela última camada de volta até à primeira. Os algoritmos responsáveis por esta atualização denominam-se de otimizadores. Deve-se selecionar o otimizador apropriado para a tarefa que se pretende concluir.

Maior parte dos otimizadores utilizam algoritmos baseados no Gradiente Descendente (*Gradient Descent*) da função de perda. O Gradiente é uma generalização multi-variável duma derivada, usado em funções com várias variáveis. Resulta num vetor composto pelas derivadas parciais da função  $f(x_1, \dots, x_n)$ , que pode ser definido por:

$$\nabla f = \left( \frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right) \quad (3.8)$$

onde  $x_1, \dots, x_n$  representam as variáveis que compõem a função  $f$ .

O Gradiente Descendente trata-se de um algoritmo iterativo, isto é, as variáveis são atualizadas a cada iteração (ou época) do algoritmo na direção negativa do gradiente da função calculado durante essa iteração. O Gradiente Descendente pode ser definido como:

$$v_{n+1} = v_n - \alpha \nabla f(v_n) \quad (3.9)$$

onde  $v$  representa o vetor com as variáveis,  $n$  é o índice da iteração e  $\alpha$  é um valor positivo, próximo de zero, denominado de taxa de aprendizagem.

Logo, de forma a descobrir os pesos e *bias* ótimos de uma Rede Neuronal, é necessário calcular o gradiente da função de perda  $\nabla L(x, W_n, b_n)$ . No entanto, o conjunto de dados utilizado para treinar uma rede é muito grande e, como tal, não é prático treinar a rede com o conjunto inteiro duma vez só. Desta forma, o gradiente,  $\nabla L$ , é calculado através da média de todos os gradientes,  $\nabla L_x$ , que são calculados para cada dado  $x$ .

Em aplicações de grande escala, calcular o gradiente para cada dado do conjunto pode ser um processo muito moroso. Adicionalmente, durante o processo de computação, o conjunto precisa de ser guardado em memória que, por sua vez, é carregado a cada iteração. De seguida lista-se e descreve-se dois dos otimizadores mais populares, em anos recentes, que permitem reduzir o tempo de treino e minimizar o consumo de memória:

- Gradiente Descendente Estocástico (*Stochastic Gradient Descent (SGD)*) [Stab] - Calcula o gradiente para um dado único do conjunto que é computacionalmente mais eficiente do que calcular para o conjunto inteiro. Adicionalmente, esse gradiente pode ser aplicado a sub-conjuntos (ou lotes) do conjunto de dados. O algoritmo começa por seleccionar, aleatoriamente, um sub-conjunto de dados ( $N_b$ ) cujo gradiente é calculado através de:

$$\nabla L_{lote} = \frac{1}{N_b} \sum_{x \in lote} \nabla L_x \quad (3.10)$$

De seguida, os pesos e *bias* são atualizados. Este processo repete-se até que o conjunto de treino inteiro tenha sido utilizado para atualizar os parâmetros;

- *Adaptive Moment Estimation (Adam)* [Staa] - Neste algoritmo, a taxa de aprendizagem é adaptada para cada um dos parâmetros. O algoritmo guarda, em memória, uma variável que monitoriza a soma do quadrado dos gradientes para cada parâmetro. A atualização dum parâmetro  $w$  define-se por:

$$\begin{aligned} m_{w_n} &= \beta_1 + (1 - \beta_1) \nabla L(w_n), \\ v_{w_n} &= \beta_2 + (1 - \beta_2) (\nabla L(w_n))^2, \\ \hat{m}_{w_n} &= \frac{m_{w_{n+1}}}{1 - \beta_1}, \\ \hat{v}_{w_n} &= \frac{v_{w_{n+1}}}{1 - \beta_2}, \\ w_{n+1} &= w_n - \alpha \frac{\hat{m}_{w_n}}{\sqrt{\hat{v}_{w_n} + \varepsilon}} \end{aligned} \quad (3.11)$$

onde  $\varepsilon$  é um valor numérico, próximo de zero, usado para impedir divisões por zero e  $\beta_1$  e  $\beta_2$  são taxas de declínio exponenciais.

## 3.2 Dados de Treino, Validação e Teste

O conjunto de dados usado num algoritmo de *Machine Learning* para a construção de um modelo divide-se em três grupos:



- Treino - Estes são os dados que são usados para treinar o modelo. São os dados que são usados para ajustar os parâmetros da Rede Neuronal, isto é, são os dados a partir do qual o algoritmo aprende;
- Validação - Estes são os dados que são usados para validar o modelo. São os dados usados para controlar o efeito de *overfitting* (3.3), logo, não são usados para ajustar os parâmetros da rede mas sim para verificar que qualquer aumento de precisão sobre os dados de treino, traduz-se num aumento de precisão sobre um conjunto de dados, até agora, desconhecido. Se a precisão sobre os dados de treino aumenta mas a precisão sobre os dados de validação mantém-se a mesma, ou diminui, então o modelo está a sofrer de *overfitting* e o treino deve ser interrompido;
- Teste - Estes são os dados que são usados para testar o modelo. São os dados que são usados para testar a utilidade do modelo em situações do mundo real.

Estes três grupos têm de ser completamente independentes, isto é, não podem conter dados em comum. Se tal acontecer, durante a fase de teste, a precisão aumentará, devido ao facto de que o modelo já conhece os dados e, por sua vez, as suas respectivas características, e a precisão final será incorreta.

**Aumento dos Dados** O termo Generalização refere-se à capacidade de um modelo devolver os resultados corretos para dados, até agora, desconhecidos. Quantos mais dados o algoritmo tiver acesso durante a fase de treino, melhor será a capacidade de generalização do modelo gerado. Em situações em que a quantidade de dados disponível é demasiado pequena e tendo em conta que apenas uma parte desses dados vai ser usada para treino, uma solução viável passa pela criação de dados “falsos”, que depois podem ser adicionados ao conjunto de dados de treino.

Ao criar dados “falsos” deve-se ter sempre o cuidado de não criar dados inválidos para a tarefa que se pretende resolver. Por exemplo, em tarefas de classificação, basta a aplicação de uma série de transformações aos dados originais para duplicar o número de exemplos. No entanto, é preciso ter em atenção que algumas transformações, mesmo que simples, não são admissíveis dependendo do conjunto de dados a que são aplicadas. Por exemplo, em tarefas de reconhecimento de caracteres, inversões horizontais e/ou rotações de 180° não se admitem, pois deixaria de haver distinção entre os caracteres ‘b’ e ‘d’ ou ‘6’ e ‘9’.

### 3.3 *Overfitting, Underfitting* e Capacidade

Durante a fase de treino, o modelo sofrerá um determinado desvio dos dados de referência, designado de Erro de Treino. Da mesma forma, o desvio sofrido durante a fase de teste designa-se de Erro de Teste. Tendo isto, e o objetivo principal de um algoritmo de *ML* em consideração, pode-se avaliar um modelo pela sua capacidade de [Lan13]:

1. Reduzir o Erro de Treino;

## 2. Reduzir a diferença entre os Erros de Treino e de Teste.

Desta forma, é possível perceber os conceitos de *overfitting* e *underfitting*. *Overfitting* ocorre quando o modelo falha na regra número 2 e o intervalo entre os Erros de Treino e de Teste torna-se muito grande. *Underfitting* ocorre quando o modelo falha na regra número 1 e não é capaz de obter um Erro de Treino baixo o suficiente.

Por outras palavras, *overfitting* ocorre quando um modelo aprende demasiado sobre os dados de treino, isto é, aprende demasiados detalhes sobre os dados, incluindo ruído, o que influencia o performance do modelo, em dados novos, negativamente. Por sua vez, *underfitting* ocorre quando um modelo não é capaz de modelar os dados de treino, nem capaz de generalizar para dados novos.

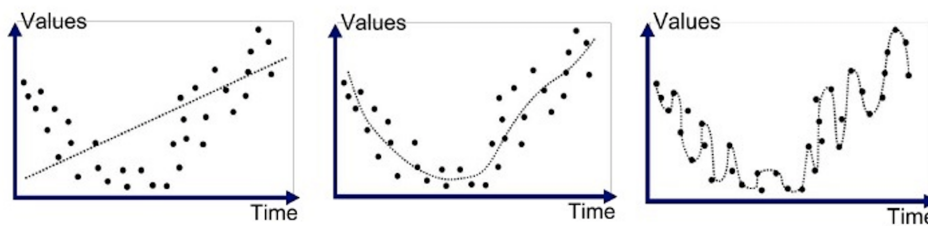


Figura 3.4: Exemplos de *Underfitting*, ótimo e *Overfitting* (da esquerda para a direita). Retirado de [Bha18]

A capacidade refere-se ao número de funções que um modelo consegue selecionar como uma possível solução. De facto, um modelo devolve os melhores resultados quando a sua capacidade é proporcional à complexidade da sua tarefa e aos dados de treino. Modelos com uma baixa capacidade não são práticos em tarefas complexas e têm a tendência a sofrer de *underfitting*. Da mesma forma, modelos com uma capacidade superior à necessária têm a tendência a sofrer de *overfitting*. Assim sendo, podemos definir a capacidade como a medida através da qual se estima a probabilidade de um modelo sofrer de *underfitting* ou *overfitting*.

## 3.4 Tarefas

Redes Neurais Convolucionais podem ser utilizadas para resolver uma série de tarefas dentro do âmbito de visão de computador. De seguida encontram-se quatro das tarefas mais populares e comuns:

- Classificação - Esta tarefa tem como objetivo receber uma imagem como entrada e devolver uma previsão da classe correspondente;
- Localização - Nesta tarefa não só se devolve a previsão da classe, como, também, se delimita o objeto de interesse com uma *bounding box*;
- Detecção - Trata-se de uma extensão da tarefa de localização que é aplicada a todos os objetos detetados na imagem (não necessitam de ser todos da mesma classe);

- Segmentação - Semelhante à de localização exceto que delimita, especificamente, os contornos do objeto de interesse ou classifica todos os pixels da imagem (segmentação semântica).

As figuras 3.5 (a), (b), (c) e (d) demonstram exemplos de cada uma das tarefas referidas.

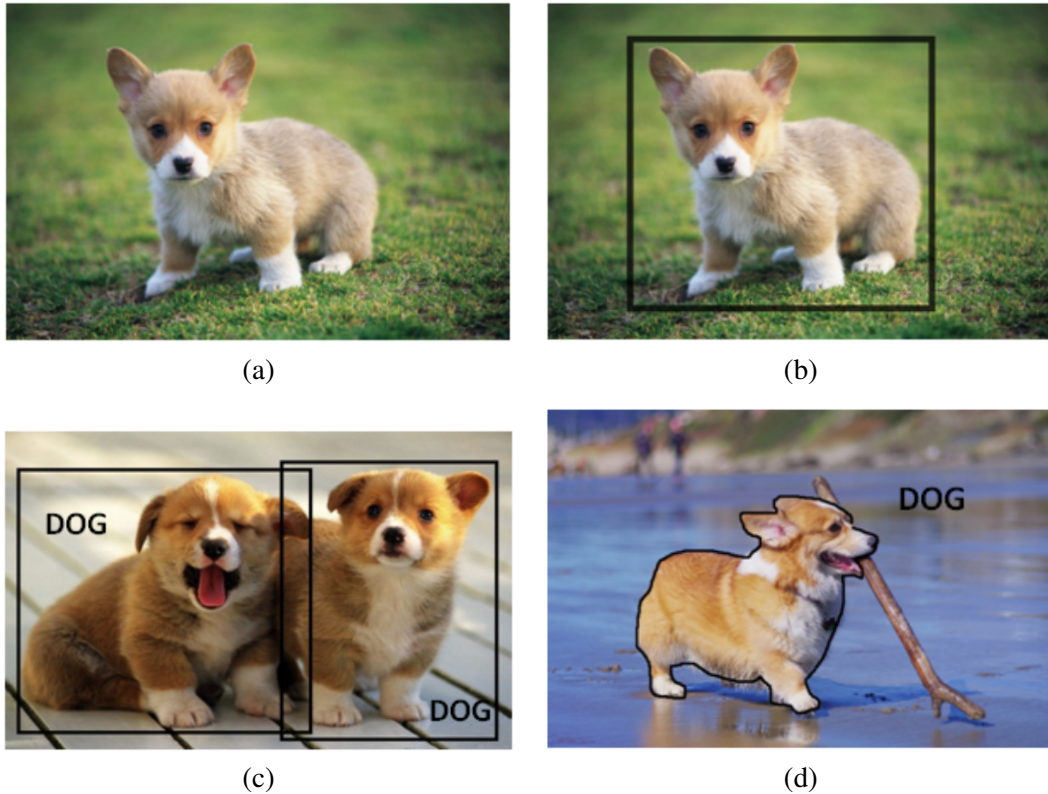


Figura 3.5: Exemplos de tarefas de (a) classificação, (b) localização, (c) detecção e (d) segmentação. Retirados de [Des16]

### 3.5 Camadas

Redes Neurais Convolucionais implementam uma combinação de vários tipos de camadas. De seguida segue-se uma listagem e descrição de cada uma delas. A figura 3.6 ilustra um exemplo de uma Rede Neuronal Convolucional.

**Hiperparâmetros** Maior parte das camadas possuem parâmetros configuráveis, parâmetros que são definidos antes do treino iniciar. Estes designam-se de hiperparâmetros. São os parâmetros que definem a estrutura da rede (por exemplo, número de unidades ocultas) e como é que a rede é treinada (por exemplo, a taxa de aprendizagem). Cada uma das camadas que se segue define o seu conjunto de hiperparâmetros respetivos.

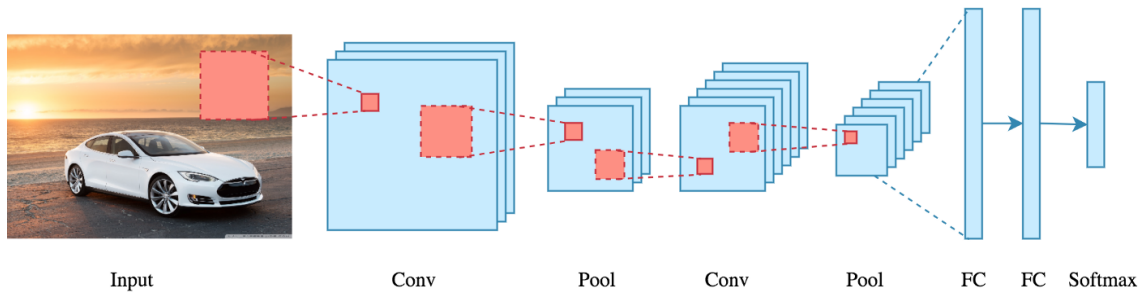


Figura 3.6: Exemplo da arquitetura convencional de uma Rede Neuronal Convolucional. Retirado de [Der17]

### 3.5.1 Convolução

Estas camadas são a razão pela qual as Redes Neurais Convolucionais têm o nome que têm, porque aplicam, obrigatoriamente, pelo menos uma operação de convolução. Tipicamente, a primeira camada de uma Rede Neuronal Convolucional é sempre uma convolução.

Estas camadas implementam uma operação de convolução 2D num volume 3D que está organizado em duas dimensões espaciais e uma dimensão de profundidade, isto é, tomando uma imagem a cores com 32 pixels de largura e altura como exemplo, esta seria interpretada como um vetor multi-dimensional (ou tensor)  $32 \times 32 \times 3$ , onde o valor 3, referente à dimensão de profundidade, representa os valores RGB da cada pixel da imagem. Esta operação utiliza um conjunto  $N_k$  de tensors de pesos, denominados de filtros ou *kernels*,  $F = f_1, \dots, f_{N_k}$  que são convolucionados com uma entrada  $x$ , produzindo um conjunto  $N_k$  de vetores 2D  $z_k$ , denominados de mapas de ativação (ou mapas de características):

$$z_k = f_k * x \quad (3.12)$$

onde  $*$  representa o operador de convolução. Ou seja, o que acontece em concreto, tendo em conta uma imagem a cores  $32 \times 32$  como exemplo, é o seguinte:

1. Como referido anteriormente, a imagem é interpretada como um vetor 3D  $32 \times 32 \times 3$ ;
2. Um filtro de tamanho, por exemplo,  $5 \times 5 \times 3$  percorre a imagem completamente desde o canto superior esquerdo até ao canto inferior direito (de cima para baixo, da esquerda para a direita);
3. Cada posição da região coberta (pelo filtro) na imagem é multiplicada pela posição correspondente no filtro e, de seguida, os resultados são somados;
4. O passo 3 repete-se até a imagem ter sido completamente percorrida;
5. Cada posição única do filtro sobre a imagem devolveu um valor que, por sua vez, é armazenado numa matriz 2D. Neste caso,  $28 \times 28 \times 1$ , sendo este o mapa de ativação resultante.

| Input |   |   |   |   | Filtro           |   |   |
|-------|---|---|---|---|------------------|---|---|
| 1     | 1 | 1 | 0 | 0 | 1                | 0 | 1 |
| 0     | 1 | 1 | 1 | 0 | 0                | 1 | 0 |
| 0     | 0 | 1 | 1 | 1 | 1                | 0 | 1 |
| 0     | 0 | 1 | 1 | 0 |                  |   |   |
| 0     | 1 | 1 | 0 | 0 |                  |   |   |
|       |   |   |   |   | Mapa de Ativação |   |   |
|       |   |   |   |   | 4                | 3 | 4 |
|       |   |   |   |   | 2                | 4 | 3 |
|       |   |   |   |   | 2                | 3 | 4 |

Figura 3.7: Exemplo de uma Convolução sem *padding*

A utilização de vários filtros em vez de só um resulta num volume de vários mapas de ativação. Desta forma, se o(s) filtro(s) correlacionarem bem como uma determinada região da entrada, o seu resultado será forte, ou seja, um valor elevado, caso contrário, é um valor baixo.

Frequentemente, é boa prática acrescentar linhas e colunas de zeros em redor da imagem de forma a preservar as suas dimensões após a convolução (*zero-padding*). É possível, também, de forma a diminuir o custo computacional, mover o filtro em passos não unitários, ou seja, mover o filtro de 2 em 2 ou 3 em 3 em vez de 1 em 1 (*strides*). Cada um destes parâmetros é configurável como hiperparâmetros da operação: as dimensões dos filtros  $f$ , o valor de *stride*  $s$  e o valor de *padding*  $p$ . Logo, é possível calcular as dimensões da saída de qualquer convolução com:

$$m = \frac{n - f + 2p}{s} + 1 \quad (3.13)$$

onde  $m$  e  $n$  representam, respetivamente, as dimensões da saída e da entrada.

### 3.5.2 Fully Connected

Matematicamente, uma camada *Fully Connected* (*FC*) pode ser interpretada como uma função que aplica uma transformação linear a um vetor de dimensão  $I$  e devolve um vetor de dimensão  $O$ . Ou seja, uma camada *FC* opera com vetores (tensors 1D) e devolve outro vetor onde cada posição é uma soma pesada de todas as posições do vetor de entrada. Os pesos são associações entre as posições dos vetores de entrada e saída.

$$O_j = \sum_{i=1}^n W_{(i,j)} * I_i \quad (3.14)$$

onde  $I_i$  representa o valor da posição de índice  $i$  do vetor de entrada,  $O$  representa o vetor de saída e  $W_{(i,j)}$  representa o peso associado aos valores das posições de índice  $i$  e  $j$  dos vetores de entrada e saída, respetivamente. Os hiperparâmetros das camadas *FC* são as dimensões dos vetores de

## Redes Neurais Convolucionais

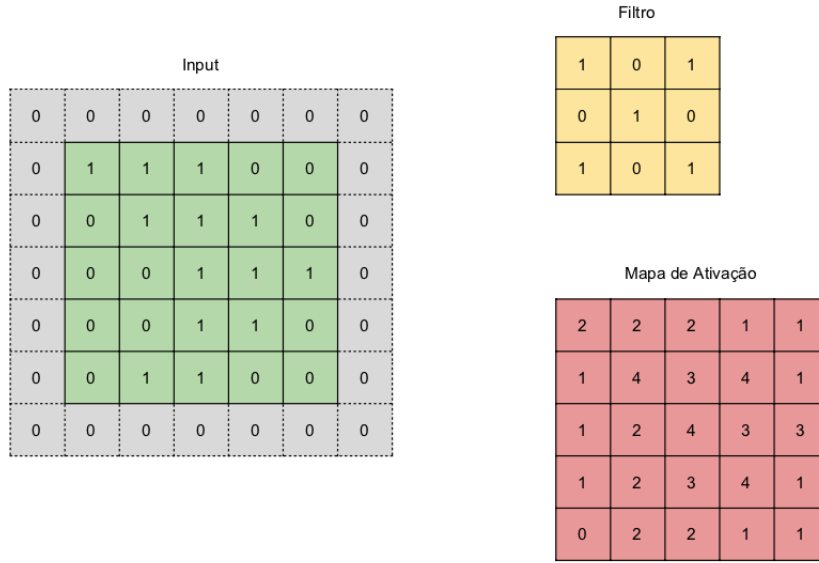


Figura 3.8: Exemplo de uma Convolução com *padding* de 1

entrada e saída, onde o tamanho do vetor de entrada pode ser inferido através do tamanho do vetor de saída da camada anterior.

Idealmente, uma arquitetura de uma Rede Neuronal Convolucional contém vários tipos de camadas, mas as arquiteturas que contêm apenas uma ou mais camadas *FC* designam-se de *Perceptrons* ou *Multi Layer Perceptrons*. Estes tipo de arquiteturas são as mais básicas que existem, visto que não exploram a correlação entre as características da entrada e, como tal, não escalam bem quando o número de pesos é  $n^2$ . Como tal, estas arquiteturas são usadas, normalmente, em tarefas de classificação ou regressão, quando o número de características é pequeno.

### 3.5.3 Pooling

Em Redes Neurais Convolucionais, as camadas de *Pooling*, também referidas como as camadas de *downsampling*, são usadas, normalmente, para reduzir o número de características de um tensor e, consequentemente, o custo computacional da rede em geral. Pode ser definido como:

$$p_R = P_{i \in R}(z_i) \quad (3.15)$$

onde  $P$  representa uma função de *Pooling* sobre uma região de pixels  $R$ . Existem vários tipos de *Pooling* sendo a *Max Pooling* a mais popular que, de forma semelhante a uma convolução, “desliza” um filtro, normalmente  $2 \times 2$  com um *padding* do mesmo tamanho, sobre a entrada mas, em vez de multiplicar e somar os valores, seleciona o valor máximo. Outros tipos incluem *Average Pooling* e *L2 Pooling*. Camadas de *Pooling* são configuráveis em termos de dimensões do filtro, valor de *stride* e valor de *padding*. A figura 3.9 demonstra um exemplo de *Max Pooling*.

| Input |   |   |   | Filtro |   |
|-------|---|---|---|--------|---|
| 1     | 0 | 2 | 3 | 6      | 8 |
| 4     | 6 | 6 | 8 | 3      | 4 |
| 3     | 1 | 1 | 0 |        |   |
| 1     | 2 | 2 | 4 |        |   |

Figura 3.9: Exemplo de *Max Pooling* com um filtro  $2 \times 2$  e *stride* de 2

### 3.5.4 Batch Normalization

É importante manter os parâmetros de uma camada estáveis e dentro de um determinado intervalo, dependendo da função de ativação associada. Para tal, uma camada de *Batch Normalization*, colocada antes de cada camada de ativação faz com que as ativações dos neurónios da entrada sigam uma distribuição Gaussiana sobre as dimensões do conjunto de dados de treino. Por outras palavras, melhora o fluxo de gradientes, permite taxas de aprendizagem superiores, reduz a dependência da inicialização dos parâmetros, diminui a necessidade de uma camada *Dropout* e reduz o tempo de treino.

### 3.5.5 Dropout

A camada *Dropout* existe principalmente para mitigar o efeito de *overfitting*. Esta camada tem uma função relativamente simples: escolher um conjunto aleatório de ativações e eliminá-las completamente, colocando os seus valores a zero. Desta forma, a rede é forçada a ser redundante, isto é, deverá ser capaz de devolver a saída correta para uma determinada tarefa mesmo que algumas ativações sejam eliminadas. É necessário ter em consideração que esta camada só deve ser utilizada na fase de treino e não na de teste. As figuras 3.10 (a) e (b) ilustram exemplos.

## 3.6 Arquiteturas

Desde 1990 que têm sido desenvolvidas várias arquiteturas de Redes Neurais Convolucionais. Esta secção tem como objetivo listar algumas das arquiteturas mais populares [Hay94].

### 3.6.1 Classificação

**AlexNet** [KSH12] Esta foi a arquitetura que popularizou o uso de Redes Neurais Convolucionais em tarefas de visão de computador. Demonstrada pela primeira vez na competição *ImageNet Large-Scale Visual Recognition Challenge*, venceu a todos os outros métodos propostos com erros



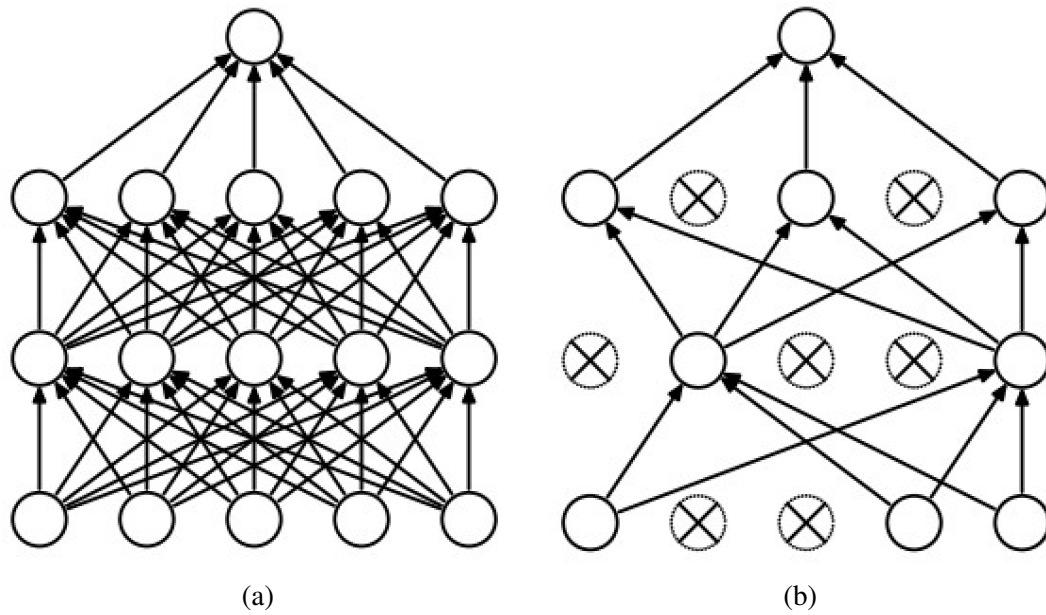


Figura 3.10: Exemplo de uma rede convencional (a) sem e (b) com *dropout*. Retirados de [dS]

Top5<sup>1</sup> de 15.3% e 16.4% (uma diferença de 11% em relação ao segundo lugar). Esta arquitetura continha 5 camadas convolucionais e 3 camadas *FC*. Usou e popularizou a função de ativação *ReLU*, as camadas de *Max Pooling*, as camadas de *Dropout* e o uso de placas gráficas unidade de processamento central do treino de modelos. Arquitetura especificada na figura 3.11.

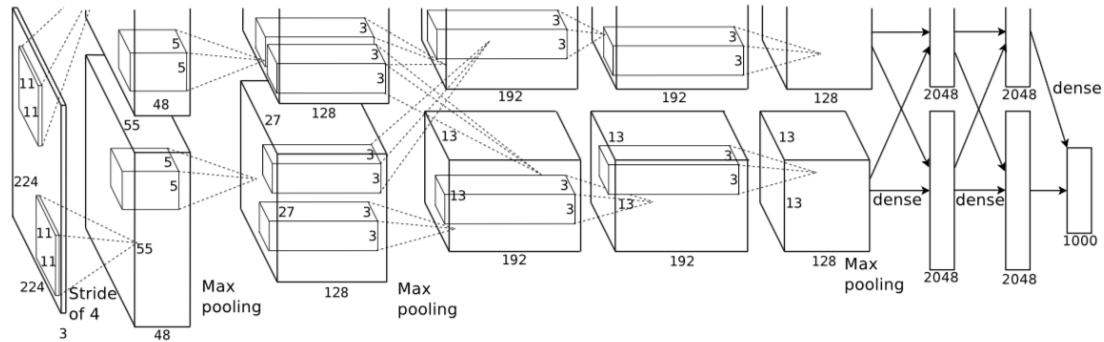


Figura 3.11: Estrutura da arquitetura *AlexNet*. Retirado de [KSH12]

**VggNet** [SZ14] Também conhecida como *VeryDeepNet*, foi apresentada durante a competição *ImageNet* de 2014 (mas não a vencedora) com uma arquitetura com quase 140 milhões de parâmetros. Demonstrou que a profundidade de uma rede é um componente crucial para obter um bom performance. Deu uso a filtros  $3 \times 3$  com *stride* e *padding* de valor 1 e filtros  $2 \times 2$ , nas camadas de

<sup>1</sup>fração de imagens para as quais as classes corretas não estão dentro das 5 classes mais prováveis devolvidas pelo modelo



*Max Pooling*, com *stride* de valor 2. Obtiveram erros Top1 de 24.8% e 7.5% para as arquiteturas de 16 e 19 camadas, respetivamente. Arquiteturas especificadas na figura 3.12.

| ConvNet Configuration       |                        |                        |                                     |                                     |                                                  |
|-----------------------------|------------------------|------------------------|-------------------------------------|-------------------------------------|--------------------------------------------------|
| A                           | A-LRN                  | B                      | C                                   | D                                   | E                                                |
| 11 weight layers            | 11 weight layers       | 13 weight layers       | 16 weight layers                    | 16 weight layers                    | 19 weight layers                                 |
| input (224 × 224 RGB image) |                        |                        |                                     |                                     |                                                  |
| conv3-64                    | conv3-64<br>LRN        | conv3-64               | conv3-64                            | conv3-64                            | conv3-64                                         |
| maxpool                     |                        |                        |                                     |                                     |                                                  |
| conv3-128                   | conv3-128              | conv3-128<br>conv3-128 | conv3-128                           | conv3-128                           | conv3-128                                        |
| maxpool                     |                        |                        |                                     |                                     |                                                  |
| conv3-256<br>conv3-256      | conv3-256<br>conv3-256 | conv3-256<br>conv3-256 | conv3-256<br>conv3-256<br>conv1-256 | conv3-256<br>conv3-256<br>conv3-256 | conv3-256<br>conv3-256<br>conv3-256<br>conv3-256 |
| maxpool                     |                        |                        |                                     |                                     |                                                  |
| conv3-512<br>conv3-512      | conv3-512<br>conv3-512 | conv3-512<br>conv3-512 | conv3-512<br>conv3-512<br>conv1-512 | conv3-512<br>conv3-512<br>conv3-512 | conv3-512<br>conv3-512<br>conv3-512<br>conv3-512 |
| maxpool                     |                        |                        |                                     |                                     |                                                  |
| conv3-512<br>conv3-512      | conv3-512<br>conv3-512 | conv3-512<br>conv3-512 | conv3-512<br>conv3-512<br>conv1-512 | conv3-512<br>conv3-512<br>conv3-512 | conv3-512<br>conv3-512<br>conv3-512<br>conv3-512 |
| maxpool                     |                        |                        |                                     |                                     |                                                  |
| FC-4096                     |                        |                        |                                     |                                     |                                                  |
| FC-4096                     |                        |                        |                                     |                                     |                                                  |
| FC-1000                     |                        |                        |                                     |                                     |                                                  |
| soft-max                    |                        |                        |                                     |                                     |                                                  |

Figura 3.12: As diferentes configurações de *VggNet*. Retirado de [SZ14]

***GoogLeNet* [SLJ<sup>+</sup>15]** Criada pela Google e, também apresentada durante a competição *ImageNet* de 2014 (vencedora), esta arquitetura provou que profundidade nem sempre é o suficiente e demonstrou ser capaz de obter resultados tão bons ou melhores do que as outras implementações até agora observadas. Esta arquitetura explorou a ideia de utilizar filtros de dimensões diferentes, em paralelo, cujos resultados são depois concatenados numa única saída. Para mitigar o grande número de parâmetros resultantes desta abordagem são utilizados convoluções  $1 \times 1$  antes das convoluções  $3 \times 3$  e  $5 \times 5$ . Estes conjuntos de convoluções paralelas denominam-se de módulos *Inception*. Composta por 22 camadas no total e 9 módulos *Inception*, obteve um erro Top5 de 6.67%. Arquitetura especificada na figura 3.13.

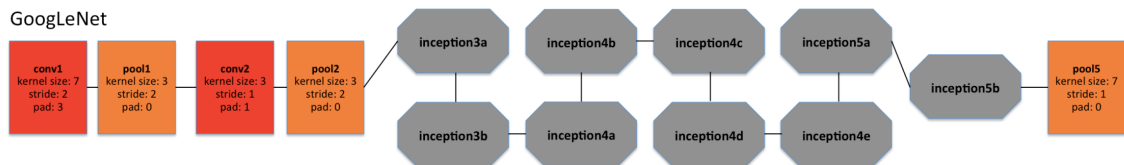


Figura 3.13: Estrutura da arquitetura *GoogLeNet*. Cada módulo *Inception* consiste em seis camadas de convoluções e uma de *pooling*. Retirado de [HCRG<sup>+</sup>16]

***ResNet* [HZRS16]** Criada pela Microsoft e vencedora da competição *ImageNet* de 2015, esta é uma das arquiteturas modernas mais profundas e complexas que existe, com um total de 152 camadas. Ultrapassou os recordes nas tarefas de classificação, deteção e localização. Os autores

exploram o facto de que ocorre uma degradação da precisão do treino em redes profundas, quando estas começam a convergir, e, como tal, implementaram blocos residuais, com o objetivo de saltar certas conexões e encurtar o caminho entre a saída e as camadas iniciais. Obteve um erro Top5 de 3.5%. Arquitetura especificada na figura 3.14.

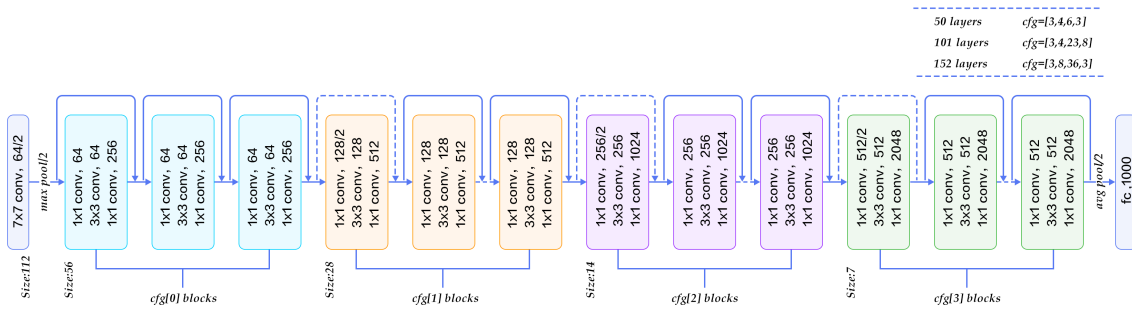


Figura 3.14: Estrutura da arquitetura *ResNet*. Retirado de [Das17]

### 3.6.2 Detecção

**R-CNN [GDDM14]** Método proposto, para a deteção em imagens 2D, baseado em regiões. Este método divide-se em três passos principais: geração de regiões diferentes que têm a maior probabilidade de conter um objeto (através de Procura Seletiva [UVDSGS13]), extração do vetor de características de cada região através de uma Rede Neuronal Convolucional e, por fim, utilização de *Support Vector Machines (SVMs)* lineares, treinadas para cada classe, que recebem os vetores como entrada e devolvem uma classificação. Os vetores também passam por uma regressão *bounding box* de forma a obter as coordenadas mais precisas. Obtiveram uma classificação *Mean Average Precision (mAP)* de 31.4%, na competição *ImageNet* de 2013 (uma melhoria de 7% em relação ao recorde anterior). Arquitetura especificada na figura 3.15.

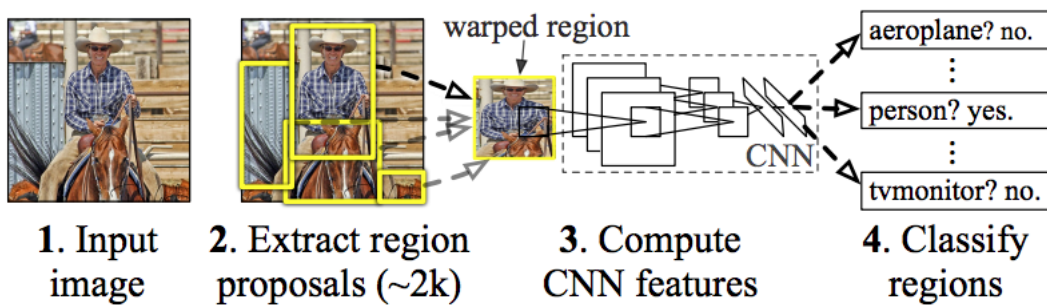


Figura 3.15: Representação de alto nível de *R-CNN*. Retirado de [GDDM14]

**Fast R-CNN [Gir15]** Uma melhoria à implementação original, *R-CNN*, através da resolução de três problemas principais: o treino estava dividido em várias fases (as referidas no parágrafo anterior), era computacionalmente intensivo e extremamente lento (53 segundos por imagem). Nesta implementação os classificadores *SVM* são substituídos por um classificador *Softmax* e são

introduzidas camadas de *ROI Pooling*, semelhante a *Pooling* convencional, mas usadas somente sobre as regiões propostas, em vez da imagem inteira original. Melhora o tempo de treino em 9 vezes e o de teste em 213 vezes. Arquitetura especificada na figura 3.16.

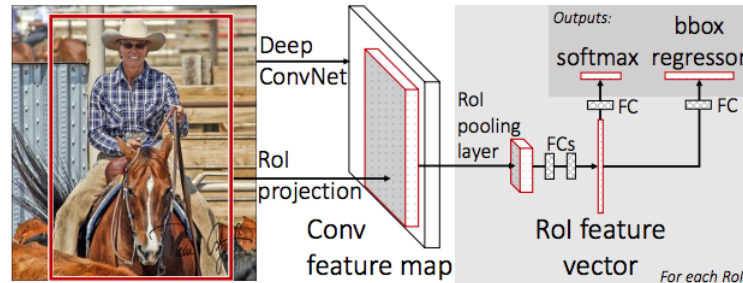


Figura 3.16: Representação de alto nível de *Fast R-CNN*. Retirado de [Gir15]

**Faster R-CNN [RHGS15]** Embora *Fast R-CNN* tenha apresentado melhorias significantes em relação a *R-CNN*, estas continuam a depender de um método de geração de regiões, processo responsável pela maioria do tempo de processamento destes métodos. Neste caso, é proposto um método denominado *Region Proposal Networks* a seguir à última camada convolucional. Esta rede é capaz de extrair as regiões propostas através dos mapas de ativação da última camada e, a partir daí, segue o mesmo esquema que *R-CNN* (*ROI Pooling*, *FC*, classificação e regressão). Arquitetura especificada na figura 3.17.

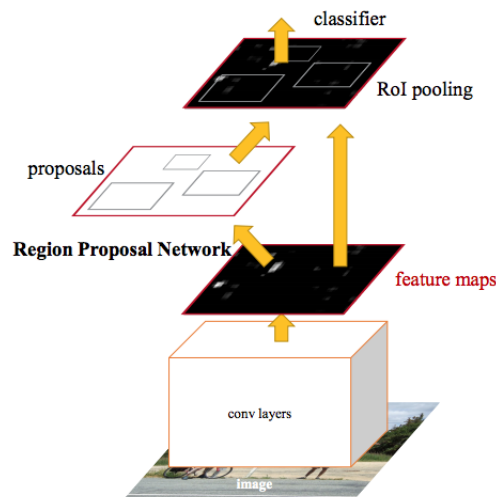


Figura 3.17: Representação de alto nível de *Faster R-CNN*. Retirado de [RHGS15]

### 3.6.3 Segmentação Semântica

**Deconvolution Network (DNN) [NHH15]** Tendo em consideração que o objetivo da segmentação semântica é classificar cada pixel da imagem de entrada, é de esperar a necessidade de uma

camada final capaz de devolver uma imagem com as mesmas dimensões espaciais que a imagem de entrada. No entanto, em arquiteturas de Redes Neurais Convolucionais convencionais, à medida que a profundidade aumenta, as dimensões da saída diminuem, a não ser que se use *padding* em todas as camadas (uma solução inválida devido ao uso de camadas de *Max Pooling*). Uma solução proposta passa pelo uso de “desconvoluções” (ou convoluções transpostas) e *Max UnPooling* que são, respetivamente, operações inversas às operações de convolução e *Max Pooling*, criando assim uma Rede Neuronal “Desconvolucional”. Arquitetura exemplo especificada na figura 3.18.

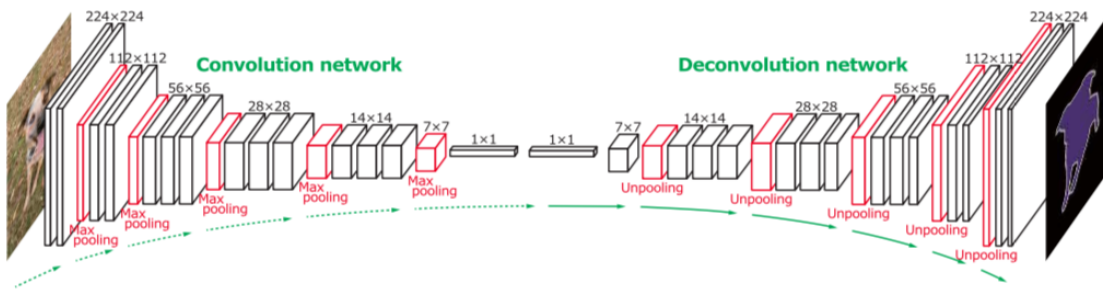


Figura 3.18: Estrutura da arquitetura *Deconvolution Network*. Retirado de [NHH15]

**Fully Convolutional Network (FCN) [LSD15]** Redes que, como o nome indica, são compostas por convoluções sem uma única camada *FC*, normalmente encontradas no final de uma Rede Neuronal. A primeira fase das *FCN*, para a segmentação semântica, é semelhante às *VggNet*, no entanto, todas as camadas *FC* são convertidas em convoluções. A saída final passa, de seguida, por um processo de *upsampling* através de convoluções transpostas e camadas de *UnPooling*. Arquitetura exemplo especificada na figura 3.19.

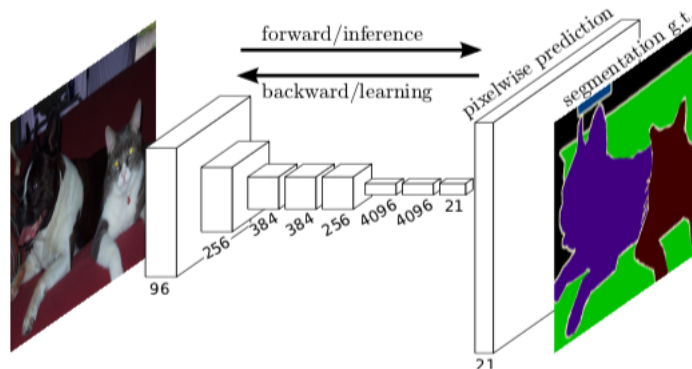


Figura 3.19: Exemplo de uma *FCN*. Retirado de [LSD15]

## Capítulo 4

# Implementação da Rede Neuronal

Este capítulo detalha as várias etapas de desenvolvimento do projeto e a implementação da solução proposta.

### 4.1 Equipamento e Ferramentas

Em termos de equipamento, o C-BER forneceu um computador sem o qual não teria sido possível o desenvolvimento deste projeto. O computador tem as seguintes especificações:

- Sistema Operativo - Ubuntu 16.04.5 LTS;
- Processador - 8x Intel Core i7-6700K @ 4.00GHz;
- Memória - 42GB;
- Placa Gráfica - NVIDIA GeForce GTX 1080 (8GB)

A linguagem de programação selecionada foi *Python* [Pyta] devido à disponibilidade de uma grande coleção de bibliotecas relacionadas com processamento de imagens e redes neuronais. Destas bibliotecas selecionou-se *OpenCV* [Ope] para todos os processos relacionados com a preparação das imagens e *PyTorch* [PyTb] para a implementação da Rede Neuronal. Adicionalmente, deu-se uso a *Tensorboard* [Ten] para o registo de toda a informação relacionada com o treino, validação e teste do método proposto.

### 4.2 Conjunto de Imagens

O desenvolvimento deste projeto teve acesso a dois conjuntos de imagens que continham, respetivamente, 50 e 150 imagens de ultra-som modo-B longitudinais da ACC. O conteúdo das imagens varia entre:

## Implementação da Rede Neuronal

- presença da Artéria Carótida Comum (ACC) (figura 4.1 (a));
- presença da ACC com uma ou mais instâncias de placa (figura 4.1 (b));
- presença da ACC e da Veia Jugular Interna (VJI) (figura 4.1 (c));
- ou uma combinação dos dois últimos casos (figura 4.1 (d)).

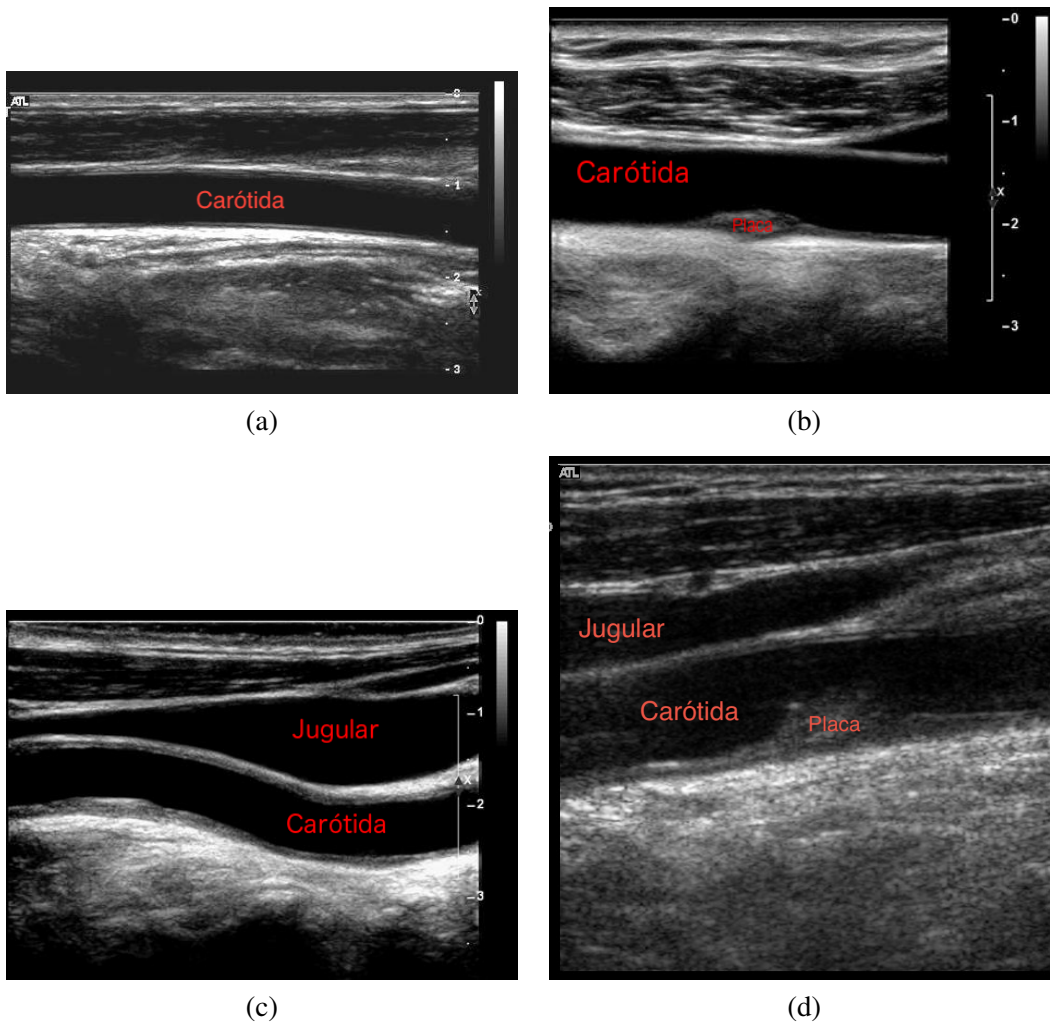


Figura 4.1: Ilustrações de imagens com (a) ACC, (b) ACC e placa, (c) ACC e VJI e (d) ACC, VJI e placa

Todas as imagens têm um ficheiro de texto associado com as segmentações manuais correspondentes. No caso do conjunto de 50, cada imagem tem três ficheiros associados: A1, A2 e B1, onde A1 e A2 são segmentações feitas pelo mesmo médico em alturas diferentes e B1 são de um médico diferente. Por outro lado, no conjunto de 150, cada imagem tem exatamente um ficheiro associado que contém somente a segmentação da parede longínqua da artéria (parede da artéria carótida mais próxima do limite inferior da imagem). Exemplos podem ser observados na figura 4.2.



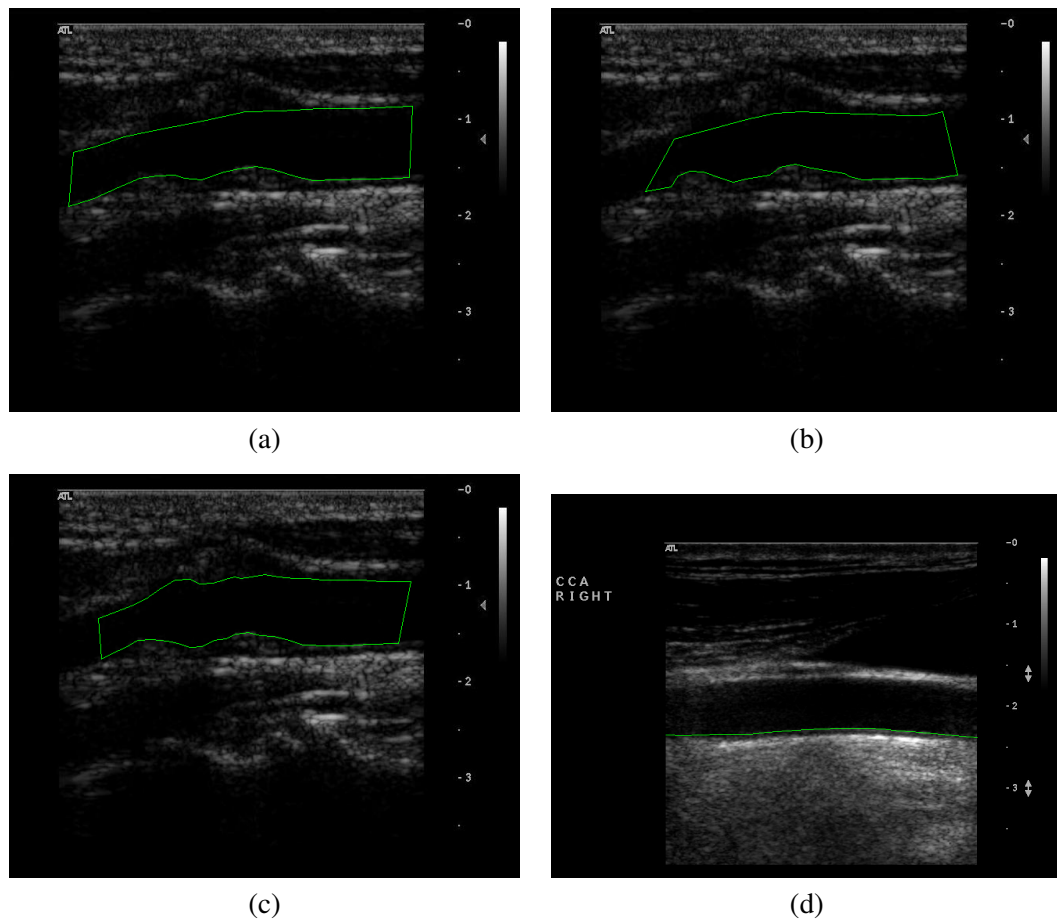


Figura 4.2: Exemplos das segmentações (a) A1, (b) A2 e (c) B1 da mesma imagem e exemplo de uma segmentação da (d) parede longínqua dos conjuntos respetivos

### 4.2.1 Preparação das Imagens

Devido aos casos expostos na secção anterior (4.2), foi necessário preparar ambos os conjuntos de imagens de modo a obter as máscaras de referência (*ground truth*) necessárias para o treino da rede.

Devido a motivos de qualidade de imagem, nem todas as imagens têm segmentações que delimitam o comprimento completo do lúmen da carótida (exemplo na figura 4.2). Como tal, todas as imagens tiveram de ser, novamente, segmentadas manualmente. De forma a poupar tempo e manter a qualidade das segmentações, as novas segmentações foram criadas a partir das existentes. Isto é, com o uso de uma ferramenta de anotação de imagens foi possível criar as novas segmentações manuais com as existentes como ponto de partida.

Em primeiro lugar, o conjunto de 50 contém três grupos de segmentações diferentes e apenas um é necessário. O grupo A1 foi selecionado por motivos de qualidade de segmentação e pelo comprimento do lúmen que estas já delimitavam. Em segundo lugar, o conjunto de 150, como referido anteriormente, contém apenas as delimitações da parede longínqua, portanto foi necessário completar a segmentação através da delimitação da parede próxima (parede da artéria carótida

mais próxima do limite superior da imagem). Todas as novas segmentações manuais foram feitas com a supervisão de um médico experiente.

No final deste passo obteve-se um conjunto de 200 imagens de ultra-som modo-B da ACC e das suas máscaras respetivas. As máscaras (exemplo na figura 4.3 (c)) tratam-se de imagens binárias onde o objeto de interesse é branco (ou 1) e todo o resto é preto (ou 0).

### 4.2.2 Seleção da Região de Interesse

Apesar das novas segmentações, que delimitam o máximo do comprimento possível do lúmen da carótida, algumas imagens continuavam com segmentações incompletas. Tal deve-se não só à qualidade das imagens mas também aos artefactos que algumas delas apresentam. Como tal, as imagens tiveram de passar por mais uma fase de processamento, mais especificamente, pela seleção da região de interesse.

Assumindo que se utilizava as imagens completas (como no exemplo da figura 4.3 (a)) para o treino da rede, existiria sempre a possibilidade de que a diferença entre a máscara obtida e a de referência seria superior à realidade. Isto é, a rede sofreria na sua aprendizagem por segmentar, corretamente, o lúmen da carótida que, de acordo com as máscaras correspondentes, não é lúmen. Como se pode observar na figura 4.3 (b), o lúmen não coberto pela região branca (correspondente à segmentação manual A1) poderia ser corretamente segmentado como parte do lúmen pela rede. No entanto, segundo a segmentação fornecida pela máscara, essa segmentação estaria parcialmente errada. Logo, de forma a eliminar esta margem de erro incorreta, todas as imagens foram recortadas verticalmente de acordo com a localização do lúmen fornecida pela máscara correspondente. Desta forma, garante-se que a diferença calculada pela função de perda só terá em consideração a zona da imagem que, segundo a máscara, contém lúmen.

Todas as imagens e máscaras passaram por um processo de seleção de região de interesse como o que pode ser observado no exemplo 4.3 (d).

### 4.2.3 Divisão em Conjuntos de Treino, Validação e Teste

Como referido na secção 3.2, é necessário dividir o conjunto de imagens em três grupos distintos: treino, validação e teste. Para conjuntos limitados, é norma utilizar uma proporção 80%/20% [Ken18] para a divisão do conjunto em conjuntos de “treino” e teste, respetivamente. A partir do conjunto de “treino” obtém-se o verdadeiro conjunto de treino e o de validação, com a mesma proporção de 80%/20%. Como tal, o conjunto original de 200 imagens foi dividido, aleatoriamente, em 128 (80% de 160) imagens de treino, 32 (20% de 160) de validação e 40 (20% de 200) de teste.

## 4.3 Implementação da Solução Proposta

Esta secção descreve a solução proposta para a resolução do problema, tal como todos os detalhes da sua implementação.



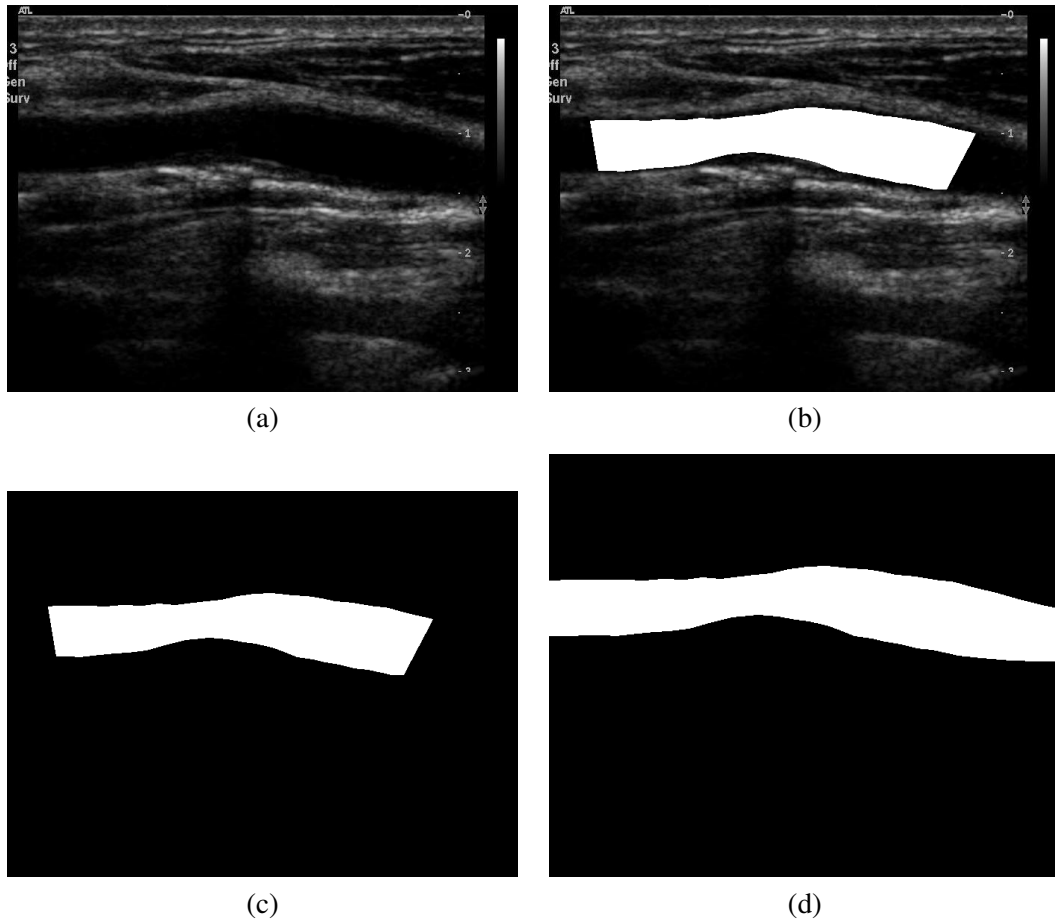


Figura 4.3: Exemplo da (a) imagem original e (b) com a máscara A1 sobreposta. (a) Máscara A1 original e (d) região de interesse selecionada

### 4.3.1 Arquitetura da Rede

A arquitetura desenvolvida para a resolução do problema baseia-se na *U-Net* que, como referido na secção 2.2.6, foi criada principalmente para resolver problemas de detecção e segmentação em imagens biomédicas. A *U-Net* já foi testada e aprovada várias vezes para resolver uma série de problemas de segmentação em imagens de ultra-som [RFB15], [BDS18], [ZSM<sup>+</sup>18].

A arquitetura original *U-Net* consiste em 19 convoluções (cada uma seguida de uma ativação *ReLU*), 4 convoluções transpostas, 4 *Max Poolings* e 4 conexões, onde cada conexão é um processo de dois passos: recorte da imagem e concatenação dessa com a imagem transposta do mesmo nível (a rede contém 5 níveis). Cada conexão necessita, obrigatoriamente, de recortar a imagem porque nenhuma das convoluções aplica *padding* às suas imagens de entrada, resultando em máscaras com dimensões espaciais menores. Ao mesmo tempo, após cada *Max Pooling* a dimensão espacial das imagens é reduzida a metade sendo assim, necessário que a dimensão espacial das imagens seja obrigatoriamente números pares antes de cada *Max Pooling*. A arquitetura pode ser observada na figura 4.4.

No caso deste projeto, a rede desenvolvida tem algumas diferenças em relação à original. Foi

## Implementação da Rede Neuronal

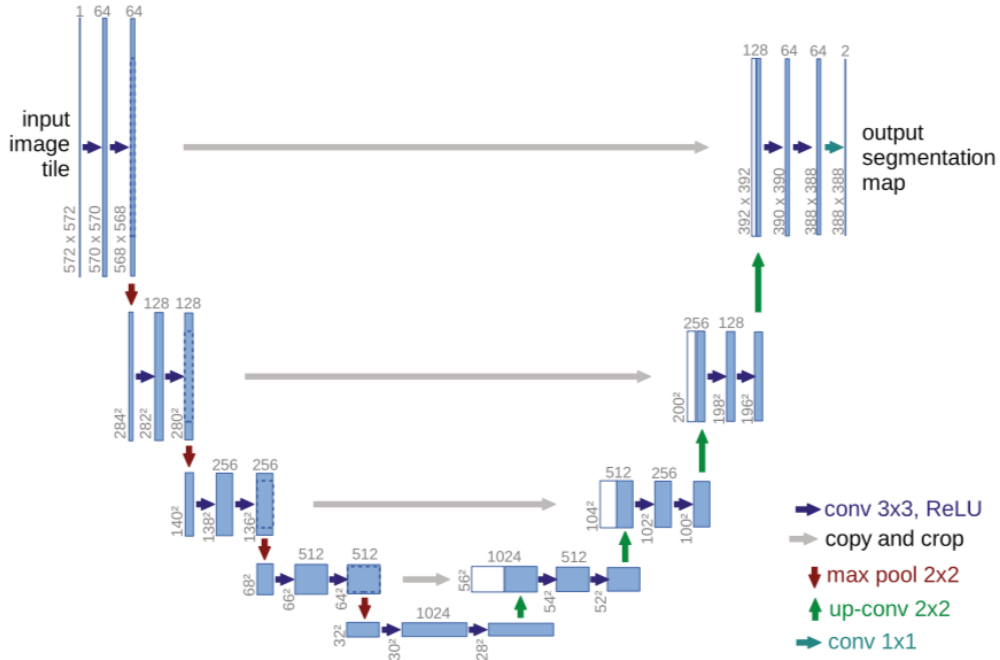


Figura 4.4: Representação da arquitetura *U-Net*. Retirado de [RFB15]

acrescentado, a seguir a cada camada de convolução e antes da ativação, uma camada de *Batch Normalization* devido às vantagens especificadas na secção 3.5.4. A saída final da rede passa por uma ativação *Sigmoid*, de forma a normalizar os valores de cada pixel entre 0 e 1. Essa saída, por sua vez, passa por um *threshold* de 0.5, garantindo assim, que a máscara devolvida é uma imagem binária. Por fim, foi definido que cada camada de convolução teria *padding* adaptativo. Isto é, o *padding* das convoluções é calculado de acordo com as dimensões dos filtros para que a imagem de saída de cada convolução tenha as mesmas dimensões espaciais que a imagem de entrada. Desta forma, elimina-se a necessidade de redimensionar as máscaras de referência para que estas tenham as mesmas dimensões espaciais que as máscaras obtidas (ou vice-versa). Normalmente, refere-se a este tipo de *padding* como “*SAME*” (semelhante ou constante). Exemplos das relações entre tamanho de filtros e *padding* encontram-se na tabela 4.1.

| Filtro       | <i>Padding</i> |
|--------------|----------------|
| $3 \times 3$ | 1              |
| $5 \times 5$ | 2              |
| $7 \times 7$ | 3              |
| $9 \times 9$ | 4              |

Tabela 4.1: Exemplos de dimensões de filtros e o *padding* correspondente. Calculados de acordo com a equação 3.13 definida na secção 3.5.1

Adicionalmente, os parâmetros das convoluções, normais e transpostas, são inicializados de acordo com [GB10] e os *bias* são inicializados a 0. A inicialização dos parâmetros é importante,

## Implementação da Rede Neuronal

pois esta pode ser a diferença entre a rede convergir numa quantidade de tempo razoável, dependendo do tamanho do conjunto de dados, ou nunca convergir durante centenas ou milhares de iterações.

Por fim, foi acrescentado um nível extra à rede, fazendo com que a rede passasse a conter 23 convoluções, 5 convoluções transpostas, 5 *Max Poolings* e 5 conexões. Portanto, a arquitetura final pode ser observada na figura 4.5.

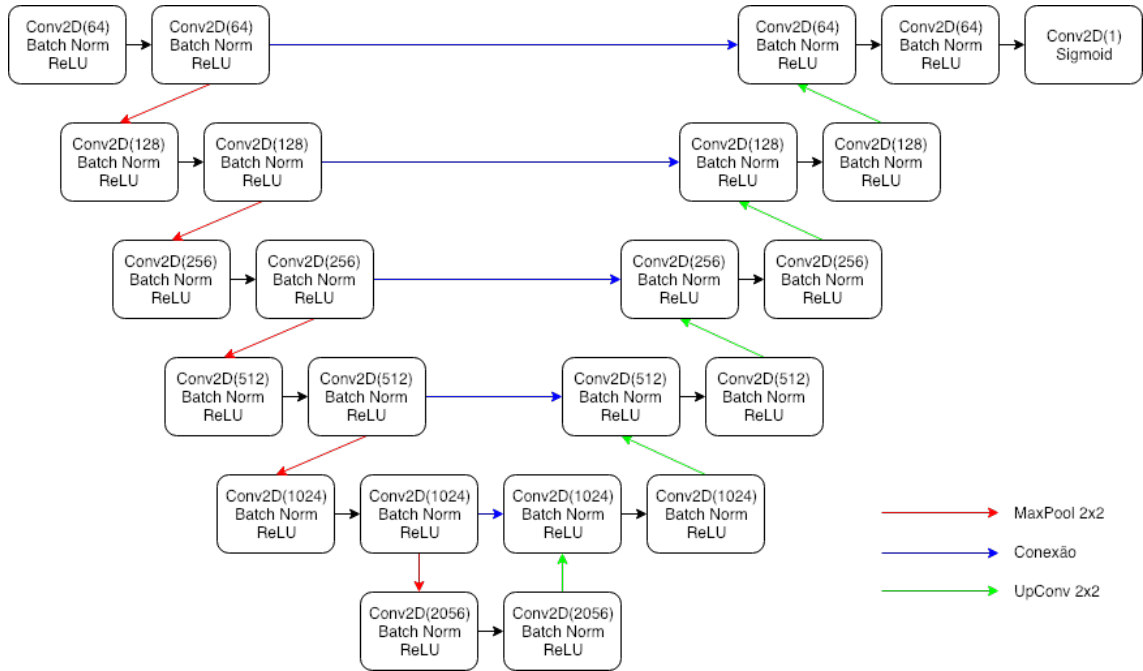


Figura 4.5: Arquitetura da rede desenvolvida

**Monitorização do Treino** Como referido anteriormente, o conjunto de imagens foi dividido em três grupos: treino, validação e teste. O grupo de validação é utilizado para monitorizar o treino do modelo e saber quando e que versão do modelo guardar em disco. Neste caso, para a monitorização do treino, é calculada e registada a pontuação DICE para cada imagem da validação seguida pela pontuação DICE média do grupo de validação completo. É guardado o modelo que devolver melhor pontuação DICE média no grupo de validação.

### 4.3.2 Redimensionamento das Imagens

Tendo em consideração, não só a restrição aplicada às resoluções espaciais das imagens pelas camadas de *Max Pooling* e pelas conexões, mas também devido a limitações de equipamento, todas as imagens são redimensionadas antes de serem processadas pela rede. As máscaras, sendo imagens binárias, são redimensionadas com uma interpolação *Nearest Neighbor*, de forma a que estas continuem a ser imagens binárias. Por outro lado, as imagens originais (após a seleção da região de interesse) são redimensionadas com uma interpolação *Cubic Spline*.

Interpolação refere-se à criação de novos dados a partir de um conjunto discreto de dados pontuais previamente conhecidos. A interpolação *Nearest Neighbor* é a mais básica e requer menos tempo de processamento que qualquer outro método de interpolação, pois só tem em consideração um pixel - o mais próximo do ponto interpolado. A interpolação *Cubic Spline* ou *Bicubic* considera os pixels vizinhos numa grelha  $4 \times 4$ . Sabendo que estes pixels estão a distâncias variadas do pixel desconhecido, os pixels mais próximos têm um maior peso no seu cálculo. É a interpolação mais utilizada em muitos problemas de processamento de imagem devido à sua relação de qualidade de imagem e tempo de processamento [Cam].

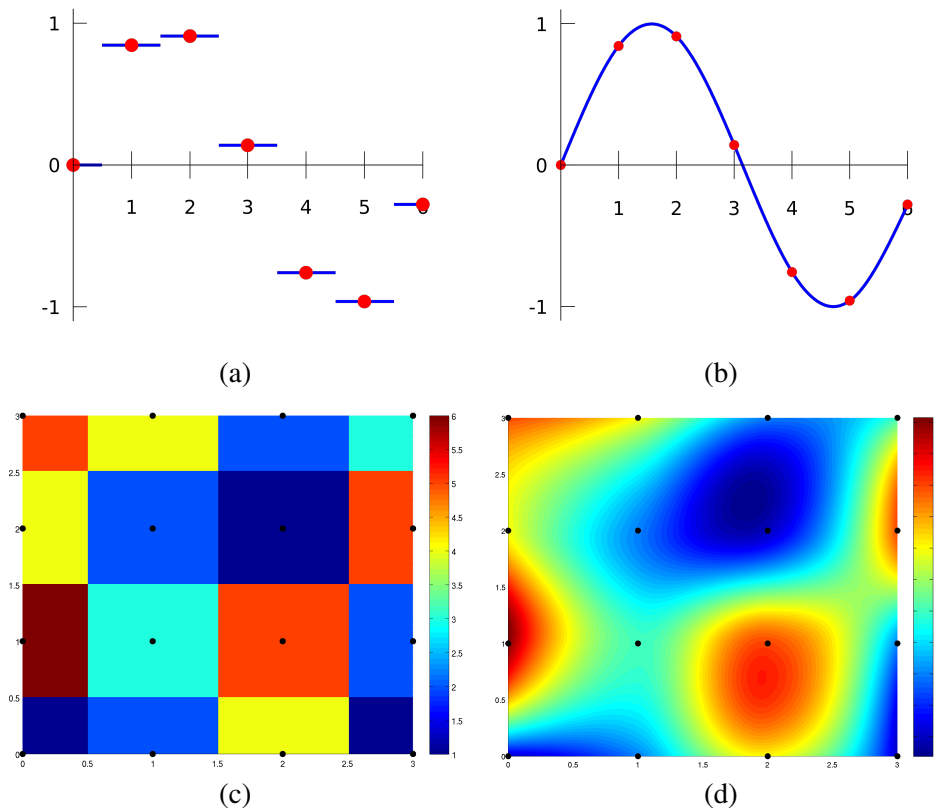


Figura 4.6: Representações gráficas e visuais das interpolações *Nearest Neighbor* [(a) e (c)] e *Cubic Spline* [(b) e (d)]. Retirado de [Wik]

### 4.3.3 Critério de Avaliação

Para comparação das máscaras obtidas e as de referência foi selecionado o coeficiente de DICE. Este pode ser definido como:

$$DICE = 2 \times \frac{|X \cap Y|}{|X| + |Y|} \quad (4.1)$$

onde  $X$  e  $Y$  representam, respetivamente, as máscaras obtida e de referência.  $|X|$  e  $|Y|$  referem-se à cardinalidade dos conjuntos  $X$  e  $Y$ . O coeficiente de DICE é um valor entre 0 e 1, que quanto

mais próximo de 1, maior a semelhança entre os conjuntos, e vice-versa.

**Função de Perda** Devido à sua eficiência enquanto critério de avaliação para a comparação de imagens 2D binárias [SDK<sup>+</sup>18], definiu-se o coeficiente de DICE, também, como a função de perda para o treino do modelo. Neste caso, sabendo que a função de perda é um valor que se pretende minimizar e que quanto maior a semelhança entre imagens, maior o coeficiente, este valor, quando calculado enquanto função de perda, tem de ser subtraído a 1. Isto é:

$$Perda = 1 - DICE \quad (4.2)$$

## 4.3.4 Otimizador

Este modelo utiliza o Gradiente Descendente Estocástico (ou *SGD*) para a propagação da perda e respetiva atualização dos parâmetros, a cada iteração. A taxa de aprendizagem inicial é 0.1 e é reduzida por um multiplicador de 0.1 a cada 20 épocas. Esta redução constante da taxa de aprendizagem impede que o modelo sofra de *overfitting*.

Foram testados modelos treinados com o *SGD* e o *Adam* como otimizadores, no entanto, verificou-se que os modelos treinados com *Adam* generalizam pior do que os modelos treinados com *SGD*. Adicionalmente, de acordo com [WRS<sup>+</sup>17] e [KS17], mesmo que *Adam* devolva melhores resultados durante o treino, isso não implica resultados igualmente bons durante a validação, ao contrário de *SGD*. Exemplos podem ser observados na figura 4.7.

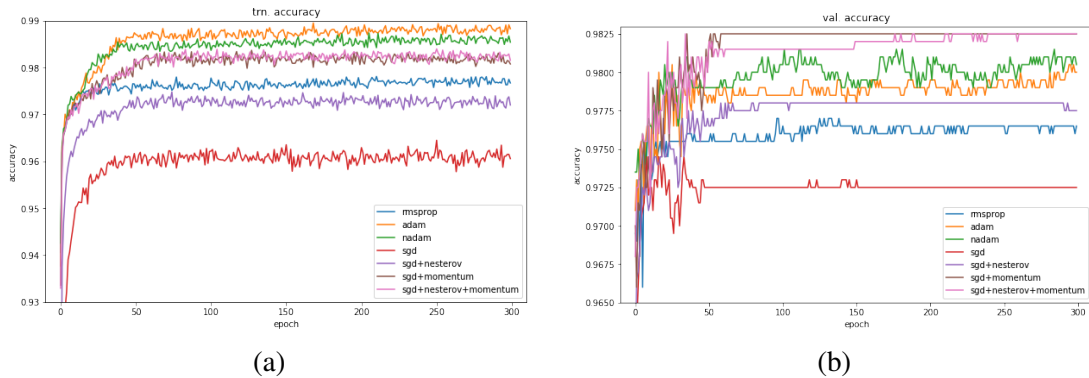


Figura 4.7: Precisão de diferentes otimizadores durante (a) o treino e (b) validação. Retirado de [Sha]

## 4.4 Configurações Experimentais

A secção que se segue apresenta as principais configurações experimentais e os resultados respetivos que foram executadas ao longo do desenvolvimento deste projeto. Todas tiveram algum tipo de impacto ou influência na decisão da configuração final seleccionada.

## Implementação da Rede Neuronal

De forma a estabelecer uma base daquilo que a rede é capaz sem nenhum tipo de otimização, o primeiro treino foi executado com os parâmetros definidos de acordo com o artigo da rede original *U-Net*. Os resultados podem ser observado nas figuras 4.8. Os parâmetros deste modelo são os seguintes:

| EP  | NC | NN | DF | TL | DI  | FP         | OT         | TA  | MO  |
|-----|----|----|----|----|-----|------------|------------|-----|-----|
| 100 | 1  | 5  | 3  | 1  | 576 | <i>BCE</i> | <i>SGD</i> | 0.1 | 0.9 |

Tabela 4.2: Parâmetros do treino inicial. EP = Épocas, NC = Número de Classes, NN = Número de Níveis da rede, DF = Dimensões dos Filtros, TL = Tamanho dos Lotes, DI = Dimensões das Imagens, FP = Função de Perda, OT = Otimizador, TA = Taxa de Aprendizagem, *BCE* = *Binary Cross-Entropy*. A EP e a TA foram definidas como 100 e 0.1 pois estas não estavam definidas no artigo

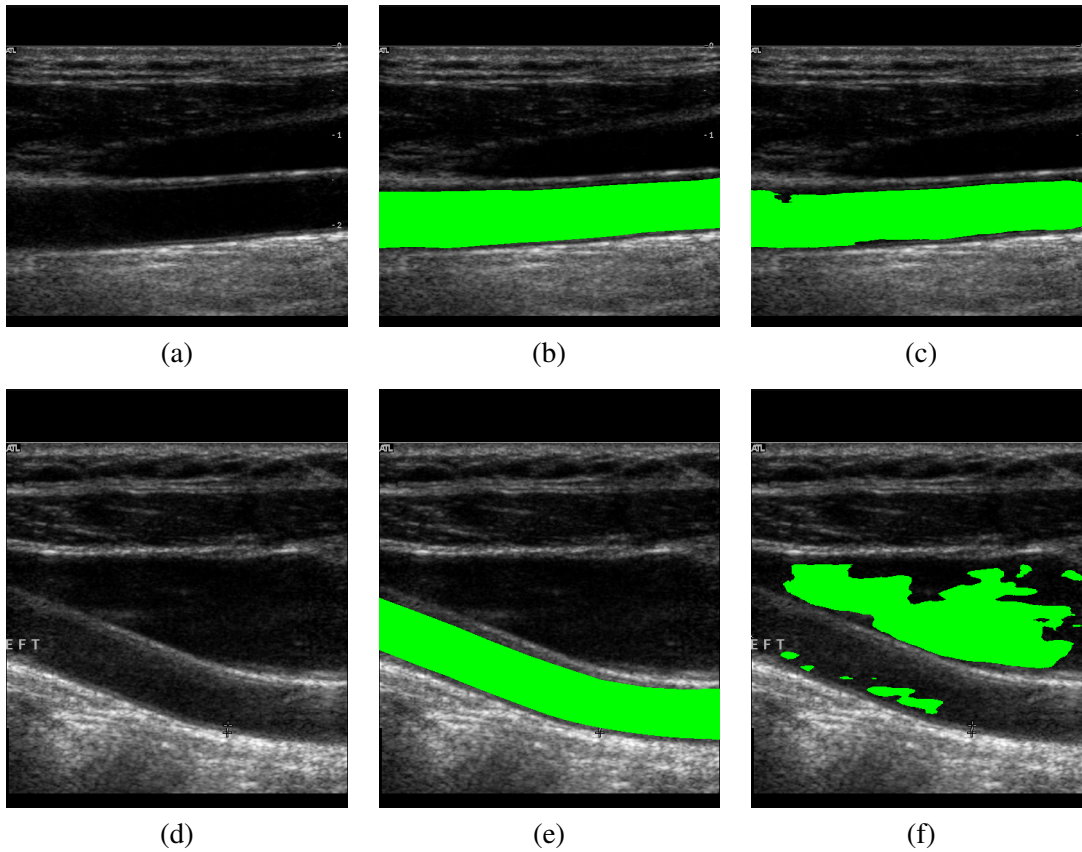


Figura 4.8: Pior e melhor imagens do conjunto de teste. (a) Melhor imagem (b) com máscara de referência e (c) com máscara obtida [DICE = 97.15%]. (d) Pior imagem (e) com máscara de referência e (f) máscara obtida [DICE = 5.69%]

Inicialmente, as figuras 4.8 (a), (b) e (c) parecem indicar que a rede foi boa escolha para a resolução do problema, mesmo só com otimizações mínimas, enquanto que as (d), (e) e (f) indicam o contrário. Este modelo obteve uma pontuação de DICE média de 65.43%, demonstrando que o melhor resultado de 97.15% não é representativo do conjunto de teste completo. Ao mesmo

tempo, este modelo serve de exemplo do que acontece quando o modelo não é treinado de acordo com o conjunto de validação. Isto é, este modelo treinou durante as épocas indicadas e não deu uso ao conjunto de validação para monitorizar o progresso do modelo, e simplesmente guardou o modelo devolvido no final da última época. Como tal, a pontuação da melhor imagem é resultado de *overfitting*, como se pode observar no gráfico 4.9. De qualquer maneira, este modelo também permitiu observar aquela que já tinha sido identificada como uma das principais dificuldades: a presença da VJI.

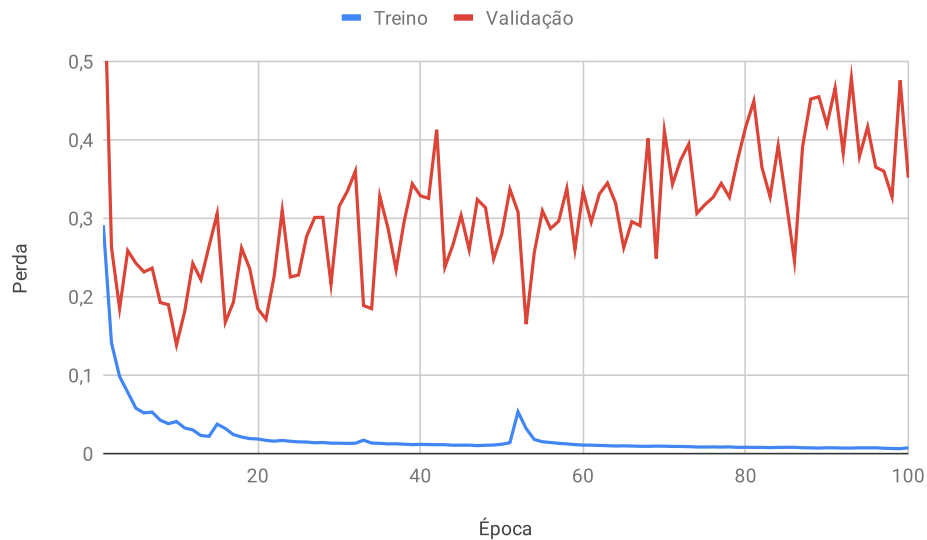


Figura 4.9: Representação gráfica da perda em função das épocas nas fases de treino. Não há convergência

De seguida, foram executadas mais uma séries de configurações diferentes para observar como é que a rede reagia a diferentes combinações de parâmetros. Em primeiro lugar, focou-se nos parâmetros que mais pareciam influenciar os resultados obtidos: a resolução das imagens e o tamanho dos lotes. Para tal, manteve-se os parâmetros definidos na tabela 4.2 mas aumentou-se o número de imagens por lote para 4 e reduziu-se o tamanho das imagens para  $128 \times 128$ . Os parâmetros deste modelo encontram-se definidos na tabela 4.3.

| EP  | NC | NN | DF | TL | DI  | FP  | OT  | TA  | MO  |
|-----|----|----|----|----|-----|-----|-----|-----|-----|
| 100 | 1  | 5  | 3  | 4  | 128 | BCE | SGD | 0.1 | 0.9 |

Tabela 4.3: Parâmetros do treino experimental

Como se pode observar nas figuras 4.10, o aumento do tamanho dos lotes permitiu à rede distinguir, corretamente, a carótida da jugular. Tal deve-se ao facto de que um maior número de imagens por lote permite à rede, a cada iteração, observar mais informação distinta, simultaneamente. É importante ter em consideração, também, que as conexões da rede *U-Net* foram criadas especificamente para melhorar a localização do objeto alvo [RFB15].

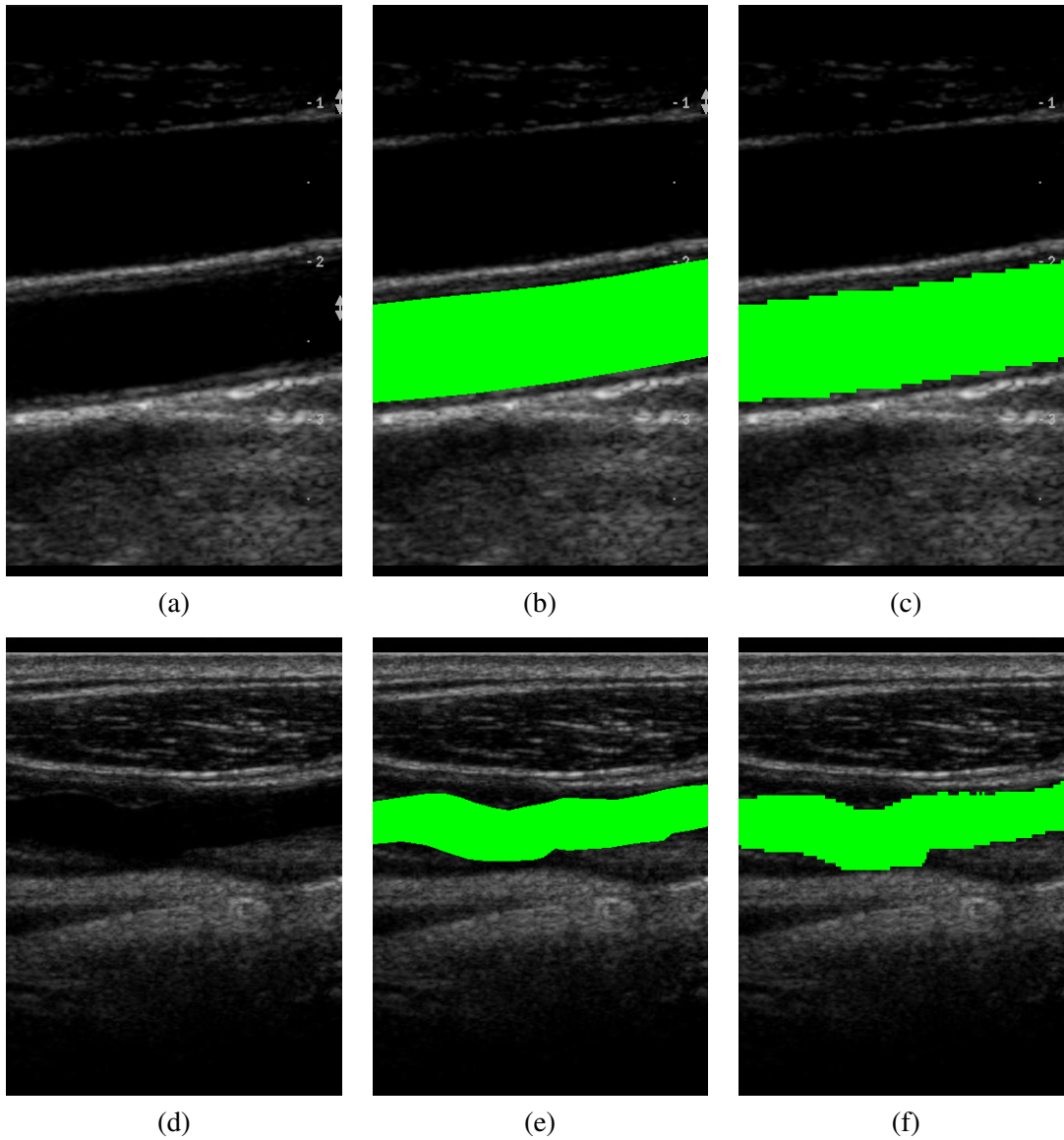


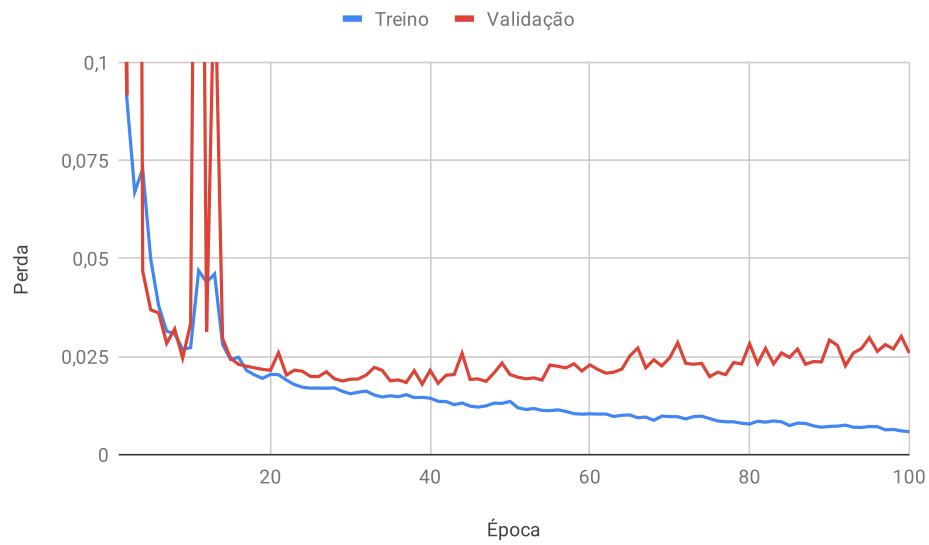
Figura 4.10: Pior e melhor imagem do conjunto de teste. (a) Melhor imagem (b) com máscara de referência e (c) com máscara obtida [DICE = 99.43%]. (d) Pior imagem (e) com máscara de referência e (f) máscara obtida [DICE = 93.30%]

Observa-se, também, que a redução do tamanho das imagens melhorou drasticamente a precisão do modelo (pontuação DICE média de 97.49%). No entanto, o gráfico 4.11 (a) revela que o modelo continua a sofrer de *overfitting*. Como se pode observar no gráfico 4.11 (a), a divergência ocorre a partir de  $\sim 20$  épocas, algo que todos os modelos testados têm em comum. Como tal, de forma a eliminar o efeito de *overfitting*, a taxa de aprendizagem é reduzida por um multiplicador de 0.1 a cada 20 épocas. O efeito é imediato (gráfico 4.11 (b)). A pontuação DICE média mantém-se igualmente alta em 97.24%.

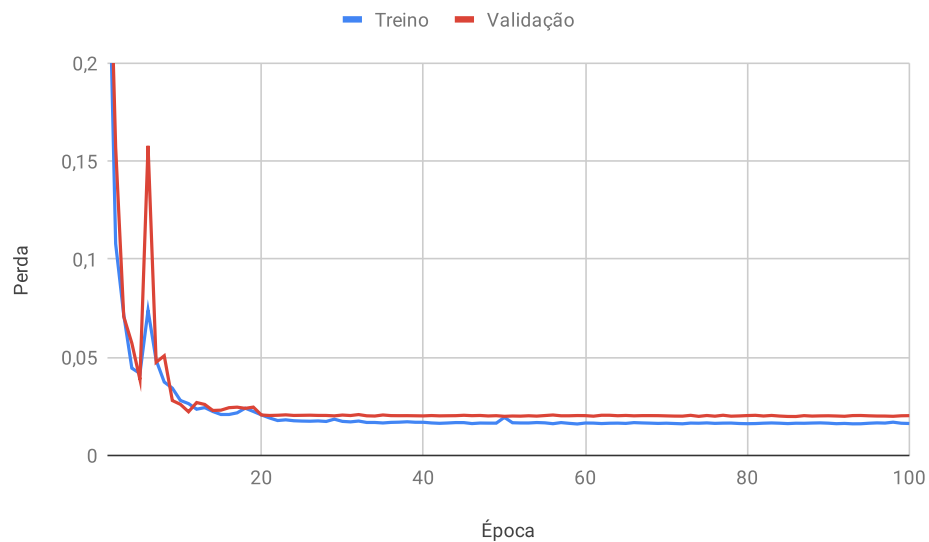
Desta forma, observou-se que as configurações com as dimensões espaciais menores,  $128 \times 128$ , continuavam a ser as que devolviam melhores resultados. Logo, tornou-se óbvio que havia



## Implementação da Rede Neuronal



(a)



(b)

Figura 4.11: Representação gráfica da perda em função das épocas nas fases de treino. (a) Com TA constante não converge e (b) com TA reduzida a cada 20 épocas converge

algum tipo de relação entre a dimensão da imagem e o seu resultado respectivo.

As imagens até agora observadas revelam que a rede aprendeu a localizar e segmentar, corretamente, não só uma zona escura, mas sim uma zona escura delimitada pelas paredes características da carótida, mesmo quando o lúmen contém muito ruído (em caso de imagens de má qualidade) e quando as paredes contém placa. No entanto, imagens de menor resolução, não só devolvem segmentações com linhas muito incrementais (e consequentemente menos exatas), como há uma

redução da margem de erro no cálculo do coeficiente de DICE em relação a imagens de maior resolução. Como tal, sabendo que as camadas de *Max Pooling* reduzem as dimensões das imagens a metade, colocou-se a hipótese de que os filtros das convoluções a cada nível conseguem observar mais em imagens de menor resolução do que em imagens de maior resolução. Logo, um filtro  $3 \times 3$  observa todo, ou quase todo, o lúmen da carótida em imagens de  $8 \times 8$  (figura 4.12), enquanto que em imagens  $512 \times 512$  observa só parte, resultando assim, em segmentações corretas mas incompletas, como se pode observar na figura 4.13 (c).

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Figura 4.12: Exemplo de uma imagem  $8 \times 8$  com um filtro  $3 \times 3$

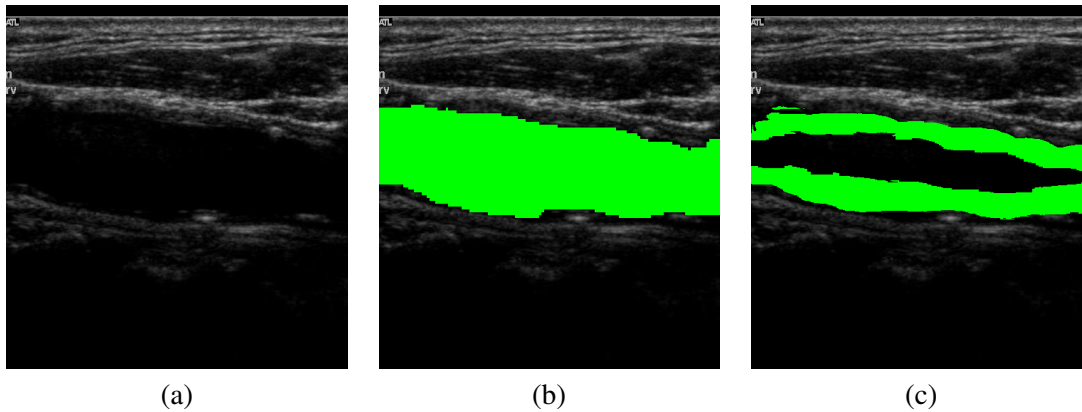


Figura 4.13: (a) Imagem original (b) com segmentação  $128 \times 128$  e (c) com segmentação  $512 \times 512$

Portanto, foram colocadas em hipótese duas soluções:

- **S1.** manter o número de níveis da rede e aumentar a dimensão dos filtros das convoluções permitindo, assim, à rede observar mais em imagens de maior resolução em todos os níveis

da rede;

- **S2.** aumentar o número de níveis da rede e manter a dimensão dos filtros das convoluções fazendo com que os filtros  $3 \times 3$  observem mais em imagens de maior resolução devido a resolução diminuída destas no último nível da rede.

É importante ter em consideração que ambas as soluções aumentam o número de parâmetros da rede e como tal aumentam o consumo de memória, limitando assim, as combinações de configurações possíveis no equipamento disponível. Neste caso as melhores combinações para ambas as soluções podem-se observar na tabela 4.4.

| EP  | NC | NN | DF | TL | DI  | FP         | OT         | TA  | MO  |
|-----|----|----|----|----|-----|------------|------------|-----|-----|
| 100 | 1  | 6  | 3  | 4  | 384 | <i>BCE</i> | <i>SGD</i> | 0.1 | 0.9 |
| 100 | 1  | 5  | 9  | 4  | 480 | <i>BCE</i> | <i>SGD</i> | 0.1 | 0.9 |

Tabela 4.4: Parâmetros das melhores configurações das soluções propostas para obter segmentações de maior resolução

A solução **S2** devolveu melhores resultados do que a **S1** (figuras 4.14). Estas configurações obtiveram uma pontuação média DICE de 97.46% e 95.88%, respetivamente.

As dimensões dos filtros têm um papel crucial na extração de características nos dados que estão a ser observados. Filtros com dimensões muito grandes caíram em desuso [Cha18] devido ao facto de que estes, muitas vezes, não extraem características ou detalhes essenciais para a tarefa a ser resolvida. Por outro lado, filtros com dimensões menores extraem uma quantidade de informação mais vasta e, como tal, mais útil nas camadas consequentes. Portanto, filtros de dimensões menores extraem características locais essenciais sem serem afetados por pixels vizinhos distantes e, como tal, sem relação. Adicionalmente, maior parte das características mais úteis e essenciais de uma imagem são, tipicamente, locais. Logo faz sentido extrair informação de poucos pixels locais ao invés de muitos pixels não relacionados.

Portanto, a alternativa **S1** sofre na sua aprendizagem devido ao facto de que os filtros de maiores dimensões extraem mais informação simultaneamente mas também mais desconexa. Por outro lado, a solução **S2** permite aos filtros extrair mais informação local essencial em imagens de maior resolução devido à profundidade superior da rede. Como tal, pode-se observar nas figuras 4.14, que o aumento do número de níveis da rede permitiu uma segmentação em imagens de maior resolução com uma taxa de sucesso média comparável às segmentações em imagens de menor resolução. Desta forma, obtém-se segmentações menos (visualmente) incrementais, mais precisas e não há uma redução da margem de erro tão significativa como nas imagens de menor resolução.

## 4.5 Configuração Final

Tendo em consideração toda a informação obtida a partir de todos os teste até agora realizados, foi possível executar um último grupo de configurações de forma a descobrir os parâmetros ótimos

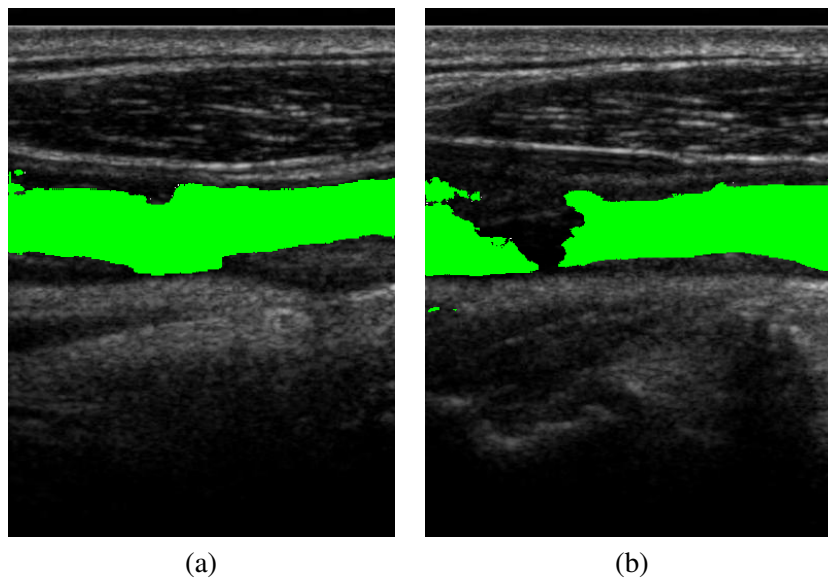


Figura 4.14: Piores imagens com respectivas segmentações das soluções propostas. (a) Com segmentação  $384 \times 384$  (solução **S2**) e (b) com segmentação  $480 \times 480$  (solução **S1**)

da rede. Em primeiro lugar, reduziu-se a resolução das imagens de  $384 \times 384$  para  $320 \times 320$  para se poder aumentar o tamanho dos lotes de 4 para 8 (devido a limitações de memória). Os parâmetros restantes encontram-se definidos na tabela 4.5. Esta configuração obteve uma pontuação DICE média de 97.57%.

| EP  | NC | NN | DF | TL | DI  | FP   | OT  | TA  | MO  |
|-----|----|----|----|----|-----|------|-----|-----|-----|
| 100 | 1  | 6  | 3  | 8  | 320 | DICE | SGD | 0.1 | 0.9 |

Tabela 4.5: Parâmetros da configuração final selecionada

## Capítulo 5

# Resultados e Discussão

Este capítulo apresenta e discute os resultados da solução implementada e compara estes com as segmentações de médicos experientes e com os resultados de um método desenvolvido pelo grupo de investigação C-BER. Adicionalmente, fornece uma descrição breve sobre o método semi-automático desenvolvido pelo grupo de investigação.

### 5.1 Resultados da Solução Implementada

Esta tese propôs e descreveu o desenvolvimento de uma Rede Neuronal Convolucional capaz de segmentar o lúmen da ACC em imagens de ultra-som modo-B com um alto nível de precisão. Os resultados obtidos permitem concluir que a solução proposta é capaz de lidar com o ruído (exemplo na figura 5.2 (g) e (h)), artefactos (exemplo na figura 5.2 (a) e (b)) e variações de contraste características de imagens ultra-som. Demonstrou ser capaz de lidar, também, com outras estruturas semelhantes à carótida presentes nas imagens (exemplo na figura 5.2 (e) e (f)) e com a presença de placa nas paredes (exemplo na figura 5.2 (c) e (d)). Um resumo dos resultados obtidos nas 40 imagens de teste pode ser observado na tabela 5.1.

|              | <b>Média</b> | <b>Mínimo</b> | <b>Máximo</b> | <b>Mediana</b> | <b>Desvio Padrão</b> |
|--------------|--------------|---------------|---------------|----------------|----------------------|
| <b>DICE</b>  | 97.57%       | 94.57%        | 99.18%        | 97.79%         | 0.012%               |
| <b>Perda</b> | 2.47%        | 0.88%         | 5.43%         | 2.25%          | 1.19%                |

Tabela 5.1: Resumo dos resultados obtidos com o método desenvolvido

Os gráficos 5.1 (a) e (b) mostram que o modelo convergiu ao longo do treino. Logo, não sofre de efeitos de *overfitting* ou *underfitting*. Desta maneira, prova-se a capacidade e potencial deste modelo para casos de mundo real. Contudo, o conjunto de imagens inteiro não inclui uma única imagem onde a bifurcação da ACC esteja presente (como exemplificado na figura 2.3 (b)) e, como tal, não há garantias de que este modelo devolveria resultados igualmente bons e robustos (às características previamente descritas) nessas situações (em imagens com bifurcação). Os gráficos

## Resultados e Discussão

5.1 (c) e (d) demonstram os valores de perda e DICE para cada imagem do conjunto de teste. As imagens segmentadas pelo métodos desenvolvido encontram-se todas no anexo B.

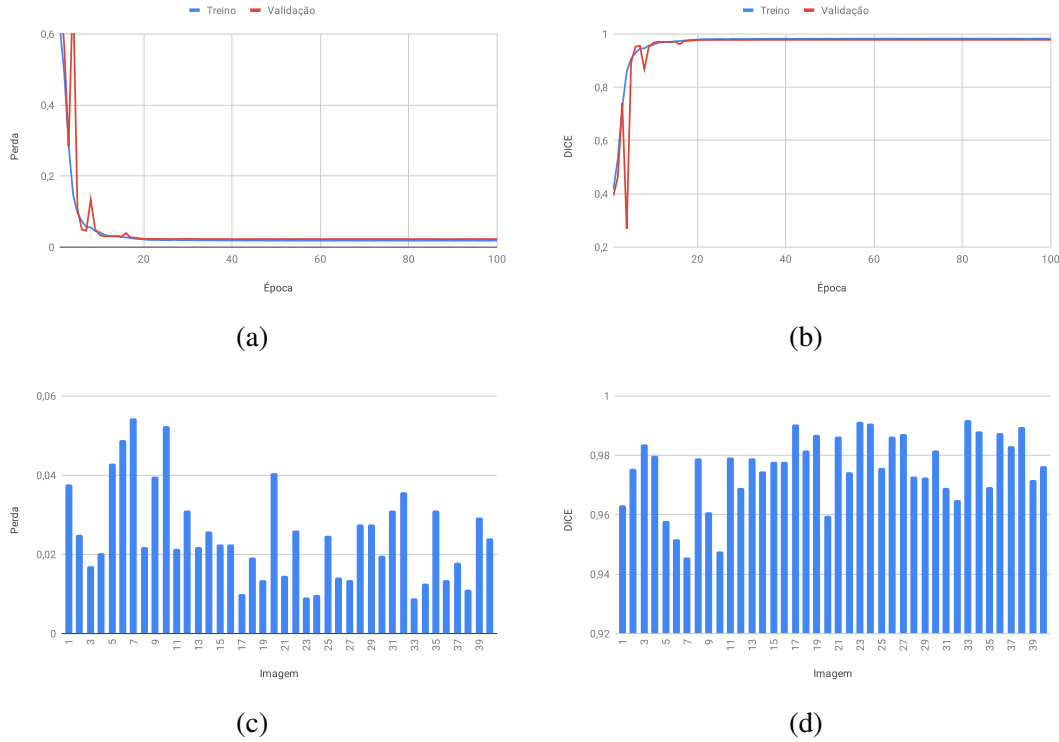


Figura 5.1: Gráficos relacionados com o treino, validação e teste da solução implementada

Tipicamente, as imagens de ultra-som contêm texto nas bordas (como se pode observar na figura 5.2 (c)) que rodeiam o conteúdo principal da imagem, isto é, a imagem adquirida pelos transdutores. De ambos os conjuntos fornecidos para esta tese (referidos na secção 4.2), apenas o conjunto de 150 incluía imagens que continham estas bordas. No entanto, o modelo devolve resultados equiparáveis para imagens com e sem bordas demonstrando, assim, que estas não afetam negativamente a segmentação obtida (reforçando mais uma vez a robustez do modelo).

Finalmente, o acrescento de um nível extra à rede (e conseqüente aumento da profundidade) permitiu-lhe obter segmentações de alta qualidade em imagens de maior resolução e com resultados equiparáveis às de menor resolução (com linhas suaves equiparáveis às de segmentações de maior resolução). Na figura 5.2 (a) e (i) pode-se observar, respetivamente, a melhor e pior imagem do conjunto de teste deste modelo.

### 5.1.1 Comparação Inter-Observador

Como referido anteriormente, o conjunto de 50 imagens tem três segmentações manuais disponíveis: A1, A2 e B1. Dessa 50 imagens 15 fazem parte do conjunto de teste. Como tal, de forma a validar os resultados obtidos pelo método desenvolvido estes foram comparados com os resultados obtidos manualmente. É importante ter em conta que a tabela 5.2 só inclui as pontuações das

## Resultados e Discussão

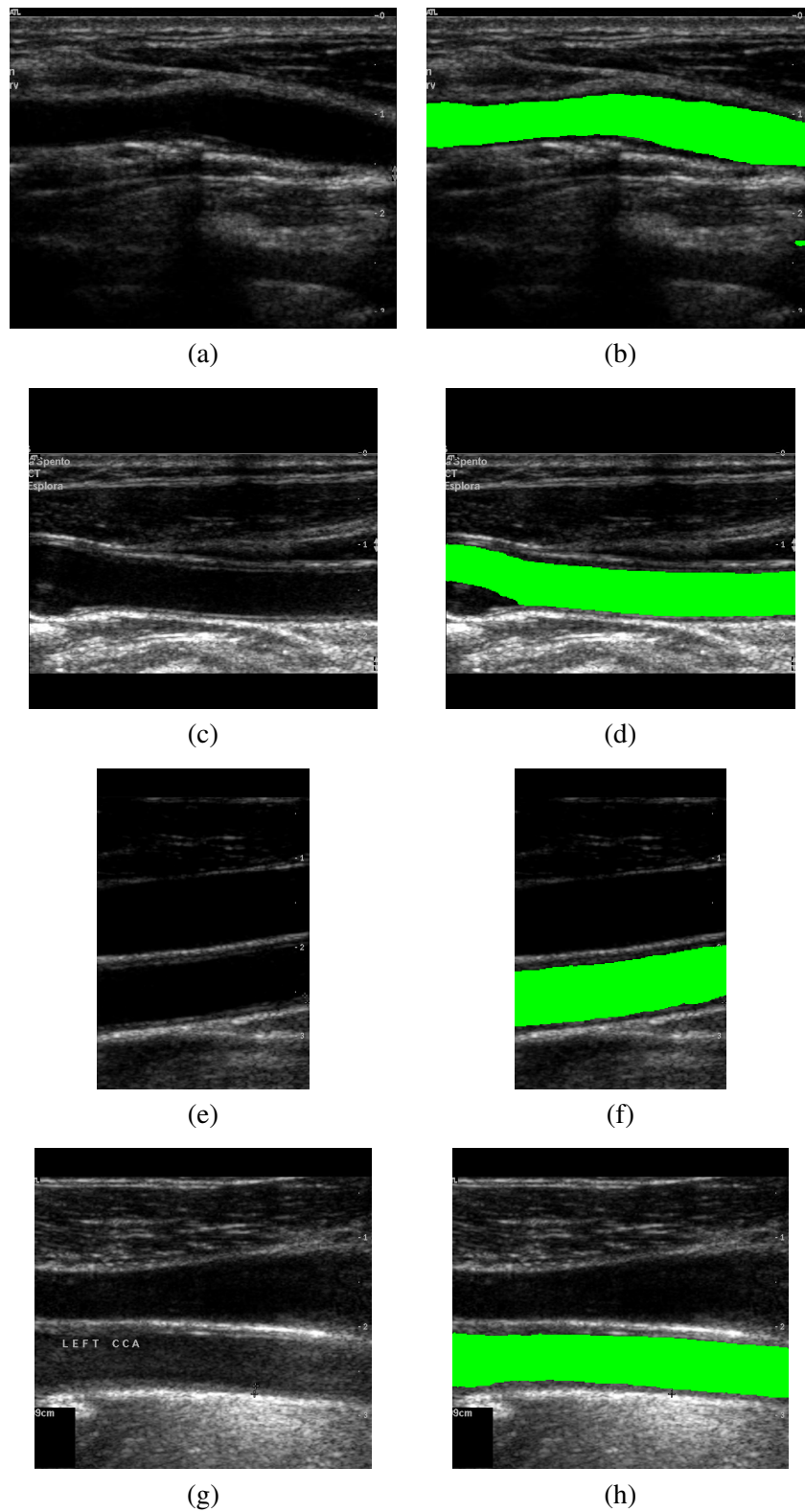


Figura 5.2: (a) Imagem original com artefacto e (b) com segmentação obtida, (c) imagem original com placa e (d) com segmentação obtida, (e) imagem original com VJI e (f) com segmentação obtida (g) imagem original com lúmen ruidoso e (h) segmentação obtida

15 imagens do conjunto de 50 que fazem parte do conjunto de teste. Como se pode observar pela tabela, as pontuações das máscaras obtidas pelo método desenvolvido encontram-se dentro dos intervalos das comparações entre observadores. Logo, pode-se concluir que as máscaras obtidas para um conjunto de imagens desconhecidas (para o modelo) têm o mesmo grau de precisão que as máscaras obtidas por médicos experientes e, como tal, cumprem o mesmo grau de exigência por estes estabelecido. A última linha da tabela apresenta a média das pontuações de cada coluna.

| A1 vs A2 | A1 vs B1 | A2 vs B1 | SP vs A1 | SP vs A2 | SP vs B1 |
|----------|----------|----------|----------|----------|----------|
| 92,41%   | 93,18%   | 98,35%   | 96,16%   | 92,31%   | 92,87%   |
| 98,61%   | 97,42%   | 97,69%   | 97,49%   | 97,63%   | 97,26%   |
| 97,35%   | 96,38%   | 97,57%   | 95,84%   | 94,28%   | 94,76%   |
| 95,36%   | 95,23%   | 96,83%   | 95,17%   | 92,39%   | 93,96%   |
| 98,01%   | 98,21%   | 97,57%   | 94,67%   | 94,01%   | 94,39%   |
| 97,72%   | 98,15%   | 97,90%   | 97,93%   | 96,98%   | 98,14%   |
| 98,73%   | 95,97%   | 98,08%   | 94,83%   | 97,06%   | 97,71%   |
| 97,15%   | 97,85%   | 98,41%   | 98,43%   | 98,19%   | 98,38%   |
| 97,86%   | 97,68%   | 97,92%   | 97,22%   | 96,03%   | 96,34%   |
| 97,97%   | 97,62%   | 98,95%   | 98,28%   | 98,40%   | 98,57%   |
| 98,18%   | 98,67%   | 98,53%   | 98,28%   | 98,39%   | 98,96%   |
| 98,06%   | 98,94%   | 99,06%   | 98,97%   | 98,74%   | 99,28%   |
| 97,32%   | 98,01%   | 97,99%   | 98,36%   | 97,74%   | 98,53%   |
| 97,44%   | 98,29%   | 98,86%   | 98,68%   | 98,38%   | 99,11%   |
| 98,61%   | 99,06%   | 98,68%   | 98,70%   | 98,73%   | 98,62%   |
| 97,39%   | 97,38%   | 98,16%   | 97,27%   | 96,62%   | 97,13%   |

Tabela 5.2: Pontuações DICE das 15 imagens originalmente do conjunto de 50. As primeiras 3 colunas apresentam as comparações inter-observador. As últimas 3 colunas apresentam as comparações das máscaras obtidas pelo método desenvolvido com as segmentações manuais A1, A2 e B1. SP = Solução Proposta

### 5.1.2 Comparação com Método Semi-Automático

De forma a reforçar a capacidade e robustez do método desenvolvido, os resultados deste foram ainda comparados com os resultados de um método semi-automático desenvolvido pelo grupo de investigação C-BER [RCD<sup>+</sup>18]. Um método já testado e validado contra técnicas tipicamente utilizadas para a resolução do mesmo problema.

Em [RCD<sup>+</sup>18], é proposto um método semi-automático para a deteção dos contornos da ACC em imagens de ultra-som modo-B. Um dos principais problemas de deteção de contornos em imagens de ultra-som é a presença de *speckle*. Tipicamente são utilizados filtros Gaussianos para remover os contornos ruidosos, no entanto, estes não são muito eficazes para a remoção de ruído em imagens de tecido homogêneas. Como tal, os autores propõem o uso de derivadas Gaussianas anisotrópicas orientadas (*oriented anisotropic Gaussian derivatives*). A ideia principal é permitir ao filtro integrar informação local da imagem de forma a fornecer uma deteção dos contornos mais robusta, precisa e exata. Adicionalmente, trata-se de um método semi-automático pois permite que



## Resultados e Discussão

os médicos façam alterações as segmentações obtidas caso eles não concordem com estas. Desta forma, os médicos teriam sempre controlo sobre os resultados, impedindo assim segmentações completamente erradas.

Desta forma, o mesmo grupo de teste utilizado pelo método proposto neste tese, foi utilizado pelo método [RCD<sup>+</sup>18] para obter as máscaras correspondentes. Em primeiro lugar, comparou-se estas com as segmentações manuais A1, A2 e B1, demonstrando que este método é também capaz de obter segmentações com o mesmo grau de precisão e exigência das de médicos experientes. Os resultados podem-se observar na tabela 5.3.

| A1 vs A2 | A1 vs B1 | A2 vs B1 | SA vs A1 | SA vs A2 | SA vs B1 |
|----------|----------|----------|----------|----------|----------|
| 92,41%   | 93,18%   | 98,35%   | 92,47%   | 92,66%   | 93,45%   |
| 98,61%   | 97,42%   | 97,69%   | 96,18%   | 96,32%   | 95,12%   |
| 97,35%   | 96,38%   | 97,57%   | 96,22%   | 94,08%   | 93,81%   |
| 95,36%   | 95,23%   | 96,83%   | 91,84%   | 88,41%   | 88,24%   |
| 98,01%   | 98,21%   | 97,57%   | 96,73%   | 95,58%   | 96,23%   |
| 97,72%   | 98,15%   | 97,90%   | 98,72%   | 97,90%   | 98,08%   |
| 98,73%   | 95,97%   | 98,08%   | 89,94%   | 88,90%   | 92,47%   |
| 97,15%   | 97,85%   | 98,41%   | 97,90%   | 98,07%   | 97,99%   |
| 97,86%   | 97,68%   | 97,92%   | 97,40%   | 97,59%   | 97,45%   |
| 97,97%   | 97,62%   | 98,95%   | 98,30%   | 98,60%   | 98,69%   |
| 98,18%   | 98,67%   | 98,53%   | 98,87%   | 98,33%   | 99,02%   |
| 98,06%   | 98,94%   | 99,06%   | 98,82%   | 98,44%   | 99,05%   |
| 97,32%   | 98,01%   | 97,99%   | 94,38%   | 93,27%   | 94,32%   |
| 97,44%   | 98,29%   | 98,86%   | 98,14%   | 98,00%   | 98,43%   |
| 98,61%   | 99,06%   | 98,68%   | 95,05%   | 94,09%   | 95,12%   |
| 97,39%   | 97,38%   | 98,16%   | 96,06%   | 95,35%   | 95,83%   |

Tabela 5.3: Pontuações DICE das 15 imagens originalmente do conjunto de 50. As primeiras 3 colunas apresentam as comparações inter-observador. As últimas 3 colunas apresentam as comparações das máscaras obtidas pelo método [RCD<sup>+</sup>18] com as segmentações manuais A1, A2 e B1. SA = Método Semi-Automático [RCD<sup>+</sup>18]

De seguida, comparou-se as máscaras obtidas pelo método desenvolvido e as máscaras obtidas por [RCD<sup>+</sup>18] com as máscaras de referência utilizadas para testar e validar o método desenvolvido. Os resultados estão apresentados na tabela 5.4. É importante ter em consideração que o método [RCD<sup>+</sup>18] não conseguiu segmentar e segmentou incorretamente 2 e 8 imagens das 40, respetivamente. Nas 8 imagens que segmentou incorretamente, e que consequentemente obteve uma pontuação de 0%, deveu-se ao facto de ter segmentado a VJI. No entanto, sabendo que se trata de um método semi-automático, não seria justo ter essas imagens em consideração na comparação. De qualquer maneira, a tabela 5.4 apresenta a média das pontuações com (valor entre parênteses) e sem as máscaras incorretas.

Desta forma, pode-se concluir que o método proposto, não só é mais robusto, mas também obtém, em média, melhores pontuações que [RCD<sup>+</sup>18]. Sabendo que [RCD<sup>+</sup>18] foi comparado contra métodos que aplicam técnicas tradicionais de deteção e segmentação em imagens de ultra-

## Resultados e Discussão

| SP vs GT | [RCD <sup>+</sup> 18] vs GT |
|----------|-----------------------------|
| 96,32%   | 92,73%                      |
| 97,55%   | 96,95%                      |
| 95,78%   | 94,02%                      |
| 95,15%   | 91,84%                      |
| 94,58%   | 96,56%                      |
| 97,88%   | 97,39%                      |
| 94,73%   | 89,86%                      |
| 97,84%   | 96,83%                      |
| 96,86%   | 96,73%                      |
| 97,83%   | 95,73%                      |
| 97,74%   | 97,82%                      |
| 99,00%   | 98,84%                      |
| 98,05%   | 94,68%                      |
| 98,71%   | 98,16%                      |
| 95,88%   | 97,88%                      |
| 98,26%   | 97,61%                      |
| 97,93%   | 96,69%                      |
| 97,79%   | 94,09%                      |
| 98,50%   | 95,82%                      |
| 97,24%   | 96,99%                      |
| 99,06%   | 0,00%                       |
| 98,98%   | 0,00%                       |
| 97,52%   | 94,82%                      |
| 98,61%   | 95,85%                      |
| 98,59%   | 95,10%                      |
| 97,26%   | 0,00%                       |
| 97,14%   | 0,00%                       |
| 98,22%   | 0,00%                       |
| 96,87%   | 91,08%                      |
| 96,51%   | 97,23%                      |
| 99,11%   | 98,81%                      |
| 98,74%   | 96,27%                      |
| 96,86%   | 0,00%                       |
| 98,73%   | 95,98%                      |
| 98,30%   | 86,73%                      |
| 98,87%   | 0,00%                       |
| 97,19%   | 0,00%                       |
| 97,62%   | 97,15%                      |
| 96,17%   | X                           |
| 97,45%   | X                           |
| 97,54%   | 95,54% (75,43%)             |

Tabela 5.4: Pontuações DICE das 40 imagens do conjunto de teste. *GT = Ground Truth*

som modo-B, pode-se assumir que o método proposto é, também, mais robusto, preciso e exato do que essas outras técnicas tradicionais.

Na figura 5.3 pode-se observar as imagens, apresentadas previamente na figura 5.2, segmentadas pelo método [RCD<sup>+</sup>18]. Demonstram que este método é também capaz de lidar (em maior parte dos casos) com a presença de artefactos de imagem (figura 5.3 (a)), com a presença de placa (figura 5.3 (b)) e com as variações de contraste típicas de imagens de ultra-som. No entanto, demonstra também dificuldades em imagens que contenham a VJI (figura 5.3 (c)) e ruído extremo (figura 5.3 (d)).

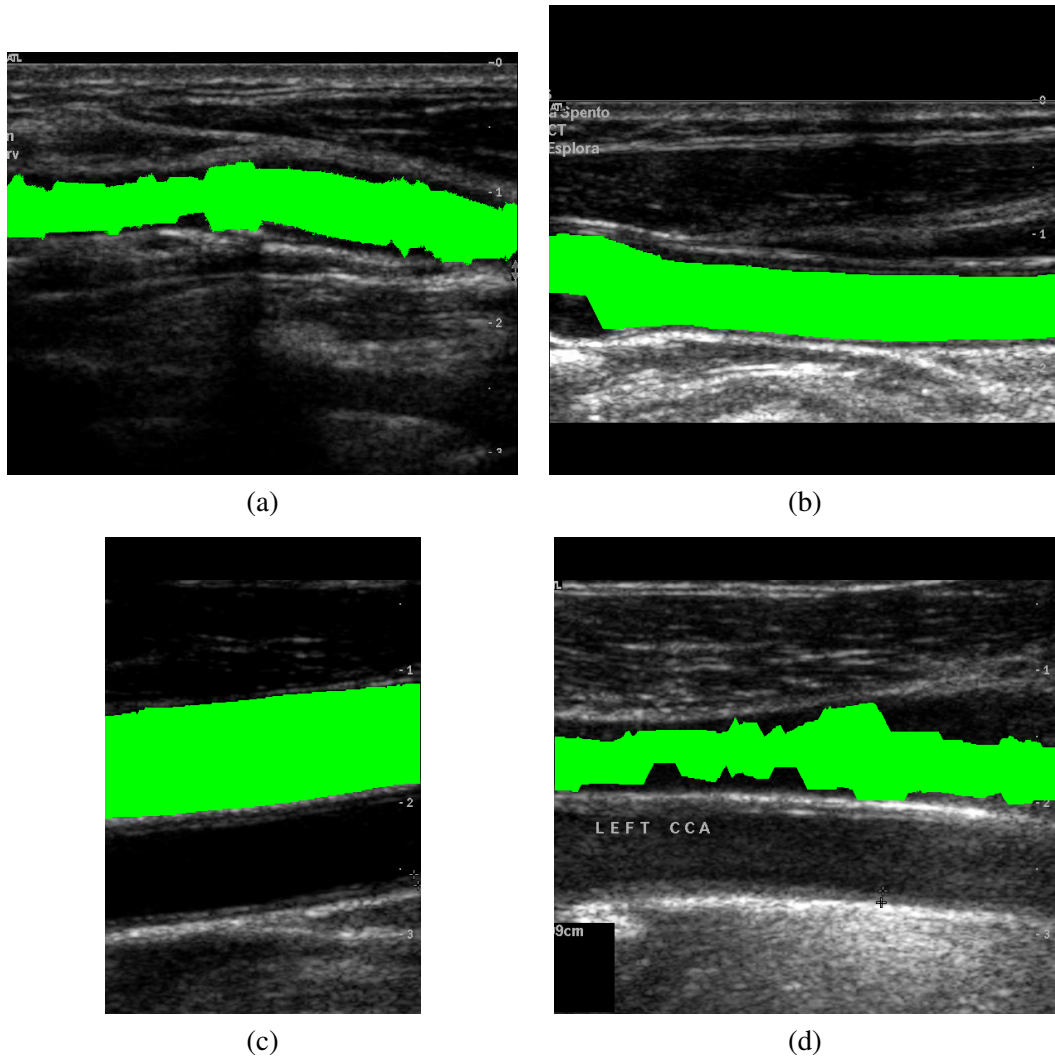


Figura 5.3: Imagens originais 5.2 (a), (c), (e) e (g) com segmentações obtidas pelo método [RCD<sup>+</sup>18]

A figura 5.4 apresenta as melhores e piores imagens obtidas pelo método proposto e pelo método [RCD<sup>+</sup>18] em comparação com as máscaras de referência. Adicionalmente, nestas é possível observar que o método [RCD<sup>+</sup>18], ao contrário do método proposto, tem dificuldades em segmentar o lúmen em imagens que contenham texto no mesmo. Admitidamente, esta é a única imagem (no conjunto de teste inteiro) com tal característica, logo não é possível concluir ou assumir que este problema escale com o número de imagens.

## Resultados e Discussão

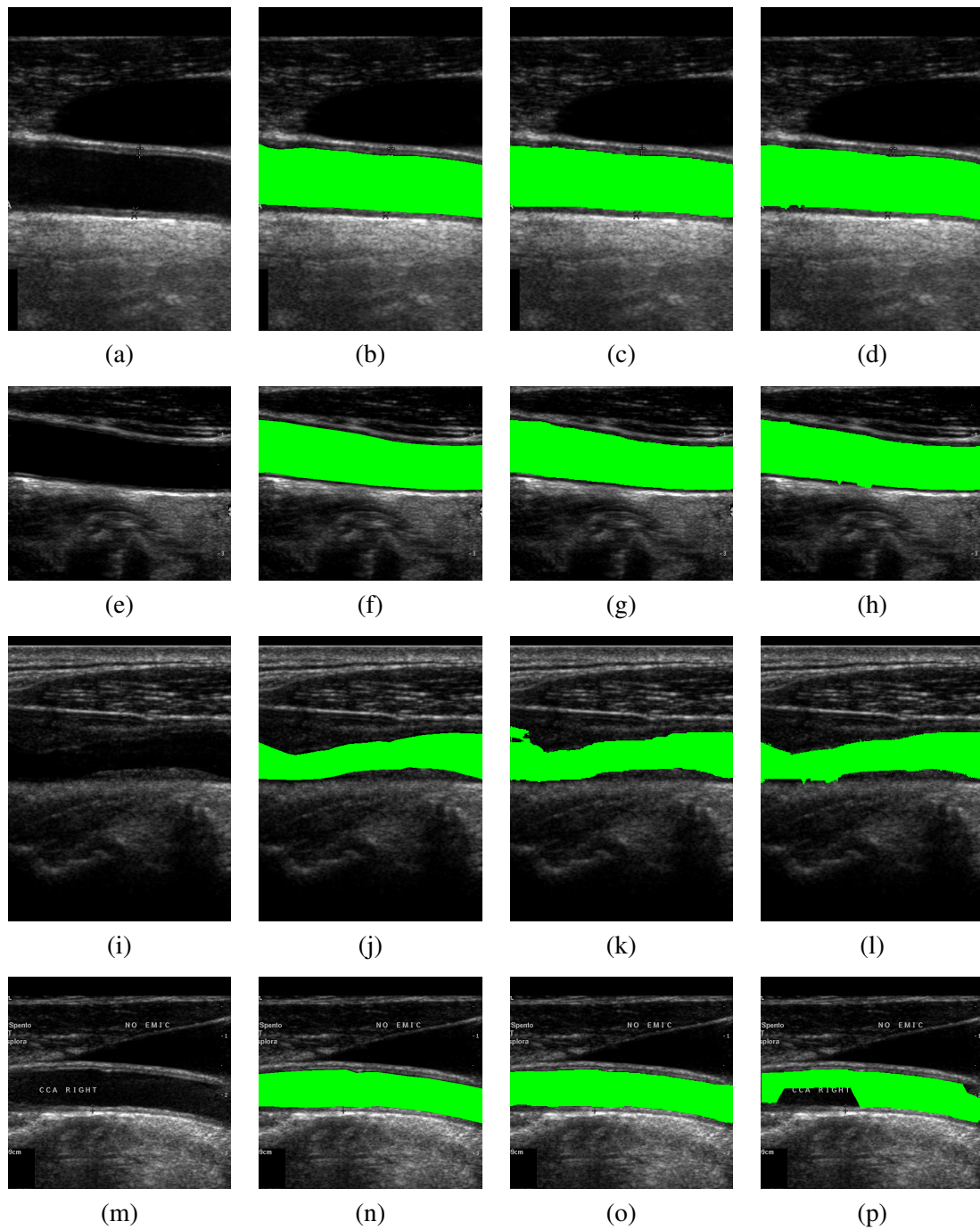


Figura 5.4: Piores e melhores imagens do conjunto de teste do método proposto e [RCD<sup>+</sup>18]. (a) Melhor imagem do método proposto com (b) máscara de referência, (c) máscara obtida pelo método proposto e (d) máscara obtida por [RCD<sup>+</sup>18]. (e) Melhor imagem do método [RCD<sup>+</sup>18] com (f) máscara de referência, (g) máscara obtida pelo método proposto e (h) máscara obtida por [RCD<sup>+</sup>18]. (i) Pior imagem do método proposto com (j) máscara de referência, (k) máscara obtida pelo método proposto e (l) máscara obtida por [RCD<sup>+</sup>18]. (m) Pior imagem do método [RCD<sup>+</sup>18] com (n) máscara de referência, (o) máscara obtida pelo método proposto e (p) máscara obtida por [RCD<sup>+</sup>18].

## Capítulo 6

# Conclusões e Trabalho Futuro

Este capítulo fornece uma conclusão geral sobre a satisfação dos objetivos estipulados e sobre possíveis expansões do trabalho desenvolvido.

### 6.1 Satisfação dos Objetivos

Foram definidos três objetivos principais na secção 1.2:

- **O1.** Demonstrou-se que é possível detetar e segmentar a ACC em imagens de ultra-som modo-B com o uso de uma Rede Neuronal Convolucional, mais especificamente, uma variação da *U-Net*, e que se pode obter resultados equiparáveis, e por vezes melhores, aos de técnicas tradicionais tipicamente utilizadas para a resolução do problema;
- **O2.** Testou-se os resultados obtidos num conjunto de imagens desconhecidos (para o modelo) e, como tal, validou-se a capacidade e potencial deste modelo para casos de uso do mundo real;
- **O3.** Comparou-se os resultados obtidos com os de uma técnica semi-automática desenvolvida pelo grupo de investigação de forma a demonstrar, não só o potencial de Redes Neurais em geral, mas também, mais uma vez, a capacidade e potencial do método desenvolvido em comparação com técnicas já testadas e aprovadas para a resolução do problema.

### 6.2 Trabalho Futuro

Este projeto foi encarado de forma a ser resolvido com uma abordagem puramente à base de Redes Neurais, no entanto, isso não significa que se deva evitar a aplicação de outros métodos antes ou depois do processamento dos dados pelo modelo treinado, de forma a melhorar os resultados ou a robustez do método em casos de uso do mundo real.

**Maior Conjunto de Imagens** Qualquer modelo treinado, independentemente da sua arquitetura e objetivo, beneficia sempre do maior conjunto de dados distintos que conseguir obter. No caso deste projeto, o mesmo continua a ser verdade, apesar do facto de que a *U-Net* foi criada de forma a conseguir aprender com conjuntos pequenos, não há dúvidas de que o modelo beneficiaria de um maior conjunto de dados de forma a aumentar a sua robustez quando exposto a casos do mundo real, onde poderá encontrar uma variação muito grande de imagens visto que estas dependem e variam muito de acordo com o equipamento e utilizador.

**Paragem Antecipada *Early Stopping* [Pre98]** Paragem Antecipada refere-se a uma técnica de regularização usada para evitar que os modelos sofram de *overfitting* quando estes são treinados com um método iterativo, como por exemplo, o Gradiente Descendente. Algoritmos de Paragem Antecipada monitorizam e controlam quantas iterações de treino podem ser executadas antes de o modelo sofrer de *overfitting*. Atualmente, o modelo desenvolvido não sofre deste problema, mas no caso de se ter acesso a um maior conjunto de imagens, algoritmos de Paragem Antecipada são uma alternativa a ter em consideração para uma monitorização do treino mais confiante e robusta.

**Pré/Pós-Processamento** Como referido anteriormente, este projeto aborda a solução do problema com um método puramente de Redes Neurais. Em casos não limitados pela mesma restrição, deve-se considerar o uso de técnicas e/ou filtros que, por exemplo, suavizam as imagens de ultra-som ou removem *speckle* e artefactos. Em termos de pós-processamento, poderia se utilizar alguma técnica de remoção de *blobs* para remover objetos indesejados da máscara obtida e, consequentemente, aumentar o grau de semelhança entre esta e a máscara de referência respetiva.

**VggNet** Uma solução alternativa a aumentar o número de níveis da rede passa por substituir o caminho de “contração” (ou *downsampling*) por uma *VGG-16* e adaptar o caminho de “expansão” (ou *upsampling*) apropriadamente. Trata-se de um método que já foi testado e validado na resolução de problemas de segmentação em imagens de ultra-som modo-B [SBIN17], [BDS18].

**Interpolação das Máscaras** Como referido anteriormente, o redimensionamento das máscaras é feito através de uma interpolação *Nearest Neighbor* que tem a vantagem referida de garantir que as máscaras continuam a ser imagens binárias independentemente das novas dimensões. No entanto, tem a desvantagem de devolver imagens com linhas muito (visualmente) incrementais onde estavam linhas com curvas mais suaves e naturais. Portanto, uma alternativa passa por aplicar uma interpolação diferente às máscaras, muito provavelmente *Cubic Spline* (como nas imagens originais), seguido de um *threshold* de 0.5, garantindo assim imagens binárias sem a desvantagem referida.

# Referências

- [BB] Ayush Goel Balint Botz. Ultrasound artifacts. URL: <https://radiopaedia.org/articles/ultrasound-artifacts-3>.
- [BDS18] Chirag Balakrishna, Sarshar Dadashzadeh e Sara Soltaninejad. Automatic detection of lumen and media in the ivus images using u-net with vgg16 encoder. *arXiv preprint arXiv:1806.07554*, 2018.
- [Bha18] Anup Bhande. What is underfitting and overfitting in machine learning and how to deal with it., 2018. URL: <https://medium.com/greyatom/what-is-underfitting-and-overfitting-in-machine-learning-and-how-to->
- [BP13] Vaijinath V Bhosle e Vrushsen P Pawar. Texture segmentation: different methods. *International Journal of Soft Computing and Engineering (IJSCE)*, 3(5):69–74, 2013.
- [BSC<sup>+</sup>17] Lilla Bonanno, Fabrizio Sottile, Rosella Ciurleo, Giuseppe Di Lorenzo, Daniele Bruschetta, Alessia Bramanti, Giorgio Ascenti, Placido Bramanti e Silvia Marino. Automatic algorithm for segmentation of atherosclerotic carotid plaque. *Journal of Stroke and Cerebrovascular Diseases*, 26(2):411–416, 2017.
- [Cam] Cambridge in Colour. Digital Image Interpolation. URL: <https://www.cambridgeincolour.com/tutorials/image-interpolation.htm>.
- [CCR12] Siqi Chen, Daniel Cremers e Richard J Radke. Image segmentation with one shape prior—a template-based formulation. *Image and Vision Computing*, 30(12):1032–1042, 2012.
- [Cha18] Arnault Chazareix. About Convolutional Layer and Convolution Kernel, 2018. URL: <https://blog.sicara.com/about-convolutional-layer-convolution-kernel-9a7325d34f7d>.
- [CUFa] Instituto CUF. Aterosclerose. URL: <https://www.saudecuf.pt/mais-saude/doencas-a-z/aterosclerose>.
- [CUFb] Instituto CUF. AVC – Acidente Vascular Cerebral. URL: <https://www.saudecuf.pt/mais-saude/doencas-a-z/avc-acidente-vascular-cerebral>.
- [Das17] Siddharth Das. CNN Architectures: LeNet, AlexNet, VGG, GoogLeNet, ResNet and more ...., 2017. URL: <https://medium.com/{\spacefactor\@{}}sidereal/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-66609>



## REFERÊNCIAS

- [DDB11] M Debakla, Khalifa Djemal e Mohamed Benyettou. Influence of noise distribution on active contour models: Medical images segmentation. *Asian Journal of Applied Sciences*, 4(2):101–111, 2011.
- [Der17] Arden Dertat. Applied Deep Learning - Part 4: Convolutional Neural Networks, 2017. URL: <https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2>.
- [Des16] Adit Deshpande. A Beginner’s Guide To Understanding Convolutional Neural Networks – Adit Deshpande – CS Undergrad at UCLA (’19), 2016. URL: <https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/>  
<https://adeshpande3.github.io/adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/>.
- [Die17] Reinhard Diestel. *Graph theory*, volume 173. Springer-Verlag, © Reinhard Diestel, 2017.
- [dS] Leonardo Araujo dos Santos. Dropout Layer. URL: [https://leonardoaraujosantos.gitbooks.io/artificial-intelligence/content/dropout\\_layer.html](https://leonardoaraujosantos.gitbooks.io/artificial-intelligence/content/dropout_layer.html).
- [FRC09] Sylvain Fischer, Rafael Redondo e Gabriel Cristóbal. How to construct log-gabor filters? 2009.
- [GB10] Xavier Glorot e Yoshua Bengio. Understanding the difficulty of training deep feed-forward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.
- [GDDM14] Ross Girshick, Jeff Donahue, Trevor Darrell e Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014.
- [GFRO16] Wilfrido Gómez-Flores e Bedert Abel Ruiz-Ortega. New fully automated method for segmentation of breast lesions on ultrasound based on texture analysis. *Ultrasound in Medicine and Biology*, 42(7):1637–1650, 2016.
- [Gir15] Ross Girshick. Fast r-cnn. *arXiv preprint arXiv:1504.08083*, 2015.
- [GSP05] Nikhil Gupta, MNS Swamy e Eugene Plotkin. Despeckling of medical ultrasound images using data and rate adaptive lossy compression. *IEEE Transactions on Medical Imaging*, 24(6):743–754, 2005.
- [Hay94] Simon Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 1994.
- [HCRG<sup>+</sup>16] Shin Hoo-Chang, Holger R Roth, Mingchen Gao, Le Lu, Ziyue Xu, Isabella Nogueiras, Jianhua Yao, Daniel Mollura e Ronald M Summers. Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5):1285, 2016.



## REFERÊNCIAS

- [HMT10] Peter R Hoskins, Kevin Martin e Abigail Thrush. *Diagnostic ultrasound: physics and equipment*. Cambridge University Press, 2010.
- [HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren e Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.
- [JMH<sup>+</sup>16] Rui Jia, Stephen J Mellon, S Hansjee, AP Monk, David W Murray e J Alison Noble. Automatic bone segmentation in ultrasound images using local phase features and dynamic programming. In *Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on*, pages 1005–1008. IEEE, 2016.
- [Ken18] Will Kenton. Pareto Principle, 2018. URL: <https://www.investopedia.com/terms/p/paretoprinciple.asp>.
- [KS17] Nitish Shirish Keskar e Richard Socher. Improving generalization performance by switching from adam to SGD. *arXiv preprint arXiv:1712.07628*, 2017.
- [KSH12] Alex Krizhevsky, Ilya Sutskever e Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.
- [Lan13] Brett Lantz. *Machine learning with R*. Packt Publishing Ltd, 2013.
- [LAS13] Joice LS Lima, Alexandra André e Antônio C Santos. Reprodução e estudo de artefatos no ultrassom. *Revista Brasileira de Física Médica*, 7(3):205–208, 2013.
- [LGB<sup>+</sup>17] Karim Lekadir, Alfia Galimzianova, Àngels Betriu, Maria del Mar Vila, Laura Igual, Daniel L Rubin, Elvira Fernández, Petia Radeva e Sandy Napel. A convolutional neural network for automatic characterization of plaque composition in carotid ultrasound. *IEEE J. Biomedical and Health Informatics*, 21(1):48–55, 2017.
- [Li16] You Li. Segmentation of Medical Ultrasound Images Using Convolutional Neural Networks with Noisy Activating Functions. *Stanford Edu*, 2016. URL: <http://cs229.stanford.edu/proj2016/report/Li-SegmentationOfMedicalUltrasoundImagesUsingConvolutionalNeuralNetwork.pdf>.
- [LKB<sup>+</sup>17] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen AWM Van Der Laak, Bram Van Ginneken e Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017.
- [LPP<sup>+</sup>07] Christos P Loizou, Constantinos S Pattichis, Marios Pantziaris, T Tyllis e Andrew Nicolaides. Snakes based segmentation of the common carotid artery intima media. *Medical & Biological Engineering & Computing*, 45(1):35–49, 2007.
- [LSD15] Jonathan Long, Evan Shelhamer e Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.
- [M<sup>+</sup>97] Tom M Mitchell et al. Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, 45(37):870–877, 1997.

## REFERÊNCIAS

- [MAM17] Kristen M Meiburger, U Rajendra Acharya e Filippo Molinari. Automated localization and segmentation techniques for b-mode ultrasound images: A review. *Computers in Biology and Medicine*, 2017.
- [NHH15] Hyeonwoo Noh, Seunghoon Hong e Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1520–1528, 2015.
- [NHPY16] Bo Ni, Fa-zhi He, Yi-teng Pan e Zhi-yong Yuan. Using shapes correlation for active contour segmentation of uterine fibroid ultrasound images in computer-aided therapy. *Applied Mathematics-A Journal of Chinese Universities*, 31(1):37–52, 2016.
- [NU] National Heart Lung and Blood Institute e U.S. Department of Health & Human Services. Atherosclerosis | National Heart, Lung, and Blood Institute (NHLBI). URL: <https://www.nhlbi.nih.gov/health-topics/atherosclerosis>.
- [Ope] OpenCV Team. OpenCV. URL: <https://opencv.org/>.
- [Pre98] Lutz Prechelt. Early stopping-but when? In *Neural Networks: Tricks of the trade*, pages 55–69. Springer, 1998.
- [Pyta] Python Software Foundation. Python. URL: <https://www.python.org/>.
- [PyTb] PyTorch Team. PyTorch. URL: <https://pytorch.org/>.
- [RAC16] José Rouco, Elsa Azevedo e Aurélio Campilho. Automatic lumen detection on longitudinal ultrasound b-mode images of the carotid using phase symmetry. *Sensors*, 16(3):350, 2016.
- [RCD<sup>+</sup>18] José Rouco, Catarina Carvalho, Ana Domingues, Elsa Azevedo e Aurélio Campilho. A robust anisotropic edge detection method for carotid ultrasound image processing. *Procedia Computer Science*, 126:723–732, 2018.
- [RFB15] Olaf Ronneberger, Philipp Fischer e Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, pages 234–241. Springer, 2015.
- [RHGS15] Shaoqing Ren, Kaiming He, Ross Girshick e Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.
- [RL17] Carol M Rumack e Deborah Levine. *Diagnostic Ultrasound E-Book*. Elsevier Health Sciences, 2017.
- [RSC11] Rui Rocha, Jorge Alves Silva e Aurélio JC Campilho. Dynamic programming and fuzzy classification for the automatic segmentation of the carotid in ultrasound images. In *MVA*, pages 552–556, 2011.
- [RSC14] Rui Rocha, Jorge Silva e Aurélio Campilho. Automatic detection of the carotid lumen axis in b-mode ultrasound images. *Computer methods and programs in biomedicine*, 115(3):110–118, 2014.

## REFERÊNCIAS

- [SBIN17] Vaanathi Sundaresan, Christopher P Bridge, Christos Ioannou e J Alison Noble. Automated characterization of the fetal heart in ultrasound images using fully convolutional neural networks. In *Biomedical Imaging (ISBI 2017), 2017 IEEE 14th International Symposium on*, pages 671–674. IEEE, 2017.
- [SDK<sup>+</sup>18] Reuben Ruby Shamir, Yuval Duchin, Jinyoung Kim, Guillermo Sapiro e Noam Harel. Continuous dice coefficient: a method for evaluating probabilistic segmentations. *bioRxiv*, page 306977, 2018.
- [Sha] Shaoanlu. Which One Is The Best Optimizer | SALu. URL: <https://shaoanlu.wordpress.com/2017/05/29/sgd-all-which-one-is-the-best-optimizer-dogs-vs-cats-toy-experiment/>
- [SL16] Erik Smistad e Lasse Løvstakken. Vessel detection in ultrasound images using deep convolutional neural networks. In *Deep Learning and Data Labeling for Medical Applications*, pages 30–38. Springer, 2016.
- [SLJ<sup>+</sup>15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke e Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
- [Staa] Stanford. CS231n Convolutional Neural Networks for Visual Recognition. URL: <http://cs231n.github.io/neural-networks-3/{#}ada>.
- [Stab] Stanford. Optimization: Stochastic Gradient Descent. URL: <http://deeplearning.stanford.edu/tutorial/supervised/OptimizationStochasticGradientDescent/>.
- [SZ14] Karen Simonyan e Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [Ten] TensorFlow Team. TensorBoard: Visualizing Learning | TensorFlow. URL: [https://www.tensorflow.org/guide/summaries{\\_\)and{\\_\)tensorboard](https://www.tensorflow.org/guide/summaries{_)and{_)tensorboard).
- [TSG<sup>+</sup>16] Nima Tajbakhsh, Jae Y Shin, Suryakanth R Gurudu, R Todd Hurst, Christopher B Kendall, Michael B Gotway e Jianming Liang. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE transactions on medical imaging*, 35(5):1299–1312, 2016.
- [UVDSGS13] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers e Arnold WM Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, 2013.
- [V. 99] Michael V. Boland. Haralick texture features, 1999. URL: [http://murphylab.web.cmu.edu/publications/boland/boland{\\_\)node26.html](http://murphylab.web.cmu.edu/publications/boland/boland{_)node26.html).
- [Wan17] Diego S Wanderley Student - FEUP. Convolutional Neural Networks: Its Application to Ultrasound Image Analysis in Gynecology. Technical report, FEUP, 2017.
- [Wik] Wikipedia. Interpolation. URL: <https://en.wikipedia.org/wiki/Interpolation>.

## REFERÊNCIAS

- [WLGW97] Inger Wendelhag, Quan Liang, Tomas Gustavsson e John Wikstrand. A new automated computerized analyzing system simplifies readings and reduces the variability in ultrasound measurement of intima-media thickness. *Stroke*, 28(11):2195–2200, 1997.
- [WLJ<sup>+</sup>15] Weining Wang, Jiachang Li, Yizi Jiang, Yi Xing e Xiangmin Xu. An automatic energy-based region growing method for ultrasound image segmentation. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 1553–1557. IEEE, 2015.
- [WRS<sup>+</sup>17] Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro e Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. In *Advances in Neural Information Processing Systems*, pages 4148–4158, 2017.
- [WS92] Donna J Williams e Mubarak Shah. A fast algorithm for active contours and curvature estimation. *CVGIP: Image Understanding*, 55(1):14–26, 1992.
- [WS11] Jackie L Whittaker e Maria Stokes. Ultrasound imaging and muscle function. *Journal of Orthopaedic & Sports Physical Therapy*, 41(8):572–580, 2011.
- [ZSM<sup>+</sup>18] Vasily Zyuzin, Porshnev Sergey, Andrey Mukhtarov, Tatyana Chumarnaya, Olga Solovyova, Anastasia Bobkova e Vladislav Myasnikov. Identification of the left ventricle endocardial border on two-dimensional ultrasound images using the convolutional neural network unet. In *2018 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBREIT)*, pages 76–78. IEEE, 2018.

## Anexo A

# Configurações de Treino

Este anexo apresenta uma tabela com todas as configurações executadas ao longo do desenvolvimento deste projeto.

| EP | NC | NN | DF | TL | DI  | FP   | OT   | TA  | MO  |
|----|----|----|----|----|-----|------|------|-----|-----|
| 50 | 1  | 5  | 3  | 16 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 256 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 256 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1  | 5  | 3  | 16 | 256 | DICE | Adam | 0.1 | 0.9 |

### Configurações de Treino

|    |   |   |   |    |     |      |      |     |     |
|----|---|---|---|----|-----|------|------|-----|-----|
| 50 | 1 | 5 | 3 | 16 | 256 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 16 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 16 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 16 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 16 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 16 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 16 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 384 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 2  | 576 | BICE | SGD  | 0.1 | 0.9 |

# Configurações de Treino

|    |   |   |   |   |     |      |      |     |     |
|----|---|---|---|---|-----|------|------|-----|-----|
| 50 | 1 | 5 | 3 | 4 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | BICE | SGD  | 0.1 | 0.9 |

### Configurações de Treino

|    |   |   |   |   |     |      |      |     |     |
|----|---|---|---|---|-----|------|------|-----|-----|
| 50 | 1 | 5 | 3 | 4 | 384 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 384 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |



### Configurações de Treino

|    |   |   |   |   |     |      |      |     |     |
|----|---|---|---|---|-----|------|------|-----|-----|
| 50 | 1 | 5 | 3 | 8 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 3 | 8 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | BICE | SGD  | 0.1 | 0.9 |

### Configurações de Treino

|    |   |   |   |   |     |      |      |     |     |
|----|---|---|---|---|-----|------|------|-----|-----|
| 50 | 1 | 5 | 5 | 2 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 2 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 5 | 4 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 2 | 480 | MSE  | SGD  | 0.1 | 0.9 |

### Configurações de Treino

|    |   |   |   |   |     |      |      |     |     |
|----|---|---|---|---|-----|------|------|-----|-----|
| 50 | 1 | 5 | 7 | 4 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 7 | 4 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 2 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | BICE | SGD  | 0.1 | 0.9 |

### Configurações de Treino

|    |   |   |   |   |     |      |      |     |     |
|----|---|---|---|---|-----|------|------|-----|-----|
| 50 | 1 | 5 | 9 | 4 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 5 | 9 | 4 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 384 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2 | 480 | MSE  | SGD  | 0.1 | 0.9 |

### Configurações de Treino

|    |   |   |   |    |     |      |      |     |     |
|----|---|---|---|----|-----|------|------|-----|-----|
| 50 | 1 | 6 | 3 | 2  | 512 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 2  | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | BCE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | BCE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | BICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | BICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | DICE | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | DICE | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | MSE  | Adam | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 1 | 6 | 3 | 4  | 384 | MSE  | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 16 | 128 | CE   | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 16 | 128 | CE   | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 16 | 128 | CE   | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 16 | 128 | CE   | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 16 | 256 | CE   | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 16 | 256 | CE   | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 16 | 256 | CE   | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 16 | 256 | CE   | SGD  | 0.1 | 0.9 |

### Configurações de Treino

|    |   |   |   |   |     |    |      |     |     |
|----|---|---|---|---|-----|----|------|-----|-----|
| 50 | 2 | 5 | 3 | 2 | 384 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 2 | 384 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 2 | 384 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 2 | 384 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 2 | 480 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 2 | 480 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 2 | 480 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 2 | 480 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 128 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 128 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 128 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 128 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 256 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 256 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 256 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 256 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 384 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 384 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 384 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 384 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 480 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 480 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 480 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 480 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 512 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 512 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 512 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 4 | 512 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 8 | 128 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 8 | 128 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 8 | 128 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 8 | 128 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 8 | 256 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 8 | 256 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 8 | 256 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 3 | 8 | 256 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 5 | 2 | 480 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 5 | 2 | 480 | CE | Adam | 0.1 | 0.9 |
| 50 | 2 | 5 | 5 | 2 | 480 | CE | SGD  | 0.1 | 0.9 |
| 50 | 2 | 5 | 5 | 2 | 480 | CE | SGD  | 0.1 | 0.9 |

# Configurações de Treino

|     |   |   |    |   |     |     |      |     |     |
|-----|---|---|----|---|-----|-----|------|-----|-----|
| 50  | 2 | 5 | 5  | 4 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 5 | 5  | 4 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 5 | 5  | 4 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 5 | 5  | 4 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 5 | 7  | 2 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 5 | 7  | 2 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 5 | 7  | 2 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 5 | 7  | 2 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 5 | 7  | 4 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 5 | 7  | 4 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 5 | 7  | 4 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 5 | 7  | 4 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 5 | 9  | 2 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 5 | 9  | 2 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 5 | 9  | 2 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 5 | 9  | 2 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 5 | 9  | 4 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 5 | 9  | 4 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 5 | 9  | 4 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 5 | 9  | 4 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 384 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 384 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 384 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 384 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 480 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 480 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 512 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 512 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 512 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 2 | 512 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 4 | 384 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 4 | 384 | CE  | Adam | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 4 | 384 | CE  | SGD  | 0.1 | 0.9 |
| 50  | 2 | 6 | 3  | 4 | 384 | CE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4 | 512 | BCE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4 | 512 | BCE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4 | 512 | BCE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4 | 512 | BCE | SGD  | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |    |    |     |      |      |     |     |
|-----|---|---|----|----|-----|------|------|-----|-----|
| 100 | 1 | 5 | 11 | 4  | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 4  | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 11 | 8  | 352 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3  | 16 | 128 | DICE | SGD  | 0.1 | 0.9 |



### Configurações de Treino

|     |   |   |   |    |     |      |      |     |     |
|-----|---|---|---|----|-----|------|------|-----|-----|
| 100 | 1 | 5 | 3 | 16 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 16 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 256 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 256 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4  | 256 | BCE  | SGD  | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |   |   |     |      |      |     |     |
|-----|---|---|---|---|-----|------|------|-----|-----|
| 100 | 1 | 5 | 3 | 4 | 256 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |   |   |     |      |      |     |     |
|-----|---|---|---|---|-----|------|------|-----|-----|
| 100 | 1 | 5 | 3 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |   |   |     |      |      |     |     |
|-----|---|---|---|---|-----|------|------|-----|-----|
| 100 | 1 | 5 | 3 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 3 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | DICE | Adam | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |   |   |     |      |      |     |     |
|-----|---|---|---|---|-----|------|------|-----|-----|
| 100 | 1 | 5 | 5 | 8 | 352 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 5 | 8 | 352 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 7 | 8 | 352 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |   |   |     |      |      |     |     |
|-----|---|---|---|---|-----|------|------|-----|-----|
| 100 | 1 | 5 | 9 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | DICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | DICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | MSE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 5 | 9 | 8 | 352 | MSE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 6 | 3 | 4 | 384 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 6 | 3 | 4 | 384 | BCE  | Adam | 0.1 | 0.9 |
| 100 | 1 | 6 | 3 | 4 | 384 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 6 | 3 | 4 | 384 | BCE  | SGD  | 0.1 | 0.9 |
| 100 | 1 | 6 | 3 | 4 | 384 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 6 | 3 | 4 | 384 | BICE | Adam | 0.1 | 0.9 |
| 100 | 1 | 6 | 3 | 4 | 384 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 6 | 3 | 4 | 384 | BICE | SGD  | 0.1 | 0.9 |
| 100 | 1 | 6 | 3 | 4 | 384 | DICE | Adam | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |   |   |     |      |      |        |      |
|-----|---|---|---|---|-----|------|------|--------|------|
| 100 | 1 | 6 | 3 | 4 | 384 | DICE | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | DICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | DICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | MSE  | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | MSE  | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | MSE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | MSE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | BCE  | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | BCE  | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | BCE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | BCE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | BICE | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | BICE | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | BICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | BICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | DICE | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | DICE | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | DICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | DICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | MSE  | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | MSE  | Adam | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | MSE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 4 | 384 | MSE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 0.0001 | 0.99 |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 0.0001 | 0.9  |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 0.001  | 0.99 |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 0.001  | 0.9  |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 0.01   | 0.99 |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 0.01   | 0.9  |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 0.1    | 0.99 |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 1.0    | 0.99 |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 1.0    | 0.9  |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 1e-05  | 0.99 |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 1e-05  | 0.9  |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 1e-06  | 0.99 |
| 100 | 1 | 6 | 3 | 8 | 320 | BCE  | SGD  | 1e-06  | 0.9  |
| 100 | 1 | 6 | 3 | 8 | 320 | BICE | SGD  | 0.0001 | 0.99 |
| 100 | 1 | 6 | 3 | 8 | 320 | BICE | SGD  | 0.0001 | 0.9  |

### Configurações de Treino

|     |   |   |    |    |     |      |      |        |      |
|-----|---|---|----|----|-----|------|------|--------|------|
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 0.001  | 0.99 |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 0.001  | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 0.01   | 0.99 |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 0.01   | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 0.1    | 0.99 |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 1.0    | 0.99 |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 1.0    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 1e-05  | 0.99 |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 1e-05  | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 1e-06  | 0.99 |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 1e-06  | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | DICE | SGD  | 0.0001 | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | DICE | SGD  | 0.001  | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | DICE | SGD  | 0.01   | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | DICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | DICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | DICE | SGD  | 1.0    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | DICE | SGD  | 1e-05  | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | DICE | SGD  | 1e-06  | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | MSE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | MSE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | BCE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | BCE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | BICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | DICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | DICE | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | MSE  | SGD  | 0.1    | 0.9  |
| 100 | 1 | 6 | 3  | 8  | 320 | MSE  | SGD  | 0.1    | 0.9  |
| 100 | 2 | 5 | 11 | 4  | 512 | CE   | Adam | 0.1    | 0.9  |
| 100 | 2 | 5 | 11 | 4  | 512 | CE   | Adam | 0.1    | 0.9  |
| 100 | 2 | 5 | 11 | 4  | 512 | CE   | SGD  | 0.1    | 0.9  |
| 100 | 2 | 5 | 11 | 4  | 512 | CE   | SGD  | 0.1    | 0.9  |
| 100 | 2 | 5 | 11 | 8  | 352 | CE   | Adam | 0.1    | 0.9  |
| 100 | 2 | 5 | 11 | 8  | 352 | CE   | Adam | 0.1    | 0.9  |
| 100 | 2 | 5 | 11 | 8  | 352 | CE   | SGD  | 0.1    | 0.9  |
| 100 | 2 | 5 | 11 | 8  | 352 | CE   | SGD  | 0.1    | 0.9  |
| 100 | 2 | 5 | 3  | 16 | 128 | CE   | Adam | 0.1    | 0.9  |



# Configurações de Treino

|     |   |   |   |    |     |    |      |     |     |
|-----|---|---|---|----|-----|----|------|-----|-----|
| 100 | 2 | 5 | 3 | 16 | 128 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 16 | 128 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 16 | 128 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 16 | 256 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 16 | 256 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 16 | 256 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 16 | 256 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 128 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 128 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 128 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 128 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 256 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 256 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 256 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 256 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 512 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 512 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 512 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 512 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 512 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 512 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 512 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 4  | 512 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 128 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 128 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 128 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 128 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 256 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 256 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 256 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 256 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 352 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 352 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 352 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 3 | 8  | 352 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 5 | 4  | 512 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 5 | 4  | 512 | CE | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 5 | 4  | 512 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 5 | 4  | 512 | CE | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 5 | 8  | 352 | CE | Adam | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |   |    |     |      |      |     |     |
|-----|---|---|---|----|-----|------|------|-----|-----|
| 100 | 2 | 5 | 5 | 8  | 352 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 5 | 8  | 352 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 5 | 8  | 352 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 7 | 4  | 512 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 7 | 4  | 512 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 7 | 4  | 512 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 7 | 4  | 512 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 7 | 8  | 352 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 7 | 8  | 352 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 7 | 8  | 352 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 7 | 8  | 352 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 9 | 4  | 512 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 9 | 4  | 512 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 9 | 4  | 512 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 9 | 4  | 512 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 9 | 8  | 352 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 9 | 8  | 352 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 5 | 9 | 8  | 352 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 5 | 9 | 8  | 352 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 4  | 384 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 4  | 384 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 4  | 384 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 4  | 384 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 4  | 384 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 4  | 384 | CE   | Adam | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 4  | 384 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 4  | 384 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 8  | 320 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 8  | 320 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 8  | 320 | CE   | SGD  | 0.1 | 0.9 |
| 100 | 2 | 6 | 3 | 8  | 320 | CE   | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | DICE | Adam | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |   |    |     |      |      |     |     |
|-----|---|---|---|----|-----|------|------|-----|-----|
| 150 | 1 | 5 | 3 | 16 | 128 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 16 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4  | 256 | BCE  | Adam | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |   |   |     |      |      |     |     |
|-----|---|---|---|---|-----|------|------|-----|-----|
| 150 | 1 | 5 | 3 | 4 | 256 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 4 | 512 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8 | 128 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8 | 128 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8 | 128 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8 | 128 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8 | 128 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8 | 128 | DICE | Adam | 0.1 | 0.9 |

### Configurações de Treino

|     |   |   |   |    |     |      |      |     |     |
|-----|---|---|---|----|-----|------|------|-----|-----|
| 150 | 1 | 5 | 3 | 8  | 128 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 128 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 128 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 128 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 128 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 128 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | BCE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | BCE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | BICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | BICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | DICE | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | DICE | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | MSE  | Adam | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 1 | 5 | 3 | 8  | 256 | MSE  | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 16 | 128 | CE   | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 16 | 128 | CE   | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 16 | 128 | CE   | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 16 | 128 | CE   | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 16 | 256 | CE   | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 16 | 256 | CE   | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 16 | 256 | CE   | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 16 | 256 | CE   | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4  | 128 | CE   | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4  | 128 | CE   | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4  | 128 | CE   | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4  | 128 | CE   | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4  | 256 | CE   | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4  | 256 | CE   | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4  | 256 | CE   | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4  | 256 | CE   | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4  | 512 | CE   | Adam | 0.1 | 0.9 |

### Configurações de Treino

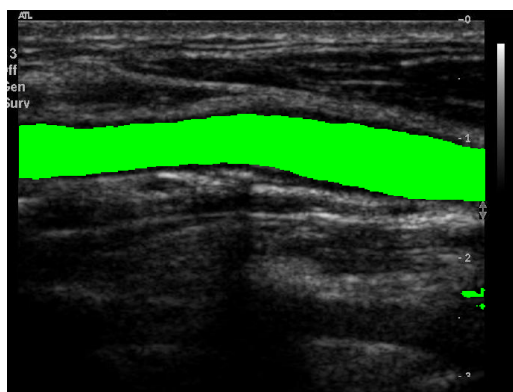
|     |   |   |   |   |     |    |      |     |     |
|-----|---|---|---|---|-----|----|------|-----|-----|
| 150 | 2 | 5 | 3 | 4 | 512 | CE | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4 | 512 | CE | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 4 | 512 | CE | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 8 | 128 | CE | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 8 | 128 | CE | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 8 | 128 | CE | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 8 | 128 | CE | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 8 | 256 | CE | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 8 | 256 | CE | Adam | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 8 | 256 | CE | SGD  | 0.1 | 0.9 |
| 150 | 2 | 5 | 3 | 8 | 256 | CE | SGD  | 0.1 | 0.9 |

Tabela A.1: Tabela de todas as configurações de treino executadas

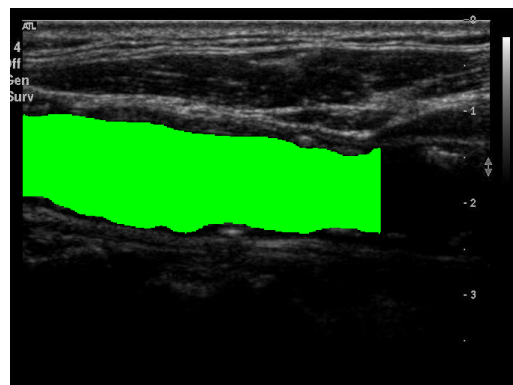
## Anexo B

### Imagens de Teste

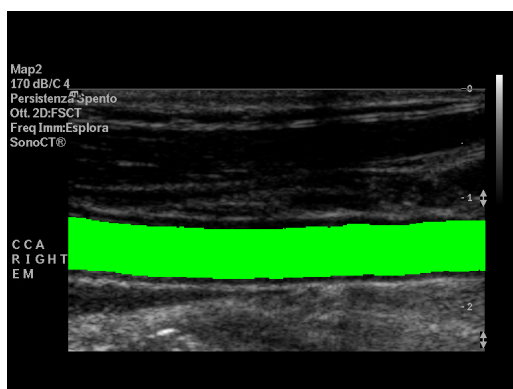
Neste anexo apresentam-se todas as imagens do conjunto de teste com as segmentações respectivas, obtidas a partir do método desenvolvido.



DICE = 96.32%



DICE = 97.54%

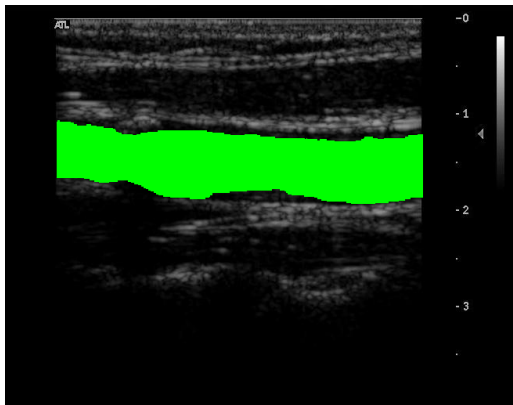


DICE = 98.37%

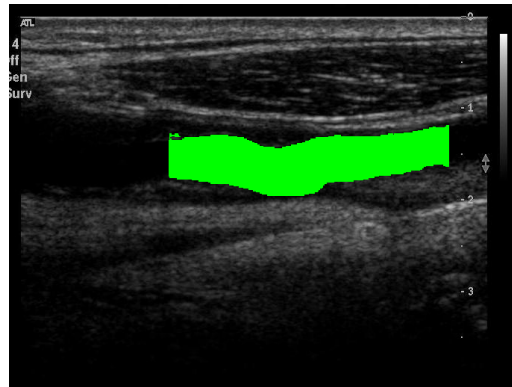


DICE = 97.99%

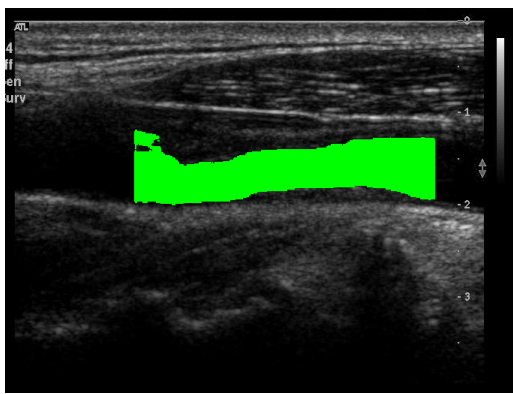
## Imagens de Teste



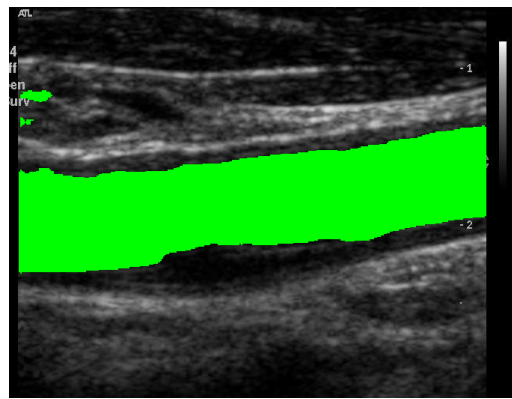
DICE = 95.78%



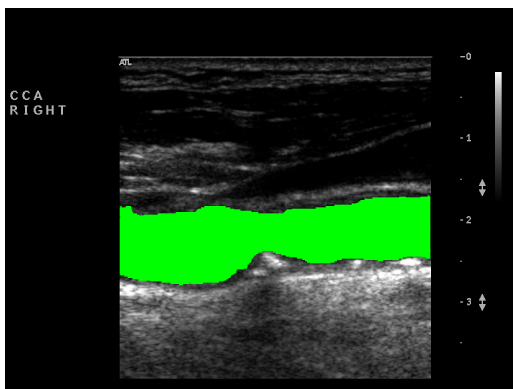
DICE = 95.19%



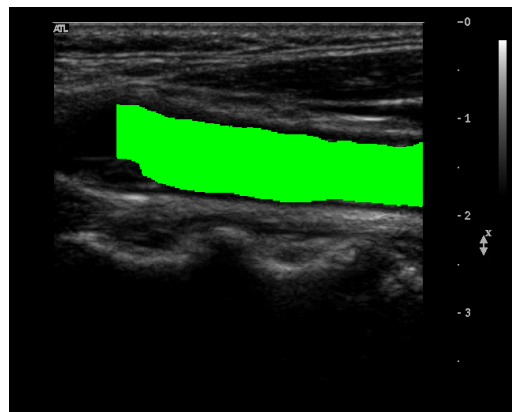
DICE = 94.57%



DICE = 97.89%



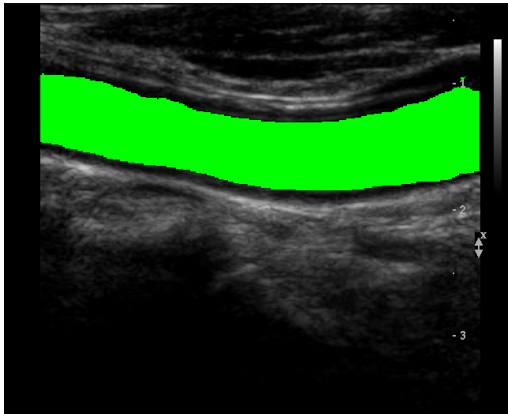
DICE = 96.08%



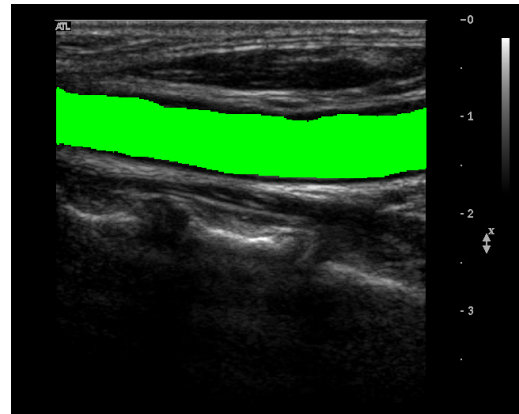
DICE = 94.76%



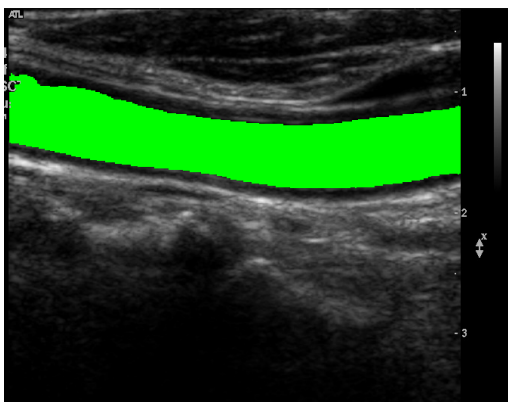
## Imagens de Teste



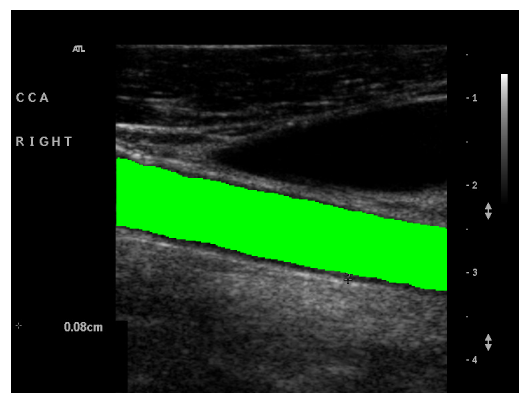
DICE = 97.92%



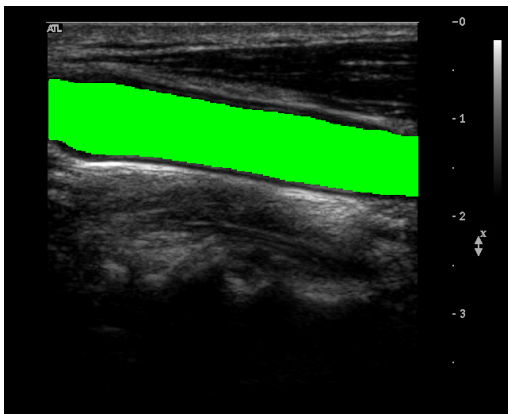
DICE = 96.89%



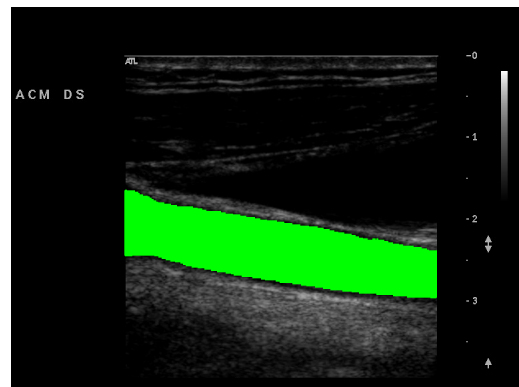
DICE = 97.88%



DICE = 97.47%

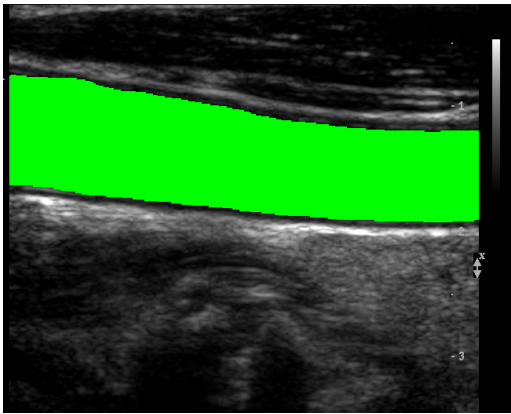


DICE = 97.79%

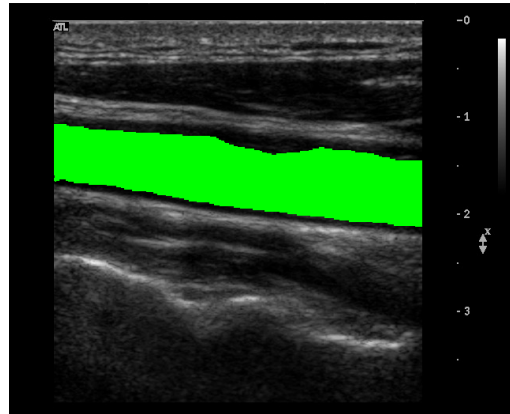


DICE = 97.79%

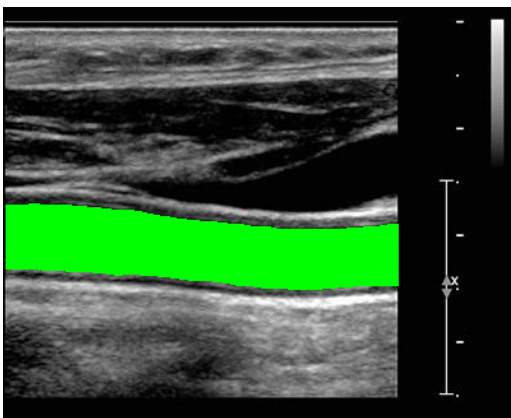
# Imagens de Teste



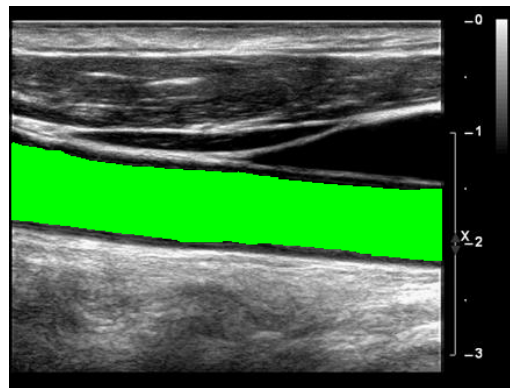
DICE = 99.03%



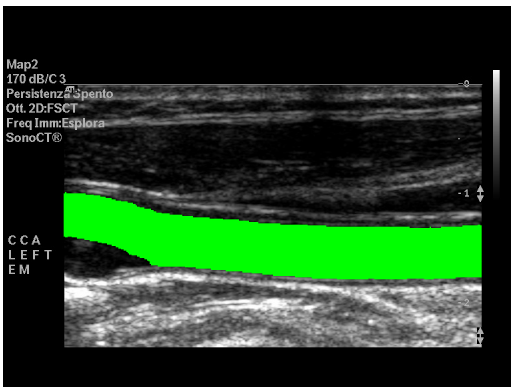
DICE = 98.17%



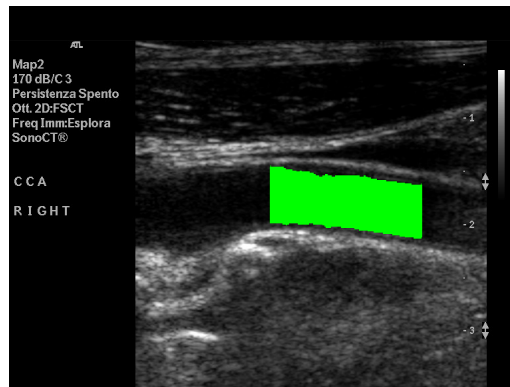
DICE = 98.67%



DICE = 95.97%

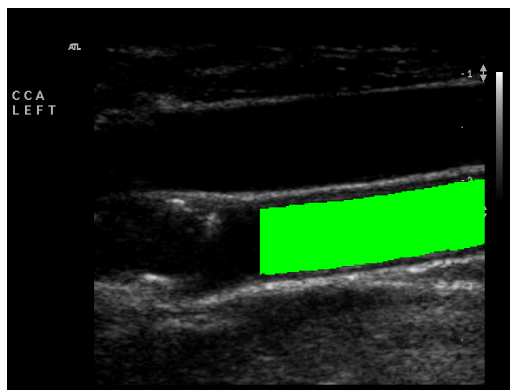


DICE = 98.62%

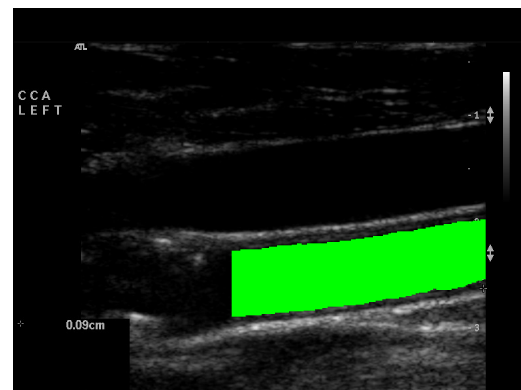


DICE = 97.43%

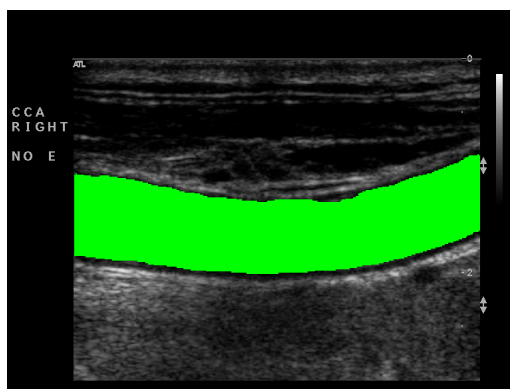
## Imagens de Teste



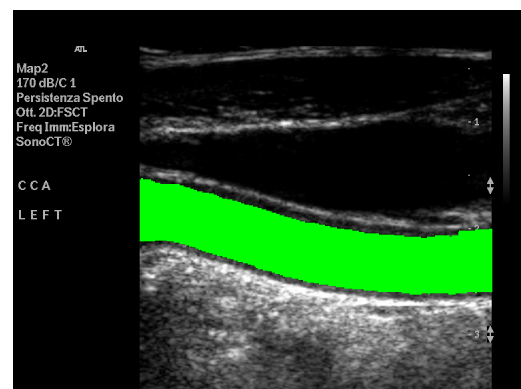
DICE = 99.13%



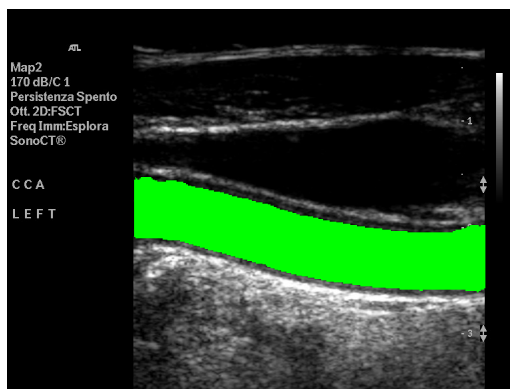
DICE = 99.07%



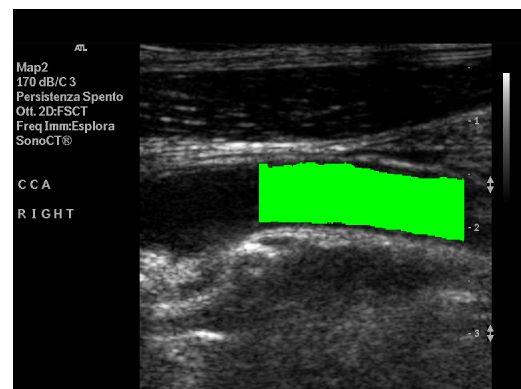
DICE = 97.58%



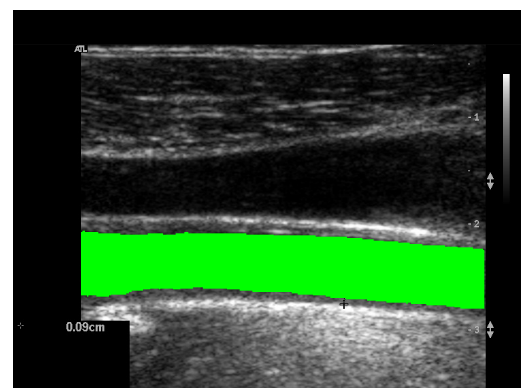
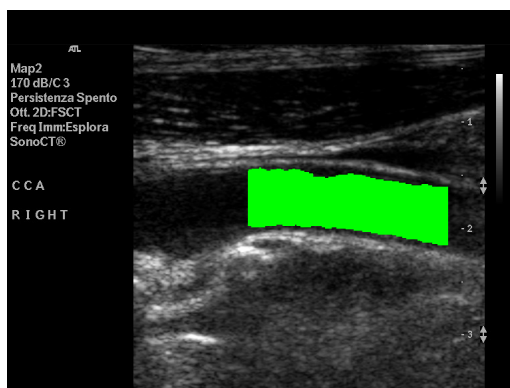
DICE = 98.61%



DICE = 98.72%

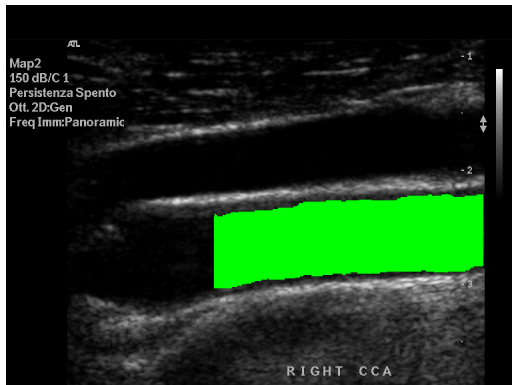


DICE = 97.28%

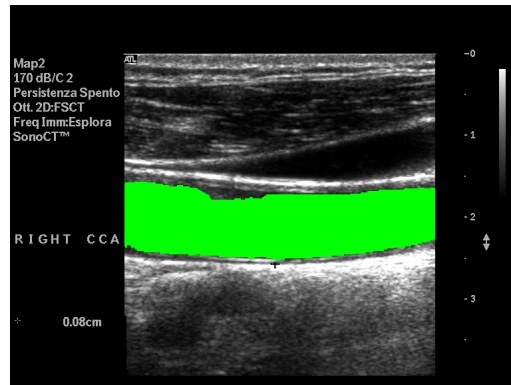


## Imagens de Teste

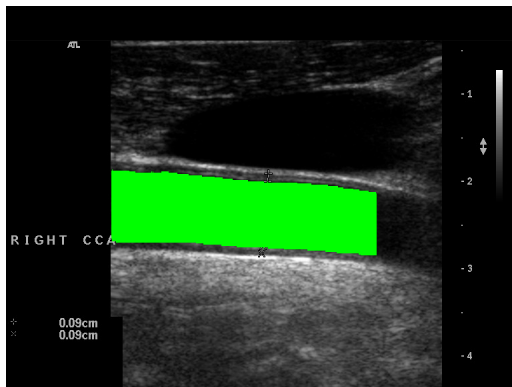
DICE = 97.24%



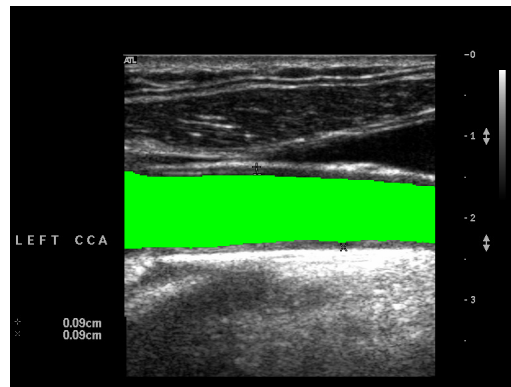
DICE = 98.17%



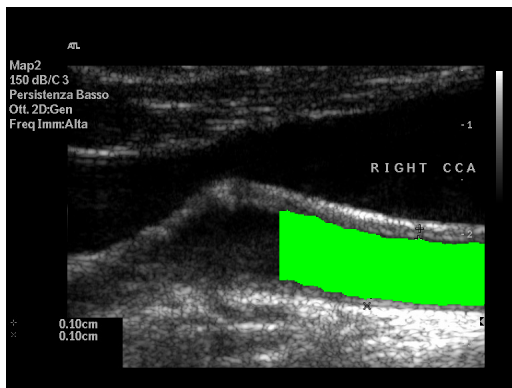
DICE = 96.91%



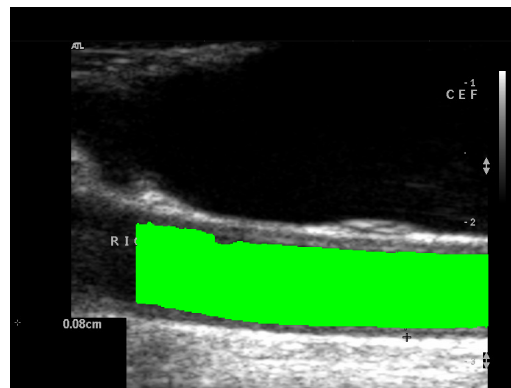
DICE = 96.49%



DICE = 99.18%



DICE = 98.80%



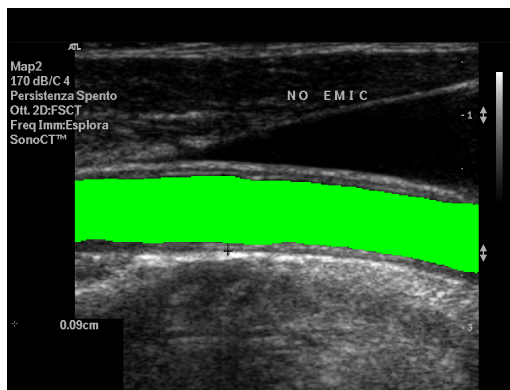
DICE = 96.94%



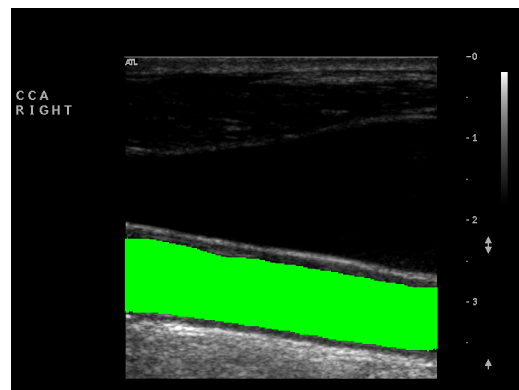
DICE = 98.74%



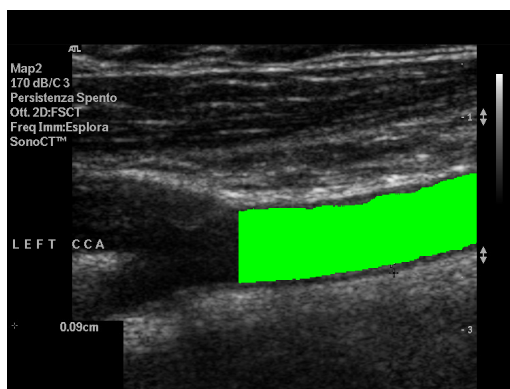
## Imagens de Teste



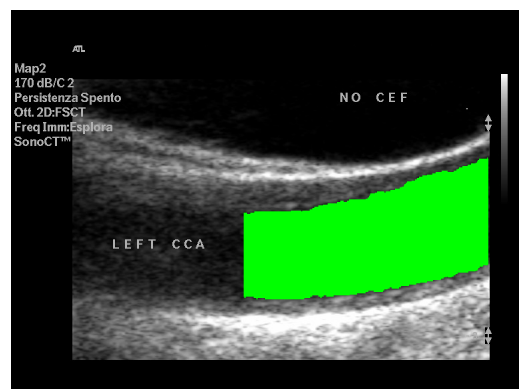
DICE = 98.30%



DICE = 98.94%



DICE = 97.15%



DICE = 97.62%