

Universidade de São Paulo
Escola Politécnica

**Segmentação de aneurismas cerebrais em imagens
3D de ressonância magnética**

Gabriel Henrique Riqueti

São Paulo
2021

Gabriel Henrique Riqueti

Segmentação de aneurismas cerebrais em imagens 3D de ressonância magnética

Trabalho de Conclusão de Curso apresentado
à banca avaliadora do curso de Engenharia
Mecatrônica da Escola Politécnica da Uni-
versidade de São Paulo para obtenção do tí-
tulo de Engenheiro Mecatrônico.

Universidade de São Paulo

Escola Politécnica

Orientador: Prof. Dr. Hae Yong Kim

São Paulo

2021

Resumo

Aneurismas cerebrais são patologias geralmente assintomáticas e com uma letalidade relativamente alta. O diagnóstico é realizado por um especialista, que examina centenas de imagens médicas diariamente em um processo repetitivo e fatigante. Para mitigar estes problemas, soluções por aprendizado profundo são desenvolvidas. Estas soluções utilizam apenas características locais da imagem para realizar a tarefa de segmentação semântica, exceto o modelo GLIA-Net que concilia as informações globais de localização e contexto do aneurisma com as informações locais referentes aos voxels vizinhos mais próximos da região em uma única rede neural e estabeleceu um novo referencial de segmentação de aneurismas em imagens de angiografia por tomografia computadorizada (ATC). Este trabalho é responsável pelas seguintes contribuições: **1)** Adaptação do projeto GLIA-Net para a segmentação semântica de aneurismas cerebrais em imagens de TOF angio-RM, uma modalidade não ionizante e não invasiva diferentemente da modalidade ATC; **2)** Treinamento do modelo GLIA-Net no banco de dados do desafio ADAM, um desafio de segmentação semântica de aneurismas intracranianos; **3)** Proposição do uso de uma média aritmética ponderada linearmente que ressalte os voxels centrais de um *patch* na composição da máscara de segmentação do exame a partir da máscara dos *patches* da imagem original. Foram obtidos os seguintes resultados em um banco de dados de validação com 23 exames: coeficiente de similaridade de Sorensen-Dice = 0,110, distância de Hausdorff $0,95 = 47,006$ mm, similaridade volumétrica = 0,406, sensibilidade por alvo = 0,447 e quantidade de falso-positivos por caso por alvo = 8,2.

Palavras-chave: Segmentação. Aneurismas cerebrais. Ressonância magnética. GLIA-Net. Aprendizado profundo.

Abstract

Intracranial aneurysms are pathologies usually asymptomatic and with a lethality relatively. The diagnosis is conducted by a specialist that examines hundreds daily in repetitive and tiresome. In order to mitigate these problems, deep learning solutions are developed. These solutions employ only local characteristics of the images to accomplish the semantic segmentation task, except the model GLIA-Net which conciliates the global information of localization and context of the aneurysm and local information regarding the neighbor voxels closer to its region in a single neural network and established a new reference for segmentation of aneurysms in computerized tomography angiography (CTA) images. This work is responsible for the following contributions: **1)** Adaptation of the GLIA-Net project for the segmentation of intracranial aneurysms in TOF MRA images, a non-invasive and non-ionizing modality differently from the CTA; **2)** Training of the GLIA-Net model in the dataset of the challenge ADAM, a challenge of semantic segmentation of intracranial aneurysms; **3)** Proposition of usage of a linear weighted arithmetic mean that emphasizes the central voxels of a patch in the composition of the segmentation mask of the exam from the patches of the original image. The following results were obtained in a validation dataset with 23 exams: Sorensen-Dice score coefficient = 0.110, Hausdorff distance $0.95 = 47.006$ mm, volumetric similarity = 0.406, sensitivity per target = 0.447 and false-positives per case per target = 8.2. **Keywords:** Segmentation. Intracranial aneurysms. MRI. GLIA-Net. Deep learning.

Lista de ilustrações

Figura 1 – Arquitetura U-Net.	29
Figura 2 – Arquitetura 3D U-Net.	29
Figura 3 – Arquitetura GLIA-Net.	34
Figura 4 – Global localizer da rede GLIA-Net.	35
Figura 5 – Feature generator da rede GLIA-Net.	35
Figura 6 – Localizer generator da rede GLIA-Net.	36
Figura 7 – Localizer loss da rede GLIA-Net.	37
Figura 8 – Global positioning adaptor da rede GLIA-Net.	37
Figura 9 – Encode block da rede GLIA-Net.	38
Figura 10 – Decode block da rede GLIA-Net.	38
Figura 11 – ResNet block da rede GLIA-Net.	39
Figura 12 – Output conv da rede GLIA-Net.	40
Figura 13 – Seções de imagem 3D do exame TOF angio-RM do dado 10072F do desafio ADAM	49
Figura 14 – Seções de imagem 3D do exame FLAIR do dado 10072F do desafio ADAM	50
Figura 15 – Seções de imagem 3D do exame de IRM ponderado em T1 do dado 10067B do desafio ADAM	51
Figura 16 – Seções de imagem 3D do exame de IRM ponderado em T2 do dado 10033 do desafio ADAM	52
Figura 17 – Seções de máscara 3D do dado 10072F do desafio ADAM. Consulte o Quadro 6 para entender o mapa de cores.	53
Figura 18 – Projeção da localização de todos os aneurismas não tratados nem rompidos do banco de dados ADAM. Os eixos x e y correspondem, respectivamente, aos eixos da direita à esquerda do paciente e de frente para trás do paciente.	54
Figura 19 – Projeção da localização de todos os aneurismas não tratados nem rompidos do banco de dados ADAM. Os eixos x e z correspondem, respectivamente, aos eixos da direita à esquerda do paciente e de baixo para cima do paciente.	55
Figura 20 – Projeção da localização de todos os aneurismas não tratados nem rompidos do banco de dados ADAM. Os eixos y e z correspondem, respectivamente, aos eixos de frente para trás do paciente e de baixo para cima do paciente.	56

Figura 21 – Histograma da localização e do tamanho dos 125 aneurismas não tratados nem rompidos do banco de dados disponibilizado aos participantes do desafio ADAM.	57
Figura 22 – Função de custo total no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	66
Figura 23 – Função de custo local no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	67
Figura 24 – Função de custo global no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	67
Figura 25 – Função de custo total no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	68
Figura 26 – Função de custo local no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	68
Figura 27 – Função de custo global no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	69
Figura 28 – Taxa de aprendizado ao longo do treinamento do modelo GLIA-Net no desafio ADAM.	70
Figura 29 – Métrica AP (precisão média) obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	71
Figura 30 – Métrica AUC (área embaixo da curva ROC) obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	71
Figura 31 – Métrica DSC (coeficiente de similaridade de Sorensen-Dice) obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	72
Figura 32 – Métrica HD95 (distância de Hausdorff considerando o percentil 95) obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	72
Figura 33 – Métrica precisão por voxel obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	73
Figura 34 – Métrica sensibilidade por voxel obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	73
Figura 35 – Métrica precisão por exame obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	74
Figura 36 – Métrica sensibilidade por exame obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.	74
Figura 37 – Métrica AP (precisão média) obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	75

Figura 38 – Métrica AUC (área embaixo da curva ROC) obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	75
Figura 39 – Métrica DSC (coeficiente de similaridade de Sorensen-Dice) obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	76
Figura 40 – Métrica HD95 (distância de Hausdorff considerando o percentil 95) obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	76
Figura 41 – Métrica precisão por voxel obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	77
Figura 42 – Métrica sensibilidade por voxel obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	77
Figura 43 – Métrica precisão por exame obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	78
Figura 44 – Métrica sensibilidade por exame obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.	78
Figura 45 – Métricas de segmentação do modelo GLIA-Net no passo 370001 para diferentes valores de limiar de segmentação calculadas no banco de dados de validação do desafio ADAM dando um peso maior aos voxels centrais na composição da máscara de segmentação a partir dos <i>patches</i> da imagem.	81
Figura 46 – Métricas de detecção do modelo GLIA-Net no passo 370001 para diferentes valores de limiar de segmentação calculadas no banco de dados de validação do desafio ADAM dando um peso maior aos voxels centrais na composição da máscara de segmentação a partir dos <i>patches</i> da imagem.	82
Figura 47 – Métricas de segmentação do modelo GLIA-Net no passo 370001 para diferentes valores de limiar de segmentação calculadas no banco de dados de validação do desafio ADAM dando um peso igual para todos os voxels na composição da máscara de segmentação a partir dos <i>patches</i> da imagem.	83
Figura 48 – Métricas de detecção do modelo GLIA-Net no passo 370001 para diferentes valores de limiar de segmentação calculadas no banco de dados de validação do desafio ADAM dando um peso constante para todos os voxels na composição da máscara de segmentação a partir dos <i>patches</i> da imagem.	84

Figura 49 – Imagem de TOF angio-RM do exame 10063F do banco de dados de validação do desafio ADAM anotada e segmentada pelo modelo GLIA-Net treinado usando a saída <i>net</i> e o limiar de segmentação 0,624. . . .	86
Figura 50 – Imagem de TOF angio-RM do exame 10063F do banco de dados de validação do desafio ADAM anotada e segmentada pelo modelo GLIA-Net treinado usando a saída <i>global localizer logits</i>	87

Lista de quadros

1	Fonte das arquiteturas de classificação de aneurismas intracranianos do estado da arte	19
2	Fonte das soluções ranqueadas no desafio ADAM	20
3	Matriz de confusão com duas classes.	21
4	Exemplo de matriz de confusão com duas classes desbalanceadas.	27
5	Resumo sobre a interpretação básica de métricas de detecção e de segmentação.	28
6	Legenda da máscara de segmentação do desafio ADAM	52
7	Composição do banco de dados de treino.	60
8	Composição do banco de dados de validação.	61

Lista de tabelas

Tabela 1	– Comparação das arquiteturas de classificação de aneurismas intracranianos do estado da arte	18
Tabela 2	– Classificação das equipes no desafio ADAM	20
Tabela 3	– Performance das redes neurais na tarefa de segmentação de aneurismas cerebrais em imagens de ATC no banco de dados de teste.	32
Tabela 4	– Resultados do treinamento de uma rede de segmentação do tipo 3D U-Net treinando apenas os <i>decoders</i> e treinando a rede inteira nos bancos de dados <i>Liver-CT</i> e <i>CP-younger fetus</i>	45
Tabela 5	– Performance de uma rede de segmentação de fígado do tipo 3D U-Net em banco de dados de teste de imagens TC anotadas partindo de parâmetros aleatoriamente inicializados (R.I) e partindo de parâmetros de um modelo pré-treinado nos bancos de dados de imagens de RM anotadas.	46
Tabela 6	– Estatísticas de localização e de tamanho dos 125 aneurismas não tratados nem rompidos do banco de dados disponibilizado aos participantes do desafio ADAM.	54
Tabela 7	– Estatísticas referentes ao espaçamento dos 113 pares de imagens disponíveis para treinamento e validação no desafio ADAM.	55
Tabela 8	– Estatísticas referentes ao tamanho dos 113 pares de imagens disponíveis para treinamento e validação no desafio ADAM.	56
Tabela 9	– Volume em voxels de cada classe nos 113 anotações com espaçamento original disponíveis para treinamento e validação no desafio ADAM.	58
Tabela 10	– Estatísticas referentes à intensidade dos voxels de valor mínimo, mediano e máximo das 113 imagens de TOF angio-RM disponíveis para treinamento e validação no desafio ADAM.	58
Tabela 11	– Métricas do melhor resultado da solução GLIA-Net no banco de dados de validação do desafio ADAM ao longo do treinamento.	80
Tabela 12	– Métricas do modelo GLIA-Net no passo 370001 no banco de dados de validação do desafio ADAM considerando a varredura completa da imagem dos exames e um peso maior aos voxels centrais na composição da máscara de segmentação a partir dos <i>patches</i> da imagem e escolhendo o limiar de segmentação que maximiza a métrica DSC	80

Tabela 13 – Métricas do modelo GLIA-Net no passo 370001 no banco de dados de validação do desafio ADAM considerando a varredura completa da imagem dos exames e um peso constante para todos os voxels na composição da máscara de segmentação a partir dos *patches* da imagem e escolhendo o limiar de segmentação que maximiza a métrica **DSC**. . . 83

Lista de abreviaturas e siglas

angio-RM	Angiografia por messonância magnética
ATC	Angiografia por tomografia computadorizada
AUC	<i>Area under curve</i> – Área abaixo da curva ROC
DSA	<i>Digital subtraction angiography</i> – angiografia por subtração digital
DSC	Coefficiente de similaridade de Sorensen-Dice
FLAIR	<i>Fluid-Attenuated Inversion Recovery</i> – recuperação por inversão de fluido atenuado
FN	Falso-negativo
FP	Falso-positivo
GPU	<i>Graphical Processing Unit</i> – Unidade de Processamento Gráfico.
HD95	Distância de Hausdorff considerando o percentil 95 no lugar do máximo das distâncias
HU	Unidade Housenfield
IC	Intervalo de confiança
IRM	Imagem por ressonância magnética
PRC	<i>Precision-Recall curve</i> – Curva precisão-revocação
RM	Ressonância magnética
ROC	<i>Receiver Operator Characteristic</i> – Característica de Operação do Receptor
TOF	<i>Time of Fly</i> – Tempo de voo
TN	Verdadeiro-negativo
TP	Verdadeiro-positivo
VS	Similaridade volumétrica
2D	Bidimensional
3D	Tridimensional

Sumário

1	INTRODUÇÃO	14
2	REVISÃO BIBLIOGRÁFICA	16
2.1	Diagnóstico de aneurismas cerebrais	16
2.1.1	Imagens de suporte	17
2.2	Detectores automáticos de aneurismas	18
2.2.1	Métricas	21
2.2.1.1	Métricas de detecção	21
2.2.1.1.1	Quantidade de falso-positivos (FP)	22
2.2.1.1.2	Sensibilidade	22
2.2.1.2	Métricas de segmentação	22
2.2.1.2.1	Coeficiente de similaridade de Sorensen-Dice	23
2.2.1.2.2	Distância de Hausdorff	23
2.2.1.2.3	Similaridade volumétrica	24
2.2.1.2.4	AUC (ROC)	25
2.2.1.2.5	Precisão média (PRC)	26
2.2.2	Arquiteturas de aprendizado profundo para segmentação	28
2.2.2.1	GLIA-Net	31
2.2.2.1.1	Função de custo	41
2.3	Transferência de aprendizado	42
3	MATERIAIS E MÉTODOS	47
3.1	Análise de dados do desafio ADAM	48
3.2	Treinamento do modelo GLIA-Net no banco de dados ADAM	59
3.2.1	Transferência de aprendizado	59
3.2.2	Protocolo de treinamento	60
3.3	Segmentação de exames de TOF angio-RM	63
3.3.1	Determinação do limiar de segmentação	65
4	RESULTADOS E DISCUSSÃO	66
4.1	Treinamento de GLIA-Net no banco de dados ADAM	66
4.1.1	Função de custo	66
4.1.2	Métricas	70
4.2	Segmentação de exames de TOF angio-RM	80
5	CONCLUSÃO	88

REFERÊNCIAS 89

1 Introdução

Aneurismas cerebrais são dilatações patológicas na parede de artérias cerebrais. Estas regiões estão propensas à ruptura, podendo provocar hemorragias subaracnoideas (ZHAO et al., 2018). A incidência de hemorragias subaracnoideas globalmente é de 7,9 por 100.000 pessoas (6,9 - 9,0, IC 95%) (ETMINAN et al., 2019), sendo que 85% destas são causadas pela ruptura de aneurismas cerebrais (van Gijn; KERR; RINKEL, 2007). Quanto à taxa de fatalidade de hemorragias subaracnoides, ela é relativamente alta mundialmente, variando de 8,3 a 66,7% (NIEUWKAMP et al., 2009).

O diagnóstico da patologia é geralmente realizado por meio da angiografia por ressonância magnética "*time-of-flight*" (TOF angio-RM), da angiografia por tomografia computadorizada (ATC) ou da angiografia por subtração digital (DSA), sendo apenas a primeira uma técnica não invasiva sem a injeção de fluidos ionizantes e sem o emprego de radiação ionizante (SETTECASE; RAYZ, 2021). Com o aumento da quantidade de imagens geradas por exame, os radiologistas possuem um tempo cada vez menor para a avaliação de cada imagem. A quantidade exorbitante de imagens a serem analisadas durante longas jornadas de 8 h podem provocar fadiga e deterioração da qualidade das avaliações (MAURER et al., 2021) (MCDONALD et al., 2015).

Estudo recente revela que algoritmos de aprendizado profundo já podem aumentar a sensibilidade de radiologistas na detecção de aneurismas cerebrais, todavia o algoritmo usado apresenta uma sensibilidade (90%) inferior a dos médicos que participaram do exame (95% e 94%), uma taxa de falsos positivos elevada (6,1) e uma comparação dos tempos de diagnóstico não foi apresentada (FARON et al., 2020).

Assim, o desenvolvimento de abordagens de detecção automáticas e semi-automáticas pode ter um impacto importante na melhoria da detecção de aneurismas intracranianos e no aumento da quantidade de exames realizados caso o tempo computacional de detecção seja inferior ao do humano.

Quanto aos tratamentos convencionais, estes consistem em métodos cirúrgicos ou endovasculares que visam isolar o aneurisma da circulação do vaso sanguíneo (ZHAO et al., 2018). Nestes tratamentos, a visualização 3D do segmento que apresenta a patologia pode auxiliar no entendimento da patologia e na aplicação do tratamento.

Estudos anteriores já implementaram arquiteturas de aprendizado profundo na classificação e segmentação de aneurismas cerebrais, todavia muitos utilizam bases de dados confidenciais de aneurismas e exames representados por voxels (UEDA et al., 2019) (ALLENBY et al., 2020), enquanto no caso do banco de dados (YANG et al., 2020) utiliza-se segmentos de vasos sanguíneos representados em nuvens de pontos, uma representação

que apesar de mais eficiente exige um procedimento de restituição não automático e trabalhoso.

Recentemente, foi publicado um desafio de detecção e segmentação de aneurismas cerebrais, chamado *ADAM - Aneurysm Detection And segMentation Challenge*, que dispõe de dados de acesso sob inscrição e permite a comparação de algoritmos de maneira mais fidedigna, usando as mesmas métricas e a mesma base de dados de teste (TIMMINS et al., 2020).

A maioria das soluções bem classificadas neste desafio empregam a seguinte estratégia: dividir a imagem em múltiplos elementos menores e segmentar separadamente cada elemento por meio de uma variação do modelo 3D U-Net (ÇIÇEK et al., 2016). Ainda que bons resultados sejam atingidos por meio desta abordagem, como consta na Tabela 2, as informações globais sobre o posicionamento do suposto aneurisma na cabeça não são utilizadas na tarefa, o que poderia melhorar os resultados de segmentação.

Neste contexto, o modelo GLIA-Net (BO et al., 2021) parece promissor por integrar informações locais e globais em uma única solução. O modelo foi treinado em uma base de dados de exames de angiografia por tomografia computadorizada (ATC) com 1363 aneurismas e 1186 casos, testado em três bases externas com pelo menos 50 aneurismas cada e ainda testado em um experimento clínico com radiologistas e residentes. Em todas estas tarefas, GLIA-Net mostrou resultados promissores, ultrapassando a performance de modelos como uma versão 3D do clássico U-Net (RONNEBERGER; FISCHER; BROX, 2015) e o antigo estado da arte em segmentação de aneurismas HEADXNet (PARK et al., 2019). Além de mostrar no experimento clínico o seu potencial para reduzir o tempo de diagnóstico e aprimorar a sua qualidade.

Assim, neste trabalho almeja-se implementar uma arquitetura de aprendizado profundo GLIA-Net capaz de segmentar exames de TOF angio-RM representados em voxels representando regiões normais e regiões com aneurismas cerebrais.

As principais contribuições deste trabalho são a adaptação do projeto GLIA-Net para a segmentação de aneurismas cerebrais em exames de TOF angio-RM, o treinamento deste modelo no desafio ADAM, o que permite a comparação dos resultados com outras soluções, e a proposição de uma composição das máscaras de segmentação de cada bloco da imagem com pesos que destaquem a região central do bloco, o que mitiga artefatos e problemas na segmentação devido a aneurismas entre blocos.

2 Revisão bibliográfica

A revisão bibliográfica foi dividida em três partes: a primeira retrata o processo de diagnóstico de aneurismas cerebrais, a segunda discorre sobre o estado da arte dos algoritmos de detecção e segmentação da patologia e a terceira aborda a transferência de aprendizado no contexto de aprendizado profundo.

2.1 Diagnóstico de aneurismas cerebrais

A captura de imagens da vasculatura cerebral é geralmente realizada por meio de três técnicas de captura de imagem: a angiografia por ressonância magnética (angio-RM), a angiografia por tomografia computadorizada (ATC) e a angiografia por subtração digital (DSA) (SETTECASE; RAYZ, 2021).

DSA é o padrão ouro usado na captura de imagens neurovasculares, sendo usado para diagnóstico de patologias, planejamento cirúrgico e em intervenções intravasculares guiadas por imagens. O método consiste na aplicação de um contraste radiopaco ionizado no vaso de interesse por meio de um cateter e na captura de sequências de imagens de raio X, que mostram o fluxo do material injetado pela vasculatura (SETTECASE; RAYZ, 2021).

ATC é uma modalidade de angiografia que consiste na aplicação de contraste ionizado intravenoso e na captura de sequências de imagens da vasculatura de interesse. Esta técnica utiliza as propriedades de atenuação de raio X dos tecidos e do contraste de forma semelhante à DSA, todavia por meios menos invasivos (SETTECASE; RAYZ, 2021). Estudos sobre a capacidade do ATC em detectar aneurismas cerebrais não rompidos comparado ao DSA revelam valores de sensibilidade entre 90% e 100% valores de especificidade entre 87,8% e 100% (HOWARD et al., 2019).

Na imagem de tomografia computadorizada, os valores da unidade da imagem são representados na escala Hounsfield (HU), que expressa o coeficiente de atenuação da onda eletromagnética de raio X do material em relação ao da água na temperatura ambiente, da seguinte maneira:

$$HU_{material} = \frac{\mu_{material} - \mu_{H_2O}}{\mu_{H_2O}} \quad (2.1)$$

onde μ_i é o coeficiente de atenuação da onda de raio X do material i e H_2O é a água. O coeficiente de atenuação de um material depende apenas da composição do material, da temperatura e da voltagem do tubo do equipamento de raio X. Assim, intervalos de

HU em uma imagem de tomografia computadorizada podem ser associados a um tecido material específico (DANCE et al., 2014).

A angiografia por ressonância magnética *Time of flight* (TOF angio-RM) é a técnica de captura de imagem sem uso de contraste mais utilizada no diagnóstico de aneurismas cerebrais não rompidos e consiste na aplicação de um campo magnético sobre a região de interesse e na captura de sequências de imagens da vasculatura de uma região (KAKA; ZHANG; KHAN, 2021). Neste exame de imagem, a emissão de sucessivos pulsos de frequências de rádio saturam a magnetização do spin de estruturas estacionárias suprimindo o sinal destas regiões, enquanto o sangue em movimento, que não exhibe o mesmo grau de saturação de spin, apresenta um sinal intenso (KUO et al., 2019). A grande vantagem desta técnica é a ausência de injeção de fluidos ionizados e de raio x, sendo uma técnica não invasiva. O estudo (SAILER et al., 2014) com 772 aneurismas indica que a TOF angio-RM possui uma sensibilidade de 95% (95% IC 89-98%) e uma especificidade de 89% (95% IC 80-95%).

Estes valores de sensibilidade de ATC e angio-RM dependem do tamanho do aneurisma, sendo que aneurismas menores que 3 mm são mais difíceis de se detectar (HOWARD et al., 2019).

2.1.1 Imagens de suporte

Além das imagens de angiografia que destacam a estrutura vascular, imagens adicionais contendo informações sobre a anatomia de uma região podem ser relevantes para o diagnóstico semi-automático ou automático, como é o caso do desafio ADAM (TIMMINS et al., 2020). Nele, além de imagens de angiografia por ressonância magnética do tipo *Time of flight*, imagens 3D do tipo ponderadas em T1, em T2 ou FLAIR também são disponíveis.

As imagens por ressonância magnética ponderadas em T1 e em T2 são imagens geradas de tal forma que a intensidade de seus voxels estão relacionados principalmente com a propriedade T1 ou T2 dos tecidos examinados, onde T1 é o tempo de relaxamento longitudinal do spin e T2 é o tempo de relaxamento transversal do spin (MAIER et al., 2018).

A sequência FLAIR (*Fluid-Attenuated Inversion Recovery*) de ressonância magnética é uma imagem de ressonância magnética ponderada em T2 que reduz drasticamente o sinal do fluido cerebrospinal (HAJNAL et al., 1992).

2.2 Detectores automáticos de aneurismas

Como resultado da captura de imagens durante o diagnóstico, produz-se uma representação volumétrica de três dimensões em voxels da vasculatura cerebral. Assim, a maioria dos artigos encontrados (SICHTERMANN et al., 2019) (ALLENBY et al., 2020) (LIU et al., 2021) (KAKA; ZHANG; KHAN, 2021) (UEDA et al., 2019) aplicam métodos de aprendizado estatístico ou alguma ferramenta manualmente desenvolvida para a detecção de aneurismas cerebrais usando a representação original dos exames em voxels.

A Tabela 1 mostra o estado da arte em detecção de aneurismas cerebrais. Todavia, a comparação é complicada já que cada artigo utiliza um banco de dados distinto e a sensibilidade dos métodos de captura de imagem e dos algoritmos de detecção são impactados pelo tamanho dos aneurismas do banco de dados.

Tabela 1 – Comparação das arquiteturas de classificação de aneurismas intracranianos do estado da arte

Modelo	Representação	SE (%)	ESP (%)	FP	F1	AUC (%)
PointCNN	Nuvem	85,81	98,95	-	0,9044	-
PointNet++	Nuvem	88,51	98,52	-	0,9029	-
SO-NET	Nuvem	83,94	98,88	-	0,8950	-
SpiderCNN	Nuvem	87,90	97,28	-	0,8722	-
DGCNN	Nuvem	83,40	97,93	-	0,8594	-
PointNet	Nuvem	69,50	93,74	-	0,6916	-
GLIA-Net	Voxel	96,2	38,9	4	-	98.2
ResNet-18	Voxel	78,0	-	2	-	-
3D-Dense-UNet	Voxel	98,6	-	1,49	-	-
Manual*	Voxel	81	86	0,2	-	-
DeepMedic	Voxel	90	-	6	-	82

Nota: **SE**, **ESP** e **FP** representam, respectivamente, sensibilidade, especificidade, falso-positivos por exame. A fonte das arquiteturas utilizadas consta no Quadro 1.

Nota: **Manual** se refere a um algoritmo desenvolvido sem um processo de aprendizado no qual os dados atualizam os parâmetros de um modelo automaticamente visando minimizar um erro.

Nota: A representação anotada como nuvem se refere à representação de nuvem de pontos.

A comparação entre métodos de detecção que utilizam diretamente as sequências de imagens representadas em voxels e os resultados obtidos a partir de banco de dados representado por nuvens de pontos (intrA) apresenta vários desafios. As métricas utilizadas em ambos são distintas e a o processo de conversão da sequência de imagens capturadas por meio de angio-RM representadas por voxels para uma nuvem de pontos exige uma restauração manual dos dados (YANG et al., 2020). Assim, os algoritmos implementados no banco de dados de nuvens de pontos não podem ser implementados de forma completamente automática e a avaliação do tempo de detecção do sistema depende da velocidade dos médicos em restaurar os modelos CAD gerados.

Quadro 1: Fonte das arquiteturas de classificação de aneurismas intracranianos do estado da arte

Modelo	Fonte
PointCNN	(LI et al., 2018) (YANG et al., 2020)
PointNet++	(QI et al., 2017) (YANG et al., 2020)
SO-NET	(LI; CHEN; LEE, 2018) (YANG et al., 2020)
SpiderCNN	(XU et al., 2018) (YANG et al., 2020)
DGCNN	(WANG et al., 2018) (YANG et al., 2020)
PointNet	(QI et al., 2016) (YANG et al., 2020)
GLIA-Net	(BO et al., 2021)
ResNet-18	(UEDA et al., 2019)
3D-Dense-UNet	(LIU et al., 2021)
Manual	(ALLENBY et al., 2020)
DeepMedic	(SICHTERMANN et al., 2019) (KAMNITSAS et al., 2017)

Assim, bancos de dados de sequências de imagens provenientes de exames de angio-RM, ATC ou DSA são mais interessantes do que bancos de dados convertidos em nuvens de pontos. Nesse sentido, o banco de dados de detecção e segmentação de aneurismas à partir sequências de imagens de TOF angio-RM, chamado ADAM (Aneurysm Detection And segMentation) (TIMMINS et al., 2020), se apresenta com muita relevância.

ADAM é não só um banco de dados como também um desafio de detecção e segmentação de aneurismas. No presente momento, o ranqueamento das equipes em ambas as tarefas está exposto na Tabela 2.

Um problema geral para a comparação da performance é que as métricas sensibilidade, especificidade, taxa de falsos-positivos e F1-score dependem do limiar escolhido para determinar se um segmento contém aneurismas ou não. A variação deste limiar resulta em um aumento da sensibilidade aliado a um aumento da taxa de falsos-positivos ou na diminuição de ambos, caracterizando a curva ROC (curva Característica de Operação do Receptor). Logo, a comparação ideal usaria uma métrica que engloba toda os pontos desta curva. A métrica AUC (Área abaixo da curva ROC) desempenha este exato papel, todavia poucos artigos revelam o seu valor.

O desafio ADAM possui vantagens tanto no aspecto da dependência de dados quanto na variabilidade das métricas. O compartilhamento de um banco de dados de teste único e secreto permite a comparação dos algoritmos mais fiel por ser independente dos dos dados e uma avaliação automática baseada em diversas métricas estabelecidas permite ponderar os diferentes erros possíveis de uma solução. Contudo, todas as métricas do desafio dependem do limiar de classificação, e portanto, este deve ser bem escolhido para obter uma boa performance nas tarefas.

Tabela 2 – Classificação das equipes no desafio ADAM

Equipe	Modelo	1	2	FP/caso	RT	DSC	HD95	VS
abc	3D U-Net E5	1	1	0,40	0,68	0,43	16,78	0,59
Joker	nnU-Net, 3D U-Net E4	3	3	0,16	0,63	0,40	8,67	0,48
JunMa	nnU-net, 3D U-Net E5	4	2	0,18	0,61	0,41	8,96	0,50
mibaumgartner	Retina U-Net E5	2	-	0,13	0,67	-	-	-
Kubiac	U-Net, QuickNAT E18	5	5	0,36	0,60	0,28	18,13	0,39
Unil_chuv	3D U-Net	7	4	1,18	0,59	0,32	22,92	0,56
xlim	AneurysmNet	6	7	4,03	0,70	0,21	36,82	0,39
quan12_ADAM	-	10	5	1,28	0,51	0,32	20,30	0,45
Interneural	NN (EfficientNet-B1)	8	6	0,88	0,49	0,17	23,98	0,36
Zelosmediacorp	3D version of U-Net E4	9	8	0,05	0,21	0,09	9,79	0,13
joana	CNN	8	-	9,37	0,63	-	-	-
Stronger	3D U-Net, 3D CNN E3	10	9	0,45	0,20	0,07	24,42	0,21
Unil_chuv	3D U-Net	11	-	1,45	0,20	-	-	-
joana	CNN	12	-	1,02	0,15	-	-	-
Trinet	-	9	10	31,80	0,76	0,11	62,35	0,53
IBBM	Btrfly	13	10	0,01	0,02	0,01	12,77	0,01
TUM_IBBM	U-Net	14	11	22,62	0,43	0,07	65,02	0,31
DeepMedic	DeepMedic	16	14	118,86	0,85	0,07	71,41	0,34

Fonte: (UTRECHT, 2020)

Nota: As colunas **1** e **2** contêm a posição de cada equipe nas tarefas 1 e 2 do desafio, sendo a tarefa 1 a detecção de aneurismas e a tarefa 2 a segmentação de aneurismas.

Nota: **FP/caso**, **RT**, **DSC**, **HD95** e **VS** correspondem às seguintes métricas falso-positivos por caso, sensibilidade, coeficiente de similaridade de Sorensen-Dice, distância de Hausdorff adaptada para o percentil 95 em mm e similaridade volumétrica.

Nota: Quanto maior o valor de **RT**, **DSC** e **VS** melhor é a performance da solução, enquanto quanto menor é o valor de **FP/caso** e **HD95** melhor a performance da solução.

Nota: **En** indica que a predição final consiste no voto majoritário de n modelos treinados de forma separada, usando *ensemble methods*.

Nota: A citação dos modelos utilizados consta no Quadro 2.

Quadro 2: Fonte das soluções ranqueadas no desafio ADAM

Model	Fonte
U-Net	(RONNEBERGER; FISCHER; BROX, 2015)
3D U-Net	(ÇIÇEK et al., 2016)
nnU-net	(ISENSE et al., 2019)
Retina U-Net	(JAEGER et al., 2018)
QuickNAT	(ROY et al., 2018)
AneurysmNet	-
NN (EfficientNet-B1)	(TAN; LE, 2019)
CNN	(NAKAO et al., 2018)
Btrfly	(SHIMIZU et al., 2020)
DeepMedic	(KAMNITSAS et al., 2017) (SICHTERMANN et al., 2019)

2.2.1 Métricas

Métricas em aprendizado estatístico supervisionado são funções responsáveis por quantificar a performance de um algoritmo em relação a uma tarefa anotada. As métricas a seguir são utilizadas para a avaliação de algoritmos no banco de dados ADAM. A sua descrição está baseada no artigo (TAHA; HANBURY, 2015) citado pelo website do desafio ADAM <<http://adam.isi.uu.nl/evaluation/>> e pela análise do repositório do mesmo desafio <<https://github.com/hjkuijf/ADAMchallenge>>. As métricas seguintes são aplicadas em cada exame/dado. A obtenção de métricas correspondentes a uma base de dados completa é feita pela média de uma métrica para todas os dados.

No caso do desafio ADAM, classificações positivas corresponde à presença de aneurismas não tratados nem rompidos e negativas à sua ausência.

Além das métricas utilizadas para a comparação de algoritmos no desafio ADAM, outras métricas são úteis para acompanhar a performance de uma rede neural profunda em treinamento.

Um conceito importante em tarefas de classificação é a matriz de confusão, que contabiliza a quantidade de acertos e erros em função das classes envolvidas. No caso da classificação com duas classes, chama-se uma classe de positiva e a outra de negativa e a matriz de confusão toma a forma do Quadro 3:

Quadro 3: Matriz de confusão com duas classes.

		Real	
		Positivo	Negativo
Predição	Positivo	TP	FP
	Negativo	FN	TN

Nota: TP (número de verdadeiro-positivos): quantidade de casos da classe positiva classificada corretamente.

Nota: FP (número de falso-positivos): quantidade de casos da classe negativa classificada incorretamente como positiva.

Nota: FN (número de falso-negativos): quantidade de casos da classe positiva classificada incorretamente como negativa.

Nota: TN (número de verdadeiro-negativos): quantidade de casos da classe negativa classificada corretamente.

2.2.1.1 Métricas de detecção

Em imagens médicas, a tarefa de detecção consiste em encontrar a posição do objeto procurado em uma imagens 2D ou 3D. Para quantificar a qualidade da localização de um objeto para um algoritmo é necessário comparar o resultado com a posição real. Para considerar uma localização correta não é necessário que a localização prevista seja

idêntica à anotada, apenas que a distância entre as posições seja menor que o raio do objeto detectado.

As métricas a seguir são responsáveis por quantificar a qualidade da localização de algoritmos.

2.2.1.1.1 Quantidade de falso-positivos (FP)

FP corresponde ao número de falso-positivos, ou seja, o número de regiões sem aneurismas não rompidos nem tratados mas detectadas como tal.

Esta métrica vale no mínimo 0 e não possui máximo real. Quanto menor o seu valor, melhor a performance do algoritmo.

2.2.1.1.2 Sensibilidade

A sensibilidade é dada por:

$$\text{Sensibilidade} = \frac{TP}{FN + TP}$$

onde TP e FP são, respectivamente, o número de verdadeiro-positivos e o número de falso-positivos, ou seja, a sensibilidade é a razão entre o número de casos aneurismas detectados corretamente e o número de aneurismas anotados.

Esta métrica pertence ao intervalo $[0, 1]$ e quanto mais a sensibilidade está próxima de 1, mais aneurismas são detectados corretamente e melhor a performance do algoritmo.

A sensibilidade e a quantidade de falso-positivos são métricas complementares no sentido que elas avaliam erros distintos e que juntos agregam todos os erros de detecção possíveis. Enquanto a quantidade de falso-positivos quantifica o número de aneurismas detectados erroneamente, a sensibilidade quantifica o número de aneurismas não detectados. Assim, a avaliação de um algoritmo de detecção deve ser feita com ambas as métricas.

2.2.1.2 Métricas de segmentação

Em imagens médicas, a tarefa de segmentação consiste em classificar cada pixel (no caso de imagens 2D) ou cada voxel (no caso de imagens 3D) em determinadas categorias.

A descrição de métricas desta seção compartilham as anotações S_t^i , S_g^i , $|A|$, TP , FN e FP . S_t^i e S_g^i são, respectivamente, o conjunto de voxels classificados como i por um algoritmo e o conjunto de voxels anotados como i . Se A for um conjunto, $|A|$ significa a cardinalidade de um conjunto A – e como trabalhamos com conjuntos finitos a cardinalidade de A é o número de elementos de A – e se A for um número real, $|A|$ é o seu módulo.

TP , FN e FP representam, respectivamente, o número de voxels verdadeiro-positivos, o número de voxels falso-negativos e número de voxels falso-positivos.

As métricas a seguir são responsáveis por quantificar a qualidade da segmentação de algoritmos.

2.2.1.2.1 Coeficiente de similaridade de Sorensen-Dice

Segundo (TAHA; HANBURY, 2015), o coeficiente de similaridade de Sorensen-Dice é a métrica mais utilizada na validação de segmentações de volumes na área biomédica e é dado por:

$$DSC = 2 \frac{|S_t^1 \cap S_g^1|}{|S_t^1| + |S_g^1|} = \frac{2TP}{2TP + FN + FP}$$

Esta métrica pertence ao intervalo $[0, 1]$ e quanto mais DSC está próximo de 1, menor o erro de segmentação e melhor a performance do algoritmo.

Como DSC mensura os erros falso-positivos e falso-positivos de forma conjunta comparando apenas com os verdadeiro-positivos, esta métrica é robusta para tarefas com dados desbalanceados, onde geralmente os verdadeiro-negativos são muito mais numerosos que os verdadeiro-positivos.

No artigo (TAHA; HANBURY, 2015), o coeficiente de similaridade de Sorensen-Dice é apresentado como uma métrica que avalia a intersecção entre a predição positiva e a anotação positiva em relação aos erros. Argumenta-se que esta métrica é ideal para mensurar a intersecção e que caso estas regiões sejam muito pequenas, intersecções vazias são comuns e a métrica é pouco representativa.

2.2.1.2.2 Distância de Hausdorff

Segundo (TAHA; HANBURY, 2015), a distância de Hausdorff é dada por:

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

com

$$h(A, B) = \max_{a \in A} \min_{b \in B} d(a, b)$$

com $d(A, B)$ sendo uma métrica.

No desafio, é utilizado uma modificação desta métrica. A distância de Hausdorff modificada consiste em substituir a função $h(A, B)$ acima por:

$$h_{95}(A, B) = \text{percentil_95}_{a \in A} \min_{b \in B} d(a, b)$$

onde *percentil_95* extrai o 95º percentil de um conjunto. Esta modificação permite a desconsideração de distâncias anômalas muito elevadas entre pontos de ambos os conjuntos.

No desafio, a métrica empregada é a distância Euclidiana, ou seja, dados que a , b são vetores compostos por n elementos a_i e b_i , temos:

$$d(a, b) = \sqrt{\sum_{i=1}^n (a_i - b_i)^2}$$

Além disso, no desafio os conjuntos A e B são as bordas dos aneurismas não rompidos nem tratados.

Esta métrica, sendo uma distância, possui valor mínimo nulo e não possui valor máximo real. Assim, quanto menor a distância de Hausdorff, melhor as bordas segmentadas correspondem as bordas dos aneurismas anotados e melhor é a performance do algoritmo de segmentação.

No artigo (TAHA; HANBURY, 2015), a distância de Hausdorff é considerada uma métrica de distância. Argumenta-se que por ser uma métrica que considera a distância espacial, a distância de Hausdorff é ideal para a avaliação dos contornos previsto em relação ao anotado. Esta métrica é ideal também para bancos de dados com regiões anotadas positivas muito pequenas, pois ainda que a intersecção entre os volumes previsto e anotado seja nula, é possível estimar a distância entre estes volumes.

2.2.1.2.3 Similaridade volumétrica

Segundo (TAHA; HANBURY, 2015), a similaridade volumétrica é dada por:

$$VS = 1 - \frac{||S_t^1| - |S_g^1||}{|S_t^1| + |S_g^1|} = 1 - \frac{|FP - FN|}{2TP + FN + FP}$$

Esta métrica pertence ao intervalo $[0, 1]$. No cenário de segmentação ideal, em que $FP = FN = 0$, temos que $VS = 1$. Assim, intuitivamente quanto maior a similaridade volumétrica, mais os voxels são classificados corretamente e melhor a performance do algoritmo de segmentação. Note que ainda é possível obter uma similaridade volumétrica alta caso os números de falso-positivos e falso-negativos forem correlacionados, em particular, é possível obter uma similaridade volumétrica igual a 1 mesmo com uma segmentação com $TP = 0$, basta que $FN = FP$, e portanto, esta métrica não deve ser usada de maneira isolada.

No artigo (TAHA; HANBURY, 2015), a similaridade volumétrica é uma métrica de intersecção. Da mesma forma que o coeficiente de similaridade de Sorensen-Dice não é recomendado para regiões de segmentação positivas muito pequenas em relação às negativas, a similaridade volumétrica não é recomendada também. Esta métrica não é recomendada também para a avaliação do alinhamento das regiões positivas anotadas e prevista, pois os falso-positivos podem cancelar com os falso-negativos, indicando erroneamente uma similaridade muito alta. Assim, a VS é ideal para a avaliação de intersecções no qual o alinhamento não é muito importante.

2.2.1.2.4 AUC (ROC)

A curva ROC ou curva Característica do Operador do Receptor (*Receiver Operator Characteristic*) é a curva dada pela especificidade e a sensibilidade de um algoritmo de classificação em um banco de dados para uma gama de limiares. No eixo da abcissa temos a taxa de falso-positivos (FPR) e no eixo da ordenada temos a taxa de verdadeiro-positivos (TPR) (FAWCETT, 2006).

A taxa de falso-positivos é dada por:

$$FPR = \frac{FP}{TN + FP} = 1 - Especificidade$$

A taxa de verdadeiro-positivos é dada por:

$$TPR = \frac{TP}{TP + FN} = Sensibilidade$$

As taxas FPR e TPR pertencem ao intervalo $[0, 1]$ e são complementares já que a FPR contabiliza o erro dos falso-positivos, enquanto a TPR contabiliza o erro dos falso-negativos. A performance do algoritmo é melhor para valores maiores de TPR e menores de FPR.

Assim, a Curva ROC ideal consiste em duas retas: a primeira vai do ponto ($FPR = 0, TPR = 0$), no qual o limiar é tão alto que todos os voxels são classificados como negativos, até o ponto ($FPR = 0, TPR = 1$), no qual todos os voxels são classificados corretamente; e a segunda vai do ponto ($FPR = 0, TPR = 1$) até o ponto ($FPR = 1, TPR = 1$), no qual o limiar é tão baixo que todos os voxels são considerados como positivos.

Como explicado em (FAWCETT, 2006), um algoritmo que classifica uma instância aleatoriamente, ou seja, independentemente das características da instância, possui valores esperados de TPR e FPR iguais, já que podemos interpretar TPR como a probabilidade do algoritmo classificar uma instância positiva como positiva e FPR como a probabilidade do algoritmo classificar uma instância negativa como positiva. Assim, a

linha diagonal $TPR = FPR$ de (0,0) a (1,1) é a linha básica do comportamento esperado de um classificador aleatório.

A métrica AUC é a área abaixo da curva (*area under curve*) e é útil para mensurar os erros falso-positivos e falso-negativos de forma independente do limiar do algoritmo. No cenário ideal, $AUC = 1$, enquanto espera-se $AUC = 0,5$ no cenário em que o classificador é aleatório.

Esta métrica pertence ao intervalo $[0, 1]$ e quanto maior o seu valor melhor a performance do classificador.

2.2.1.2.5 Precisão média (PRC)

Apesar da curva ROC ser muito interessante por ponderar ambos os erros de classificação para todos os valores de limiar, esta curva não é muito representativa, podendo ser extremamente otimista no cenário de dados desbalanceados. Uma alternativa mais informativa é a curva Precision-Recall (PRC) (SAITO; REHMSMEIER, 2015).

A curva PRC também é capaz de ponderar os erros falso-positivos e falso-negativos para todos os valores de limiar e possui a precisão (*precision*) como abcissa e a revocação (*recall*) como ordenada.

A precisão é dada por:

$$Precisão = \frac{TP}{TP + FP}$$

A revocação, também chamada de TPR ou de sensibilidade, é dada por:

$$Revocação = \frac{TP}{TP + FN}$$

Ambas as métricas pertencem ao intervalo $[0, 1]$ e quanto maior os seus valores, melhor a performance do algoritmo. A precisão e a revocação são complementares, já que a precisão considera os erros falso-positivos e a revocação considera os erros falso-negativos.

Assim, a curva PRC ideal consiste na reta que começa em aproximadamente ($recall = 0, precision = 1$), no qual o limiar está muito alto e um número ínfimo de voxels são detectados corretamente como positivos, e termina em ($recall = 1, precision = 1$), no qual todos os voxels são corretamente classificados.

No cenário em que o classificar é aleatório, ou seja, classifica as instâncias independentemente de suas características, espera-se uma linha, chamada de base, dada por $precision = P/(P + N)$ onde P é o número de classes positivas e N é o número de classes negativas do banco de dados. Logo, espera-se que $AP = P/(P + N)$ para um classificador

aleatório. Nota-se que apenas para bancos de dados com classes balanceadas espera-se obter $AP = 0,5$ com um classificador aleatório (SAITO; REHMSMEIER, 2015).

A métrica AP, ou precisão média (*average precision*), é a precisão média da curva PRC, que corresponde à área abaixo da curva PRC já que a abcissa revocação varia de 0 a 1. Esta métrica também pertence ao intervalo $[0, 1]$ e quanto maior o seu valor, melhor a performance do algoritmo.

No cenário de dados desbalanceados, como é o caso da tarefa de segmentação de aneurismas cerebrais, a métrica AP é mais representativa que a métrica AUC. Isso ocorre, pois a curva PRC pondera os erros falso-negativos e falso-positivos utilizando apenas estes erros e o número de verdadeiro-positivos, que é a classe menos numerosa geralmente, já a curva ROC pondera os erros falso-negativos e falso-positivos utilizando tanto o número de verdadeiro-positivos quanto o número de verdadeiro-negativos e como a classe negativa geralmente é muito mais numerosa que a positiva a métrica considera os erros de maneira desigual, permitindo um alto valor de AUC com uma performance relativamente baixa.

No artigo (SAITO; REHMSMEIER, 2015), Saito explica que a curva ROC independe da proporção do número de casos das classes positiva e negativa. Assim, no caso de dados desbalanceados apesar da curva ROC poder apresentar precisão baixa em certas regiões, a taxa FPR continua baixa na mesma região e a curva ROC não se altera. Enquanto isso, a curva PRC se adapta de acordo com a proporção do número de casos das classes positiva e negativa, acompanhando a mudança na performance dos classificadores.

O exemplo do Quadro 4 ilustra um cenário de dados desbalanceados em que um ponto da curva ROC não é representativo da performance do algoritmo, enquanto a curva PCR é mais informativa.

Quadro 4: Exemplo de matriz de confusão com duas classes desbalanceadas.

		Real	
		Positivo	Negativo
Predição	Positivo	500	1600
	Negativo	500	8400

Fonte: Adaptado de (SAITO; REHMSMEIER, 2015).

$$TPR = recall = \frac{500}{500 + 500} = 0,5\%$$

$$FPR = \frac{1600}{1600 + 8400} = 0,06\%$$

$$precision = \frac{500}{500 + 1600} = 0,02\%$$

Uma taxa FPR abaixo de 1% induz um aparente baixo número de erros falso-positivos. Todavia, quando comparado o número de falso-positivos com o número de verdadeiro-positivos, nota-se que este erro é bem expressivo. A baixa precisão captura esta informação sobre o erro de falso-positivos.

Para uma simples e rápida interpretação dos resultados, as métricas explicadas nesta seção estão resumidas no Quadro 5.

Quadro 5: Resumo sobre a interpretação básica de métricas de detecção e de segmentação.

Métrica	Mínimo	Máximo	Melhor
Falso-positivos (FP)	0	negativos (N)	↓
Falso-negativos (FN)	0	positivos (P)	↓
Verdadeiro-positivos (TP)	0	positivos (P)	↑
Verdadeiro-negativos (TN)	0	negativos (N)	↑
Sensibilidade = <i>recall</i>	0	1	↑
Precisão	0	1	↑
AUC (Área abaixo da curva ROC)	0	1	↑
AP (precisão média da curva PRC)	0	1	↑
Coefficiente de Sorensen-Dice (DSC)	0	1	↑
Distância de Hausdorff considerando apenas 95% das menores distâncias (HD95)	0	-	↓
Similaridade volumétrica (VS)	0	1	↑

Fonte: Autor.

Nota: Na coluna **Melhor**, ↓ representa que uma redução na métrica indica uma melhor performance da solução, enquanto ↑ representa que um aumento na métrica indica uma melhor performance da solução.

2.2.2 Arquiteturas de aprendizado profundo para segmentação

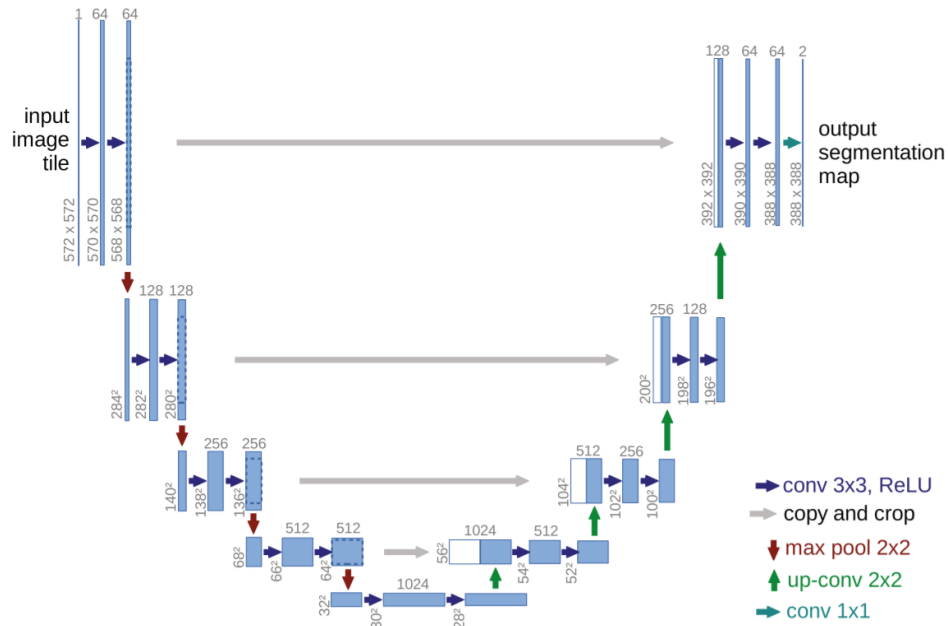
Em razão das vantagens proporcionadas pelo desafio ADAM e a representação em voxels já mencionada na seção 2.2, foca-se apenas em arquiteturas de aprendizado profundo em que os dados são representados em voxels.

Os modelos ranqueados no desafio ADAM (Tabela 2) revelam que U-Net e suas variações são a arquitetura predominante e de excelente performance. A arquitetura original do modelo U-Net consta na Figura 1.

A ideia principal por trás deste modelo é uma rede completamente convolucional ("*fully convolutional network*") em que por um caminho a rede extrai informações do contexto em uma resolução cada vez menor e com o auxílio de conexões de maior resolução, a resolução da informação extraída é aumentada até a resolução da imagem de entrada (RONNEBERGER; FISCHER; BROX, 2015).

Este modelo processa imagens bidimensionais (RONNEBERGER; FISCHER; BROX, 2015). Assim, a sua versão para imagens tridimensionais foi desenvolvida, sendo chamada

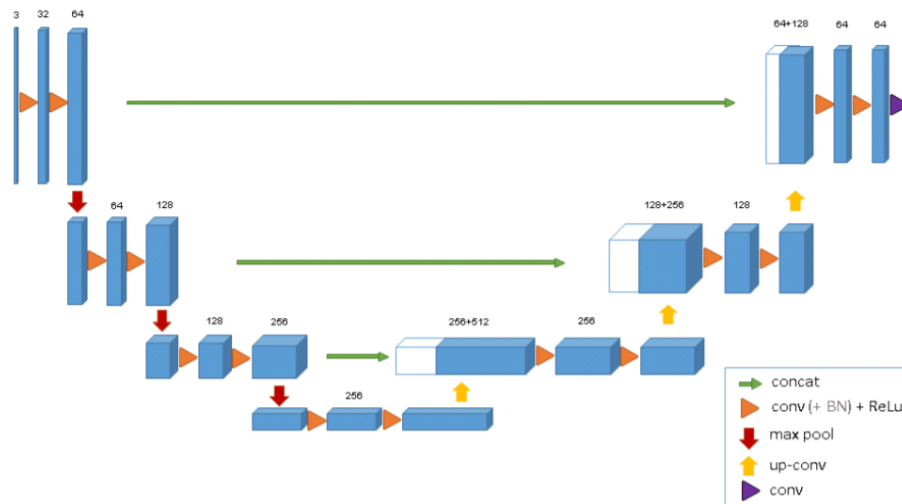
Figura 1 – Arquitetura U-Net.



Fonte: (RONNEBERGER; FISCHER; BROX, 2015).

de 3D U-Net (ÇIÇEK et al., 2016). Como mostrado na Figura 2, a ideia central por trás do model U-Net remanesce, porém as operações passam a atuar em 3 dimensões no lugar de 2.

Figura 2 – Arquitetura 3D U-Net.



Fonte: (ÇIÇEK et al., 2016)

A partir da Tabela 2, nota-se que usar uma rede neural do tipo 3D U-Net não garante os melhores no desafio. As equipes Stronger e Unil_Chuv usaram esta rede, porém obtiveram performances inferiores a outras redes neurais. O artigo (ISENSEE et al.,

2019) justifica esta discrepância. Nele, argumenta-se que o desenvolvimento de soluções especializadas de segmentação semântica é não trivial e dependente de propriedades do banco de dados alvo e do *hardware* disponível. Os autores desenvolveram uma plataforma de acesso aberto chamada nnU-Net ("nenhuma rede nova") para tratar da adaptação dos hiperparâmetros do modelo para o treinamento da rede, como *batch size*, *patch size*, a topologia da rede 3D U-Net e o pré-processamento, a partir de uma série de regras heurísticas. No mesmo artigo, foi documentado que a implementação nnU-Net foi aplicada em 10 desafios biomédicos internacionais de segmentação, englobando 19 bancos de dados diferentes, 49 tarefas de segmentação diferentes envolvendo uma variedade de órgãos e as modalidades IRM, ATC e microscopia eletrônica e das 49 tarefas a implementação nnU-Net estabeleceu um novo estado da arte em 29. Em particular, os trabalhos das equipes Unil_chuv (NOTO et al., 2020a) e Stronger (NOTO et al., 2020b) usaram a rede 3D U-Net com *patches* de tamanho, respectivamente, 32 x 32 x 32 e 64 x 64 x 32, enquanto os trabalhos das equipes ABC (ABC, 2020), JunMa (MA, 2020) e Joker (YANG, 2020), que obtiveram os melhores resultados do desafio, usaram a rede 3D U-Net com os respectivos tamanhos de *patch* 128 x 128 x 128, 256 x 224 x 56 e 225 x 192 x 56. É possível que tamanhos de *patch* mais elevados sejam responsáveis pela diferença de performance.

Nota-se também que as equipes mais bem classificadas utilizaram alguma forma de *ensemble methods*, métodos conjuntos. Nestes métodos, vários modelos são treinados independentemente e a previsão final da solução é calculada a partir do balanço da previsão dos modelos treinados.

A equipe com as melhores avaliações nas tarefas de detecção e segmentação é a equipe ABC. Os resultados foram obtidos usando a rede profunda 3D U-Net (ÇIÇEK et al., 2016), *patch size* igual a 128 x 128 x 128, uma função de custo dada pela combinação linear das funções de custo de Dice e de entropia cruzada binária e um otimizador Adam (KINGMA; BA, 2015). O resultado é a obtido usando um método conjunto com cinco modelos. Uma vasta quantidade de operações aleatórias de aumento de dados artificial foi utilizada no treinamento da rede, incluindo mudança de escala, rotação, espelhamento, mudança no contraste e a aplicação dos ruídos aditivos Gaussiano e de Poisson. Não são providos mais detalhes sobre o pré-processamento de dados (ABC, 2020).

Entre as equipes mais bem ranqueadas está a equipe JunMa, que utilizou a implementação nnU-Net. Considerando uma entrada apenas de imagens TOF angio-RM, a ferramenta nnU-Net inferiu os seguintes hiperparâmetros: *patch size* igual a 256 x 224 x 56, *batch size* igual a 2, rede 3D U-Net com cinco blocos convolucionais no caminho de diminuição da resolução e no caminho do aumento da resolução cada um com uma convolução 3D com *kernels* cúbicos de lado 3 seguidos por uma normalização por instância e por uma função de ativação Leaky ReLU (Equação 2.2) (XU et al., 2015). O pré-processamento é constituído pela exclusão de voxels pretos, considerados como plano

de fundo, normalização Z e as seguintes operações aleatórias para aumento de dados: rotação, mudança de escala, espelhamento e transformação gamma. A otimização da rede neural é feita pelo algoritmo do gradiente descendente estocástico. O resultado final é calculado usando um método conjunto com cinco redes. Na sua breve descrição, não há mais detalhes sobre a implementação destas operações de aumento de dados nem do *ensemble methods*. (MA, 2020).

A equipe Joker, terceira colocada em segmentação e em detecção, também implementou uma rede 3D U-Net usando a implementação nnU-Net. O pré-processamento de dados consiste no redimensionamento das imagens de tal forma que todas possuam o espaçamento mediano do banco de dados, na saturação dos valores da imagem nos percentis 0,5% (mínimo) e 99,5% (máximo) e na normalização Z da imagem saturada. A etapa de aumento de dados é formada pelas operações aleatórias: transformação afina (rotação, translação e deformação elástica), espelhamento e aumento por correção gamma, que na mudança das intensidades por meio de um expoente aleatório comum à imagem. No modelo, a normalização por *batch* (Batch Normalization) é substituída pela normalização por grupo e a função de ativação ReLU (Equação 2.4) é substituída pela Leaky ReLU (Equação 2.2) e função de custo usada é a *Dice ranking loss*, visando reduzir o erro de patologias menores. No treinamento, utilizou-se um *patch size* igual a 192 x 224 x 56 e um *batch size* igual a 6 (YANG, 2020). O resultado final é o voto majoritário de quatro modelos treinados independentemente e com considerações distintas: variou-se as imagens de entrada (ou apenas TOF angio-RM ou ambas as imagens) e variou-se a classificação dos aneurismas tratados ou rompidos (ou como aneurismas não tratados nem rompidos ou como regiões normais).

Com base nas descrições dos trabalhos anteriores e na Tabela 2, múltiplas redes neurais profundas 3D U-Net agrupadas usando métodos conjuntos é a solução predominante entre as soluções que obtiveram os melhores resultados no desafio ADAM. Todavia, os detalhes da sua microarquitetura, bem como o otimizador, o *patch size*, o *batch size* e as operações para o aumento de dados variam entre as equipes participantes do desafio. Como as soluções ranqueadas possuem múltiplas diferenças quando comparadas duas a duas, não é possível inferir as melhores configurações de forma independente.

2.2.2.1 GLIA-Net

GLIA-Net é um modelo de aprendizado profundo de segmentação desenvolvido para a segmentação de imagens 3D de ATC. O artigo (BO et al., 2021) relata que a rede GLIA-Net foi treinada em um banco de dados com 1186 imagens tridimensionais de ATC abrangendo 1363 aneurismas cerebrais e testado em um banco de dados com 152 imagens abrangendo 126 aneurismas cerebrais. Para comparar os resultados, as redes de segmentação semântica HeadXNet (PARK et al., 2019) e uma versão 3D da rede U-

Net (RONNEBERGER; FISCHER; BROX, 2015) foram treinadas e testadas de maneira análoga nos mesmos bancos de dados. Os resultados destes modelos no banco de dados de teste são mostrados na Tabela 3 e indicam que o modelo GLIA-Net apresenta a melhor performance. No mesmo artigo, um estudo clínico é apresentado, mostrando que o modelo pode reduzir o tempo e melhorar o diagnóstico de aneurismas cerebrais por um neurologista.

Tabela 3 – Performance das redes neurais na tarefa de segmentação de aneurismas cerebrais em imagens de ATC no banco de dados de teste.

Modelo	P	SV	DSC	HD95	AUC	AP	ST	FP/exame
U-Net	14,0	71,3	23,2	19,6	98,8	17,5	73,3	30,8
HeadXNet	16,2	55,6	23,2	15,9	98,2	25,0	54,9	15,9
GLIA-Net	48,8	72,9	57,9	9,07	98,2	61,9	82,1	4,38

Fonte: Adaptado de (BO et al., 2021).

Nota: **P**, **SV**, **DSC**, **HD95**, **AUC**, **AP**, **ST** e **FP/exame** representam, respectivamente, precisão, sensibilidade por voxel, coeficiente de similaridade de Sorensen-Dice, a distância de Hausdorff adaptado para 95% dos pares de voxels de aneurismas anotados e previstos, área sobre a curva ROC, precisão média da curva PRC, sensibilidade por aglomerado de voxels conectados e falso-positivos por exame.

Nota: As métricas **ST** e **FP/exame** são mensuradas por aglomerados de voxels vizinhos classificados como aneurismas, enquanto as outras métricas são mensuradas por voxels.

Nota: Todas as métricas são dadas em porcentagem exceto **HD95** dada em milímetros e **FP/exame**.

Nota: Apesar da rede do tipo U-Net ter obtido a métrica AUC superior às outras redes, o valor 98,8% está dentro do intervalo de confiança de 95 % da performance das redes GLIA-Net e HEADXNet.

Como aneurismas cerebrais são relativamente pequenos (da ordem de alguns milímetros) em relação a cabeça, que é mensurada na ordem de decímetros, exames de imagem tridimensionais para o diagnóstico da patologia geram imagens com uma quantidade de voxels exorbitante (no caso do desafio ADAM, na ordem de 10^7 e 10^8), no qual uma razão ínfima representa a anomalia.

Por esta razão, é desafiador criar soluções baseadas em arquiteturas 3D U-Net com entrada igual a uma imagem tridimensional inteira do exame com resolução original. Em geral, divide-se a imagem do exame em blocos (*patches*) na forma de paralelepípedos e a máscara de segmentação da imagem do exame completo é a composição das máscaras de segmentação geradas pela rede neuronal para cada um desses blocos. O problema desta solução é que apenas as informações locais dos blocos do exame de imagem são consideradas.

Uma possível alternativa seria a redução de dimensão da imagem tridimensional original para que esta possa ser utilizada como entrada da rede neuronal. Todavia, apesar de neste caso a rede aprender informações globais da imagem para segmentação perde-

se resolução e, portanto, as informações locais. Como aneurismas são extremamente pequenos em relação ao tamanho da imagem completa, a perda é considerável.

Neste cenário, aparece a solução *Global Localization-based IA* (GLIA-Net), que teve como objetivo conciliar a extração de informações locais e globais da imagem. A solução é capaz de extrair informações de localização de um elemento na imagem da cabeça como um todo ao mesmo tempo que extrai informações locais de elementos com a mesma resolução que a imagem original. A macro arquitetura da solução é apresentada na Figura 3.

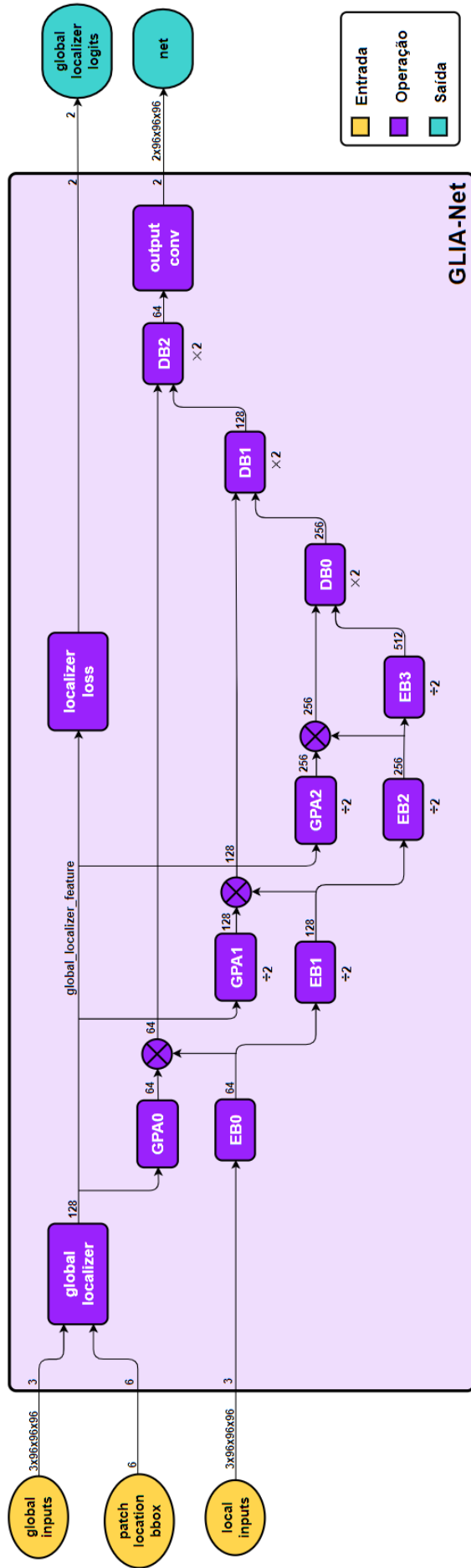
A solução GLIA-Net utiliza a estratégia de dividir a imagem 3D em uma série de blocos (*patches*) menores sobrepostos cujo tamanho é suficientemente pequeno para que as operações possam ser implementadas na memória limitada da Unidade de Processamento Gráfico (*Graphical Processing Unit – GPU*), o que permite desfrutar de um elevado grau de paralelismo, e por conseguinte, uma alta velocidade de execução. A segmentação semântica da imagem 3D completa consiste na composição da segmentação semântica de cada bloco individual. Assim, para cada bloco, o modelo GLIA-Net recebe como entrada o bloco (*local inputs*), a posição do bloco na imagem completa (*patch location bbox*) e a imagem completa com dimensão reduzida (*global inputs*) e produz como saída uma máscara classificando os voxels em aneurismas ou em regiões normais (*net*) e uma estimativa da probabilidade do bloco apresentar um aneurisma usando apenas *global inputs* (*global localizer logits*).

A macro arquitetura da rede GLIA-Net é composta por dois componentes principais: a rede de localização global (*global localization network*), responsável por extrair informações globais da distribuição de aneurismas cerebrais na imagem global, e a rede de segmentação local (*local segmentation network*), responsável por segmentar a imagem semanticamente. Ainda que esta diferenciação seja importante conceitualmente, os dois componentes não são isolados. A informação intermediária da *global localization network* chamada de *global localizer feature* é transmitida a múltiplos níveis da *local segmentation network* por meio dos componentes adaptadores do posicionamento global (*Global Positioning Adaptor – GPA*) e pela multiplicação elemento a elemento das informações locais em cada nível, como mostrado na Figura 3.

A rede *local segmentation network* é basicamente uma versão da rede 3D U-Net com quatro *encoders* (*EB*) e três *decoders* (*DB*) com a modificação já mencionada nas conexões diretas (*skip connections*) por meio dos módulos *GPA*s. Sua entrada é o bloco local da imagem (*local inputs*) e a sua saída é a máscara de segmentação correspondente ao bloco de entrada (*net*). Esta rede isolada, sem os módulos *GPA*s, e com normalização por *batch* (IOFFE; SZEGEDY, 2015) no lugar de normalização por grupo (WU; HE, 2018) é a versão 3D da rede U-Net utilizada nos testes do artigo (BO et al., 2021).

A rede *global positioning network* é composta por dois componentes: o localizador

Figura 3 – Arquitetura GLIA-Net.



Fonte: Adaptado de (BO et al., 2021).

Nota: O número em cima das linhas corresponde ao número de canais do *feature map* representado pela linha.

Nota: $\div 2$ indica que a operação divide a profundidade, a altura e a largura do *feature map* por 2. $\times 2$ indica que a operação dobra a profundidade, a altura e a largura do *feature map*.

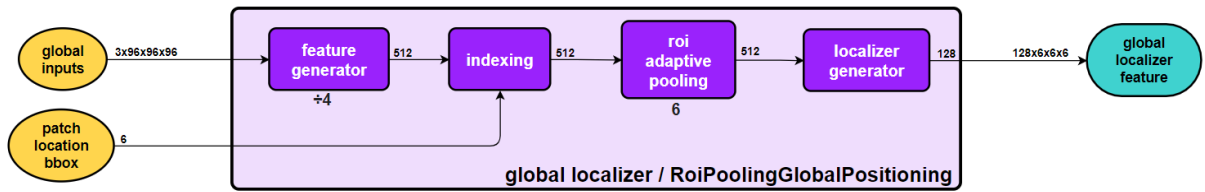
Nota: \otimes representa a operação de multiplicação de tensores termo a termo.

Nota: Apenas os tensores *patch location bbox* e *global localizer logits* são unidimensionais. Os tensores restantes possuem quatro dimensões, desconsiderando o tamanho do *batch*.

global (*global localizer*), que recebe como entrada a imagem global reduzida dimensionalmente (*global inputs*) e a localização do bloco local na imagem global (*patch location bbox*) e produz como saída o mapa de características de localização global do bloco (*global localizer feature*), e o localizador (*localizer loss*), que recebe como entrada a saída do *global localizer* e produz como saída as probabilidades do bloco local apresentar aneurismas (*global localizer logits*).

Conforme a Figura 4, o *global localizer* é constituído por quatro etapas principais em série: o gerador de características (*feature generator*), a indexação (*indexing*), o posicionamento adaptativo de regiões de interesse (*roi adaptive positioning*) e o gerador de localização (*localizer generator*).

Figura 4 – Global localizer da rede GLIA-Net.



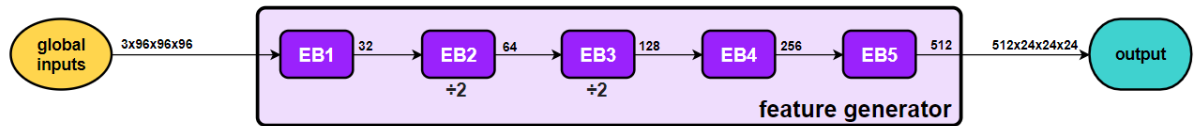
Fonte: Adaptado de (BO et al., 2021).

Nota: O número em cima das linhas corresponde ao número de canais do *feature map* representado pela linha.

Nota: $\div 4$ indica que a operação divide a profundidade, a altura e a largura do *feature map* por 4. 6 abaixo da operação indica que esta iguala a profundidade, a altura e a largura do *feature map* a 6.

O *feature generator*, exibido na Figura 5, é constituído por cinco blocos de *encoders* em série e produz um mapa de características da imagem global inteira. As dimensões largura, altura e profundidade deste mapa são iguais as respectivas dimensões da imagem global reduzida dimensionalmente (*global inputs*) divididas por quatro.

Figura 5 – Feature generator da rede GLIA-Net.



Fonte: Adaptado de (BO et al., 2021).

Nota: O número em cima das linhas corresponde ao número de canais do *feature map* representado pela linha.

Nota: $\div 2$ indica que a operação divide a profundidade, a altura e a largura do *feature map* por 2.

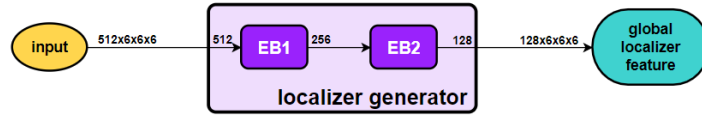
O mapa de características globais gerado pelo *feature generator* possui informações da imagem inteira. A operação de indexação é responsável por utilizar a posição do bloco

na imagem completa (*patch location bbox*) para extrair justamente a região do mapa de características globais referente à região do bloco em questão.

A operação de posicionamento adaptativo da região de interesse (*roi adaptive positioning*) aplica a operação do PyTorch `AdaptativeMaxPool3d`, que seleciona valores máximos do mapa de características da região de interesse gerado pela operação de indexação de tal forma a gerar um novo mapa de características da região referente ao bloco com altura, largura e profundidade iguais a 6.

O componente *localizer generator* recebe o mapa de características produzido pela camada *roi adaptive pooling* e gera um novo mapa de características globais da região de interesse (o bloco), chamado de *global localizer feature*, que é utilizado pela rede de segmentação local (*local segmentation network*) por conter informações provenientes da imagem completa. Esta operação é realizada por dois blocos de *encoders*, como apresentado na Figura 6.

Figura 6 – Localizer generator da rede GLIA-Net.



Fonte: Adaptado de (BO et al., 2021).

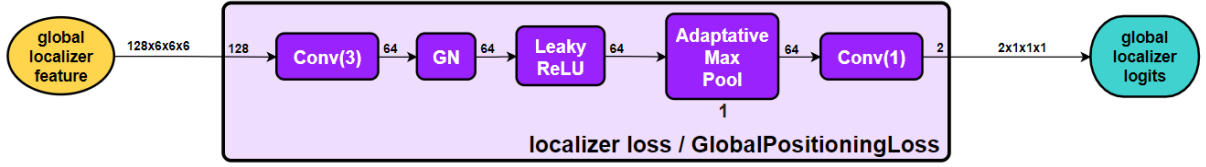
Nota: O número em cima das linhas corresponde ao número de canais do *feature map* representado pela linha.

A partir do *global localizer feature*, o localizador (*localizer loss*) estima a probabilidade do bloco de interesse apresentar um aneurisma ou não, chamada de *global localizer logit*. Assim, a rede *global localization network* produz uma estimativa da probabilidade de um bloco da imagem apresentar aneurismas utilizando apenas a imagem inteira e reduzida dimensionalmente e a posição no bloco na imagem completa. O *localizer loss* é exibido na Figura 7 e é formado pelas operações em série: convolução 3D com *kernels* cúbicos de lado 3, normalização por grupo (WU; HE, 2018), a função de ativação Leaky ReLU (Equação 2.2) (XU et al., 2015), `AdaptativeMaxPool3d` com a saída em cubos de lado 1 e convolução 3D com *kernels* cúbicos de lado 1 e a função sigmoide (Equação 2.3), que é utilizada para gerar números entre 0 e 1 a partir de uma entrada real, o que pode ser entendido como uma probabilidade.

$$f : \mathbb{R} \rightarrow \mathbb{R}, f(x) = \begin{cases} x & \text{se } x \geq 0 \\ 0,01x & \text{caso contrário} \end{cases} \quad (2.2)$$

$$f : \mathbb{R} \rightarrow [0, 1], f(x) = \frac{1}{1 + \exp(-x)} = \frac{\exp(x)}{1 + \exp(x)} \quad (2.3)$$

Figura 7 – Localizer loss da rede GLIA-Net.



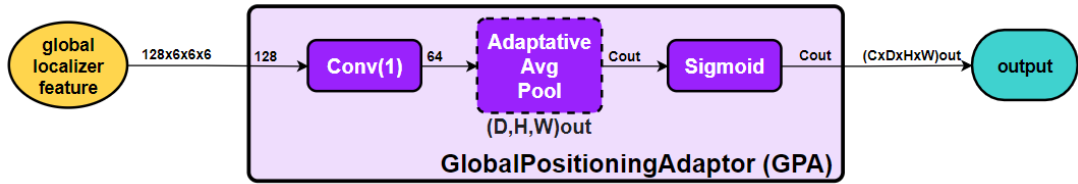
Fonte: Adaptado de (BO et al., 2021).

Nota: O número em cima das linhas corresponde ao número de canais do *feature map* representado pela linha.

Nota: 1 abaixo da operação indica que esta iguala a profundidade, a altura e a largura do *feature map* a 1.

Os *global positioning adaptors* possuem a estrutura mostrada na Figura 8. Basicamente um GPA é composto por uma operação de convolução 3D com *kernels* cúbicos de lado 1 e a função de ativação sigmoide (Equação 2.3). A operação *AdaptiveMaxPool3d* é empregada quando as dimensões altura, largura e profundidade do *feature map* de entrada e da saída do GPA são diferentes.

Figura 8 – Global positioning adaptor da rede GLIA-Net.



Fonte: Adaptado de (BO et al., 2021).

Nota: O número em cima das linhas corresponde ao número de canais do *feature map* representado pela linha.

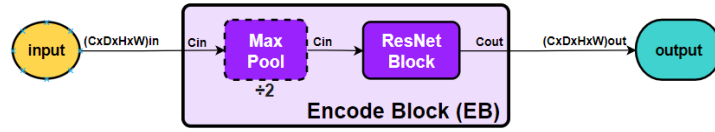
Nota: $(D, H, W)_{out}$ abaixo da operação indica que esta iguala as dimensões do *feature map* profundidade a D , a altura a H e a largura a W .

Nota: O contorno pontilhado indica que a operação só é empregada quando a mudança de tamanho de pelo menos uma das dimensões profundidade, altura e largura é necessária.

O bloco *encoder* utilizado por diversos componentes da rede GLIA-Net é constituído por uma operação *MaxPooling3d* quando for necessário reduzir com *kernel* cúbico de lado e passo iguais a 2 – uma operação que consiste em interpretar o *feature map* de entrada como cubos de lado 2 justapostos e selecionar o valor máximo de cada cubo – a dimensão do *feature map* por 2 e um bloco ResNet. Este componente é apresentado na Figura 9.

Já o bloco *decoder* é utilizado apenas na rede *local segmentation network* e é constituído por uma operação de convolução transposta 3D com *kernels* cúbicos de lado

Figura 9 – Encode block da rede GLIA-Net.



Fonte: Adaptado de (BO et al., 2021).

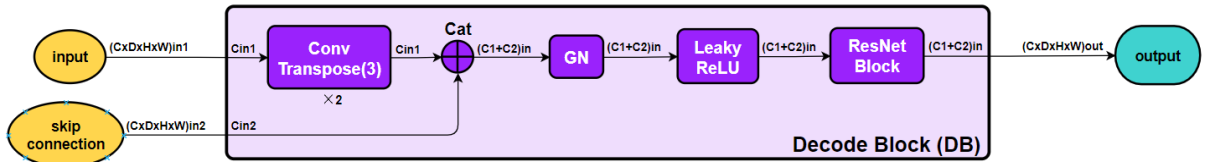
Nota: O número em cima das linhas corresponde ao número de canais do *feature map* representado pela linha.

Nota: $\div 2$ indica que a operação divide a profundidade, a altura e a largura do *feature map* por 2.

Nota: O contorno pontilhado indica que a operação só é empregada quando a mudança de tamanho de pelo menos uma das dimensões profundidade, altura e largura é necessária.

3 dobrando as dimensões altura, largura e profundidade da entrada de menor tamanho, a concatenação na dimensão dos canais do resultado da convolução transposta com a *skip connection*, a normalização por grupo (*Group Normalization*) (WU; HE, 2018), a função de ativação Leaky ReLU (Equação 2.2) (XU et al., 2015) e um bloco ResNet. Este componente é apresentado na Figura 10.

Figura 10 – Decode block da rede GLIA-Net.



Fonte: Adaptado de (BO et al., 2021).

Nota: O número em cima das linhas corresponde ao número de canais do *feature map* representado pela linha.

Nota: \oplus com *Cat* acima representa a operação de concatenação de tensores na dimensão dos canais.

Como abordado nos componentes antecessores, a microarquitetura principal utilizada pela rede GLIA-Net é o bloco ResNet. Os Blocos Residuais (*Residual Blocks*) foram desenvolvidos inicialmente como os blocos fundamentais da rede ResNet implementada para a tarefa de classificação de imagens. A ideia central por trás do Bloco Residual é formular a função que deseja-se que a rede aprenda $H(x)$ como $F(x) + x$, sendo $F(x)$ o resíduo de interesse. Esta formulação, segundo resultados experimentais, permite facilitar a otimização da rede neural em relação a formulação direta (HE et al., 2015).

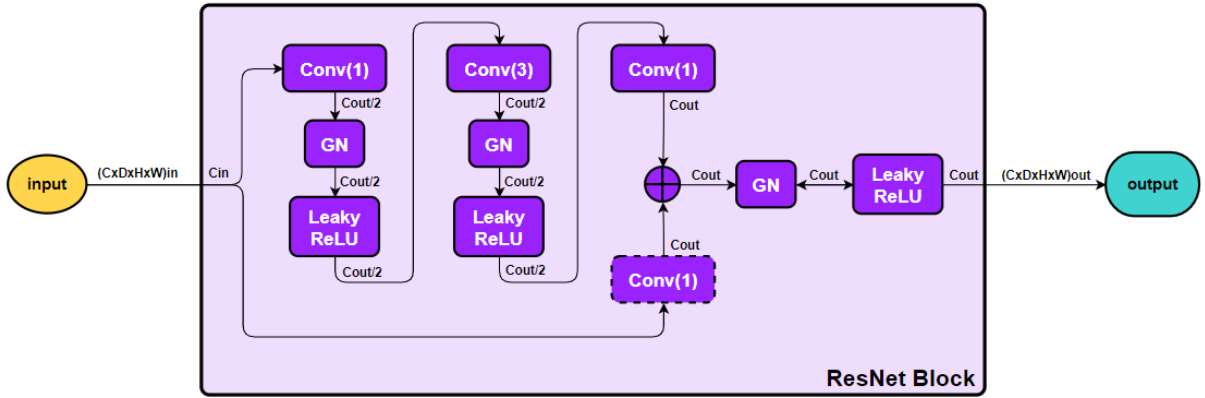
Em particular, o Bloco Residual gargalo (*bottleneck building block*), também proposto no mesmo artigo, é utilizado em GLIA-Net. Nele, emprega-se para aproximar uma função $F(x)$ três convoluções em série seguidas por uma função de ativação *Rectified Li-*

near Unit, ou ReLU (Equação 2.4). A segunda convolução possui *kernels* quadrados de lado 3, sendo a única operação do bloco que é aplicada no voxel em questão em seus vizinhos imediatos. A primeira e a última convolução possuem *kernels* quadrados de lado 1, e portanto, operam apenas na dimensão dos canais. A primeira reduz o número de canais do *feature map* para que a convolução com *kernels* de lado 3 não seja tão custosa computacionalmente e a terceira restaura o número de canais do *feature map* de entrada (HE et al., 2015).

$$f : \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}, f(x) = \begin{cases} x & \text{se } x \geq 0 \\ 0 & \text{caso contrário} \end{cases} \quad (2.4)$$

Como está exibido na Figura 11, a implementação do *bottleneck building block* na rede GLIA-Net é ligeiramente diferente. Todas as operações de convolução utilizadas no bloco possuem *kernels* cúbicos no lugar de quadrados, as convoluções obrigatórias são seguidas de uma normalização por grupo (*Group Normalization*) (WU; HE, 2018) e as funções de ativação ReLU são substituídas por Leaky ReLU (Equação 2.2) (XU et al., 2015). Além disso, caso o número de canais dos tensores de entrada e de saída do bloco sejam distintos, emprega-se uma operação de convolução com *kernels* de tamanho unitário na entrada do bloco para igualá-los.

Figura 11 – ResNet block da rede GLIA-Net.



Fonte: Adaptado de (BO et al., 2021).

Nota: O número em cima das linhas corresponde ao número de canais do *feature map* representado pela linha.

Nota: \oplus representa a operação de soma de tensores termo a termo.

Nota: O contorno pontilhado indica que a operação só é empregada quando o número de canais dos tensores de entrada e de saída são distintos.

Segundo (WU; HE, 2018), a normalização por grupo usada na rede GLIA-Net

consiste na seguinte operação:

$$\hat{x} = \frac{x - \mu}{\sigma} \quad (2.5)$$

onde μ e σ são, respectivamente, a média e o desvio padrão da entrada x . Esta operação é definida para x sendo um conjunto de canais abrangendo a largura, a altura e a profundidade de um mapa de características (*feature map*) de uma rede neural. A saída final da normalização por grupo (y) é calculada da seguinte forma a fim de evitar perdas de representatividade por causa da normalização:

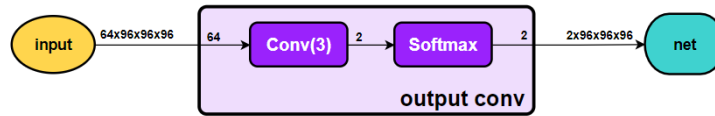
$$y = \gamma \hat{x} + \beta \quad (2.6)$$

onde γ e β são parâmetros de escala e deslocamento aprendidos no treinamento.

Como esta normalização opera independentemente do *batch*, as estatísticas μ e σ podem ser estimadas da mesma forma no treinamento e na inferência e o tamanho do *batch* pode ser dissociado à quantidade de memória exigida para executar o modelo. Ainda sim, resultados apontam que a performance de redes neurais com esta normalização é independente ao número de *batch* (WU; HE, 2018).

Finalmente, a Figura 12 apresenta o bloco *output conv*, que gera a máscara de segmentação do bloco de interesse. Este bloco é simplesmente formado por uma operação de convolução com *kernels* cúbicos de lado igual a 3 seguida por uma função de ativação *softmax* (Equação 2.7), que torna todos os valores da máscara de segmentação entre 0 e 1, sendo que a soma dos canais de um mesmo voxel resulta sempre em 1. Dessa forma, a máscara de segmentação contém uma estimativa da probabilidade de cada voxel do bloco de interesse ser classificado como aneurisma ou não.

Figura 12 – Output conv da rede GLIA-Net.



Fonte: Adaptado de (BO et al., 2021).

Nota: O número em cima das linhas corresponde ao número de canais do *feature map* representado pela linha.

$$i \in [1, N] \cap \mathbb{N}, f : \mathbb{R}^N \rightarrow [0, 1]^N, f(x_i) = \frac{\exp(x_i)}{\sum_{j=1}^N \exp(x_j)} \quad (2.7)$$

onde N é o número de elementos de um vetor \mathbf{x} e x_i é o i -ésimo termo do vetor \mathbf{x} .

Como já foi explicado, a segmentação semântica de uma imagem 3D inteira consiste na composição da segmentação semântica de cada bloco individual e que para cada bloco, o modelo GLIA-Net recebe como entrada o bloco, a posição do bloco na imagem completa e a imagem completa com dimensão reduzida e produz como saída uma máscara classificando os voxels em aneurismas ou em regiões normais e uma estimativa da probabilidade do bloco apresentar um aneurisma. A máscara de segmentação semântica é formada pela união das máscaras semânticas de probabilidade de cada bloco, utilizando a média aritmética para determinar o valor regiões sobrepostas. Para obter uma classe exata e fixa para cada voxel, utiliza-se um limiar igual a 0,5.

Um processo análogo pode ser aplicado aos *global localizer logits*, tomando uma máscara de segmentação de cada bloco no qual cada voxel é igual à probabilidade do bloco apresentar aneurismas usando apenas informações globais (*global localizer logits*). A composição desta máscara de segmentação para imagem completa representa a probabilidade de cada região da imagem apresentar um aneurisma considerando apenas a imagem completa de dimensão reduzida como entrada (*global inputs*).

2.2.2.1.1 Função de custo

Para guiar a rede neural GLIA-Net para que esta assuma valores de tal forma que a solução desempenhe melhor na tarefa desejada, segmentação de aneurismas, com o passar do treinamento é necessário de uma função de custo.

Como já foi explicado, a rede GLIA-Net classifica voxels e blocos em duas classes: aneurisma ou normal. Assim, todos os componentes da função de custo são para a tarefa de classificação ou segmentação binária. O artigo (BO et al., 2021) detalha que a função de custo total (L_{Total}) é descrita pela seguinte equação:

$$L_{Total} = \omega_{Global} L_{Global} + \omega_{Local} L_{Local} \quad (2.8)$$

onde ω_{Global} e ω_{Local} são pesos ponderadores e L_{Global} e L_{Local} são as funções de custo global e local, respectivamente.

Em particular, para a saída da rede *global localization network*, que é um par de números reais, a função de custo global (L_{Global}) é aplicada como *softmax cross-entropy loss*, descrita por:

$$L_{Global} = -z \ln \hat{z} - (1 - z) \ln (1 - \hat{z}) \quad (2.9)$$

onde z é o valor (0 ou 1) da classe real do bloco (*patch*) e \hat{z} é a predição da classe dada pelo modelo. \hat{z} é calculado pela aplicação da função *softmax* (Equação 2.7) sobre

os *global localizer logits*, resultando na probabilidade do bloco apresentar um aneurisma \hat{z} ou não $1 - \hat{z}$.

Enquanto para a rede *local segmentation network*, a função de custo local (L_{Local}) é uma função de custo exponencial logarítmica (*exponential logarithmic loss*), expressa por:

$$L_{Local} = \omega_{Dice} L_{Dice} + \omega_{Cross} L_{Cross} \quad (2.10)$$

onde ω_{Dice} e ω_{Cross} são pesos ponderadores, L_{Dice} e L_{Cross} são funções de custo de dice (*dice loss*) e de entropia cruzada (cross-entropy loss), respectivamente.

A função de custo de dice visa minimizar a diferença de forma da máscara de segmentação predita e a anotada e é descrita pelo seguinte valor esperado:

$$L_{Dice} = \mathbb{E} \left(-\ln \frac{2y\hat{y} + \varepsilon}{y + \hat{y} + \varepsilon} \right)^{\gamma_{Dice}} \quad (2.11)$$

onde $\mathbb{E}(\cdot)$ é a função que calcula a média de do valor \cdot iterando sobre todos os voxels do bloco (*patch*) que está sendo segmentado, y e \hat{y} são, respectivamente, a classe anotada e a probabilidade de representar um aneurisma de cada voxel, γ_{Dice} é um parâmetro que controla a não-linearidade da função de custo de dice e ε é uma constante para suavizar a função.

Para finalizar, a função de custo de entropia cruzada (L_{Cross}) é descrita por:

$$L_{Cross} = \mathbb{E} (\omega_p (-y \ln \hat{y} - (1 - y) \ln (1 - \hat{y}))^{\omega_{Cross}}) \quad (2.12)$$

onde $\mathbb{E}(\cdot)$ é a função que calcula a média de do valor \cdot iterando sobre todos os voxels do bloco (*patch*) que está sendo segmentado, y e \hat{y} são, respectivamente, a classe anotada e a probabilidade de representar um aneurisma de cada voxel, γ_{Cross} é um parâmetro que controla a não-linearidade da função de custo de entropia cruzada e ω_p é um peso ponderador que desconta o valor de voxels próximos a borda do aneurisma e mantém o valor de pixels internos e externos à aneurismas. Este desconto é importante, pois há uma alta variabilidade de anotações entre especialistas (SUINESIAPUTRA et al., 2015).

2.3 Transferência de aprendizado

Seguindo o artigo (PAN; YANG, 2010), esta seção define os conceitos de **domínio** (*domain*) e **tarefa** (*tasks*), que são utilizados para apresentar o conceito de **transferência por aprendizado** (*transfer learning*).

O domínio consiste de dois componentes: um espaço de características χ e uma distribuição de probabilidades marginais $P(X)$, onde $X = x_1, x_2, \dots, x_n \in \chi$. Assim pode-se definir um domínio como $D = \{\chi, P(X)\}$.

Para um domínio específico $D = \{\chi, P(X)\}$, uma tarefa consiste de dois componentes: um espaço de anotações v e uma função preditiva objetiva $f(\cdot)$, que não é conhecida mas pode ser aprendida por meio de dados de treinamento, formados por pares $\{x_i, y_i\}$ onde $x_i \in X$ e $y_i \in v$. Dessa forma, uma tarefa pode ser descrita por $\tau = \{v, f(\cdot)\}$. Para um novo x , a função $f(\cdot)$ pode ser utilizada para prever a sua anotação correspondente, tentando aproximar a anotação y associada a x com $f(x)$. A partir de uma perspectiva probabilística, $f(x)$ pode ser escrito como $P(y|x)$.

No cenário de segmentação de aneurismas, o espaço de características χ é o conjunto de imagens dos exames (RM, ATC ou DSA), X é um conjunto de imagens (RM, ATC ou DSA) de um único exame, x_i é cada voxel dos exames, o espaço de anotações v é o conjunto de máscaras de segmentação que anotam se um voxel dos exames corresponde a um aneurisma ou a uma região normal e a função $f(x)$ é a função que realiza a segmentação dos exames em aneurismas e em regiões normais.

No contexto de transferência de aprendizado, utiliza-se o domínio D_S e a tarefa τ_S fontes e o domínio D_T e a tarefa τ_T alvos. Assim, para o domínio fonte, temos: $D_S = \{(x_{S1}, y_{S1}), \dots, (x_{Sn_S}, y_{Sn_S})\}$ com $x_{Si} \in \chi_S$ sendo uma instância de dados fonte e $y_{Si} \in v_S$ a sua anotação correspondente; e para o domínio alvo, temos: $D_T = \{(x_{T1}, y_{T1}), \dots, (x_{Tn_T}, y_{Tn_T})\}$ com $x_{Ti} \in \chi_T$ sendo uma instância de dados alvo e $y_{Ti} \in v_T$ a sua anotação correspondente. n_S e n_T são, respectivamente, o número de instâncias da fonte e o número de instâncias alvo, no qual geralmente $0 < n_T \ll n_S$.

Com base nas definições de domínio e tarefa, pode-se definir transferência de aprendizado:

Dado um domínio fonte D_S , uma tarefa fonte T_S , um domínio alvo D_T e uma tarefa alvo T_T , a transferência de aprendizado visa melhorar o aprendizado da função preditiva alvo $f_T(\cdot)$ em D_T usando o conhecimento em D_S e T_S , onde $D_S \neq D_T$ ou $T_S \neq T_T$.

Voltando ao cenário de segmentação de aneurismas, particularmente se apenas a modalidade dos exames é alterada, o que caracterizaria em uma diferença entre os espaços de características fonte e alvo, temos $D_S \neq D_T$ e $T_S = T_T$.

Segundo (TAN et al., 2018), a **transferência de aprendizado profundo** (*deep transfer learning*) pode ser definida como um caso particular de transferência de aprendizado:

Dado uma tarefa de transferência de aprendizado definida por $(D_S, T_S, D_T, T_T, f_T(\cdot))$, define-se uma tarefa de transferência de aprendizado profundo o caso particular no qual $f_T(\cdot)$ é uma função não-linear descrita por uma rede neural profunda.

Este artigo também propõe a classificação da transferência de aprendizado profundo em quatro categorias: baseada em instâncias (*instance-based deep transfer learning*), baseada em mapeamento (*mapping-based deep transfer learning*), baseada em adversários (*adversarial-based deep transfer learning*) e baseada em rede (*network-based deep transfer learning*).

Na **transferência de aprendizado profundo baseada em instâncias**, instâncias parciais do domínio fonte são utilizadas no conjunto de dados de treinamento do domínio alvo, utilizando um peso particular.

Na **transferência de aprendizado profundo baseada em mapeamento**, os dados dos domínios fonte e alvo são mapeados em um novo espaço de dados. Este novo domínio é utilizado na rede neural profunda.

Por último, a **transferência de aprendizado profunda baseada em adversários** se refere ao uso de instâncias adversárias devidamente escolhidas de modo a encontrar representações aplicáveis tanto no domínio fonte como no domínio alvo.

A **transferência de aprendizado profunda baseada em rede** se refere ao reuso parcial de uma rede neural pré-treinada no domínio fonte, incluindo a sua estrutura e a conexão entre seus parâmetros, como uma parte integral da rede neural profunda utilizada no domínio alvo. Nesta abordagem, trata-se as primeiras camadas da rede neural como extratores gerais e versáteis de características e as últimas camadas como estimadores particulares de uma tarefa específica em um domínio específico.

No contexto particular das redes de segmentação semântica que seguem a estrutura básica do modelo 3D U-Net, o artigo (KARIMI; WARFIELD; GHOLIPOUR, 2020) apresenta indícios de que os parâmetros das camadas mais próximas à saída da rede neural são os mais alterados no treinamento da rede. Os autores compararam o treinamento de uma mesma rede neural partindo de dois cenários: de parâmetros inicializados aleatoriamente e de parâmetros pré-treinados em outras tarefas de segmentação, variando idade do paciente e órgão segmentado. Como métrica, foi usada uma média ponderada da correlação entre os parâmetros de cada camada das duas redes neurais treinadas por meio da Análise por Correlação Canônica (ACC) (BACH; JORDAN, 2005). Os resultados indicam que os parâmetros dos *encoders* aleatoriamente inicializados e pré-treinados mudam menos e são mais semelhantes que os parâmetros aleatoriamente inicializados e pré-treinados dos *decoders*. A explicação proposta para este fenômeno é que os *decoders* são as últimas camadas e, portanto, pequenas mudanças nas camadas precedentes impactam na entrada desses componentes. Os autores interpretam que os *encoders* extraem mapas de características genéricas, enquanto os *decoders* extraem informações particulares para a tarefa específica do treinamento fino.

A partir desta interpretação, questionou-se a necessidade de treinar os *encoders* em

uma rede do tipo 3D U-Net para segmentação. Para investigar esta questão, os autores de (KARIMI; WARFIELD; GHOLIPOUR, 2020) realizaram o treinamento de uma rede de segmentação do tipo 3D U-Net inicializada com valores aleatórios de duas formas distintas: (1) congelando os parâmetros dos *encoders* e treinando apenas os *decoders* e (2) treinando a rede inteira. Com o congelamento, o tempo de otimização dos parâmetros da rede caiu em 40% e a queda de performance foi marginal. O experimento foi realizado em duas tarefas de segmentação de imagens médicas e os resultados constam na Tabela 4.

Tabela 4 – Resultados do treinamento de uma rede de segmentação do tipo 3D U-Net treinando apenas os *decoders* e treinando a rede inteira nos bancos de dados *Liver-CT* e *CP-younger fetus*.

Banco de dados	Congelamento dos <i>encoders</i>	DSC
Liver-CT	Não	0,967
Liver-CT	Sim	0,940
CP-younger fetus	Não	0,896
CP-younger fetus	Sim	0,884

Fonte: Adaptado de (KARIMI; WARFIELD; GHOLIPOUR, 2020).

Nota: **DSC** representa o coeficiente de similaridade de Sorensen-Dice.

Assim, mostram que o treinamento apenas dos *decoders* da rede permite um aumento na velocidade de treinamento do modelo sem perdas consideráveis na performance.

O artigo também estuda o efeito da transferência de aprendizado no treinamento da rede de segmentação semântica no cenário de mudança de modalidade, de ressonância magnética para tomografia computadorizada (ambas anatômicas), de mudança de idade, de mudança de tarefa e de mudança no protocolo de aquisição. No caso particular de mudança de modalidade, a transferência de aprendizado gera um aumento marginal na performance, conforme a Tabela 5, e uma redução de considerável no tempo de convergência do algoritmo (KARIMI; WARFIELD; GHOLIPOUR, 2020).

Tabela 5 – Performance de uma rede de segmentação de fígado do tipo 3D U-Net em banco de dados de teste de imagens TC anotadas partindo de parâmetros aleatoriamente inicializados (R.I) e partindo de parâmetros de um modelo pré-treinado nos bancos de dados de imagens de RM anotadas.

Quantidade de dados de treino	Inicialização	DSC	HD95 [mm]
15	R.I.	$0,97 \pm 0,01$	$5,07 \pm 1,94$
15	R.I.	$0,97 \pm 0,01$	$4,75 \pm 1,81$
6	T.L.	$0,95 \pm 0,01$	$5,47 \pm 2,00$
6	T.L.	$0,96 \pm 0,01$	$5,25 \pm 2,09$

Fonte: Adaptado de (KARIMI; WARFIELD; GHOLIPOUR, 2020).

Nota: **DSC** representa o coeficiente de similaridade de Sorensen-Dice e **HD95** representa a distância de Hausdorff considerando o percentil 95 das menores distâncias.

Nota: Os bancos de dados empregados são disponibilizados no desafio CHAOS (KAVUR et al., 2021).

3 Materiais e Métodos

Para desenvolver a solução para a tarefa de segmentação de imagens médicas em aneurismas e regiões saudáveis, este trabalho usa o banco de dados aberto do desafio ADAM (TIMMINS et al., 2020). Este banco de dados aberto contém 113 exames de ressonância magnética da cabeça anotados e é destinado ao desenvolvimento de soluções de detecção e segmentação de aneurismas pelos participantes.

Cada exame contém uma imagem 3D de TOF angio-RM e uma imagem 3D suplementar de angio-RM, podendo ser ponderada em T1, em T2 ou ser do tipo FLAIR. Está disponível também a versão da imagem 3D suplementar alinhada com a imagem 3D de TOF angio-RM de tal forma que os eixos x, y e z representem, respectivamente, as direções direita-esquerda, frente-trás e baixo-cima. Os parâmetros da operação de alinhamento bem como os parâmetros de captura de imagem são fornecidos. Quanto às anotações, uma máscara 3D classificando cada voxel da imagem de TOF angio-RM e um arquivo texto contendo as posições e o tamanho em mm do aneurisma são fornecidos,

Utiliza-se uma abordagem de aprendizado profundo para realizar a segmentação automática de aneurismas destes exames de ressonância magnética. Visando acelerar o treinamento, este é executado na plataforma Google Colab, que dispõe de GPUs gratuitas K80 e pagas T4 e P100. O serviço Google Colab Pro foi assinado para se beneficiar de GPUs com maior capacidade de processamento e mais memória RAM e do disco. Em particular, a GPU P100 de 16 GB de RAM foi empregada na execução da rede neural.

A plataforma Colab disponibiliza a GPU P100 da Nvidia, 25,46 GB de memória RAM e 147,15 GB de memória do disco rígido, sendo que o sistema já ocupa 38,99 GB. Assim, o ambiente fornece espaço no disco rígido suficiente para salvar o banco de dados aberto ADAM, que ocupam 30 GB quando compactado e menos de 60 GB quando descompactado.

De acordo com revisão bibliográfica, a arquitetura GLIA-Net é a mais promissora para a segmentação de aneurismas intracranianos, superando outras arquiteturas do estado da arte e mostrando o seu potencial para melhorar a qualidade e agilizar a detecção de aneurismas em diagnóstico clínico. A implementação original, disponível em <<https://github.com/MeteorsHub/GLIA-Net>>, foi implementada na linguagem de programação Python e usa a biblioteca de aprendizado profundo PyTorch. O trabalho está atribuído à licença de acesso aberto CC BY-NC-ND.

Como o projeto original de GLIA-Net aplica a segmentação de imagens de ATC e o nosso objetivo é segmentar imagens a partir de imagens de angio-RM, é necessário adaptar o projeto original para esta nova modalidade. Assim, o proprietário do projeto

GLIA-Net foi contactado e este permitiu a extensão do projeto a partir de um *fork*, disponível em <<https://github.com/gabrielriqui3ti/GLIA-Net>>.

3.1 Análise de dados do desafio ADAM

Para adaptar a solução GLIA-Net, desenvolvida para segmentar imagens de exames de ATC, para segmentar imagens de exames de TOF angio-RM, é necessário compreender o pré-processamento da entrada e os dados fornecidos pelos bancos de dados de ressonância magnética e de tomografia computadorizada.

O banco de dados disponibilizado pelo desafio ADAM, contém 113 dados que seguem a seguinte estrutura de arquivos:

```

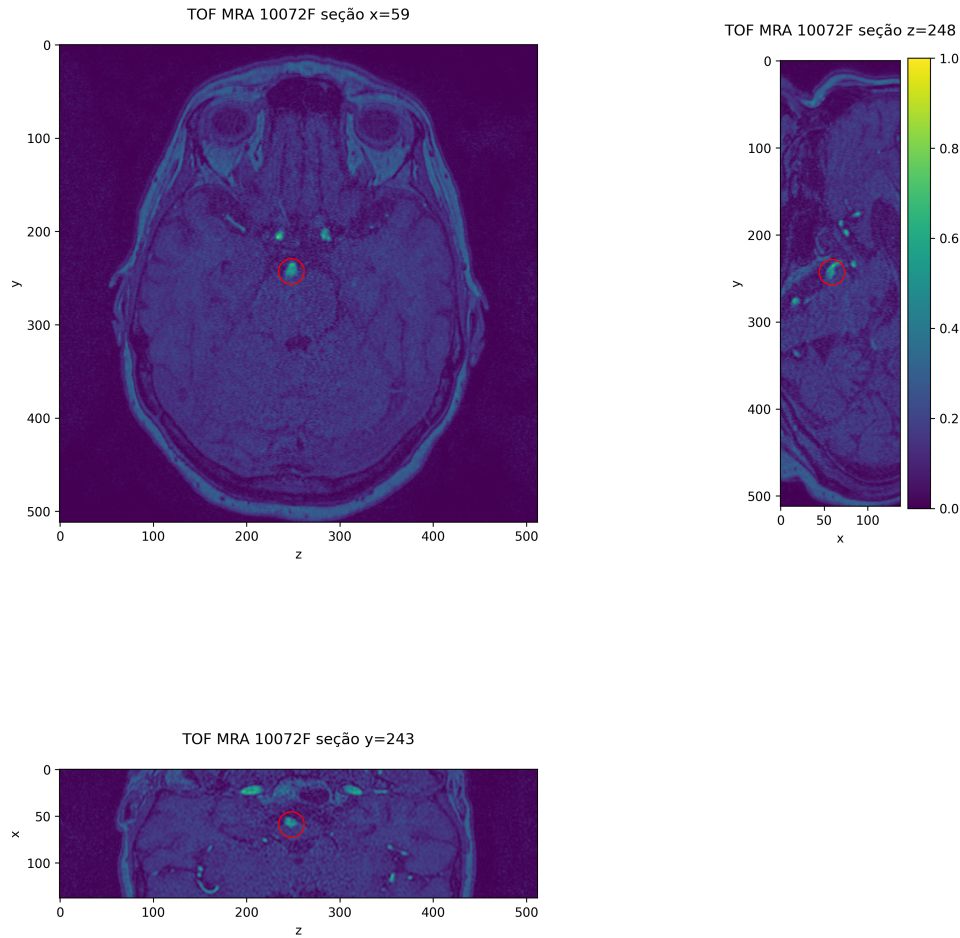
dato
├── orig
│   ├── reg_struct_to_TOF.txt
│   ├── ScanParams_struct.txt
│   ├── ScanParams_TOF.txt
│   ├── struct.nii.gz
│   ├── struct_aligned.nii.gz
│   └── TOF.nii.gz
├── pre
│   ├── struct.nii.gz
│   ├── struct_aligned.nii.gz
│   └── TOF.nii.gz
├── aneurysms.nii.gz
└── location.txt
```

Na pasta `orig` temos os arquivos de exames de imagem 3D originais e as informações para gerar os arquivos pré-processados, enquanto na pasta `pre` temos os exames de imagem 3D pré-processados. Todas os arquivos com extensão `.nii.gz` estão no formato de dados comprimido e padronizado NIfTI, específico para imagens de neurociência.

`TOF.nii.gz` é um arquivo que contém uma imagem 3D de um exame de TOF angio-RM. Para todos os dados, este arquivo assume o mesmo alinhamento, conforme a Figura 13.

`struct.nii.gz` e `struct_aligned.nii.gz` são arquivos que contêm uma imagem 3D de um exame de IRM, podendo este ser de uma das três modalidades: ponderada em T1, ponderada em T2 ou FLAIR, variando de dado para dado. Note que estas modalidades são exames anatômicos e não exames de angiografia, e portanto, não são usados para ressaltar a estrutura vascular cerebral, mas sim para diferenciar os tecidos da cabeça. O arquivo `struct_aligned.nii.gz` difere do arquivo `struct.nii.gz` por

Figura 13 – Seções de imagem 3D do exame TOF angio-RM do dado 10072F do desafio ADAM



Fonte: Autor.

Nota: O círculo vermelho artificialmente sobreposto à imagem localiza o aneurisma não tratado nem rompido e a posição do seu centro foi obtida por meio do arquivo de localização do mesmo dado.

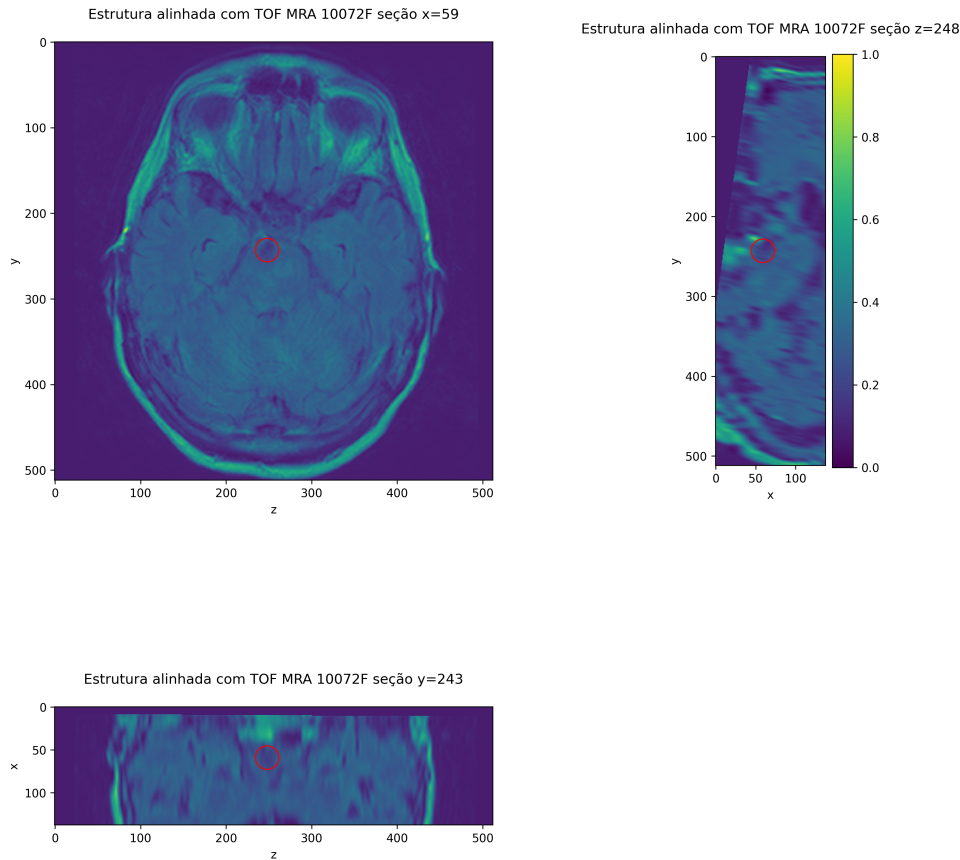
Nota: Os valores da imagem foram normalizados linearmente de tal forma que os valores mínimo e máximo da imagem original passam a valer 0 e 1 na imagem normalizada.

apresentar o mesmo alinhamento e a mesma escala que a imagem 3D do exame TOF angio-RM, conforme a Figura 14.

Esta transformação afina está documentada no arquivo `reg_struct_to_TOF.txt` e provoca artefatos na imagem 3D gerada, como regiões nulas devido ao limite da imagem original e à perda de foco devido à mudança de escala, como pode-se observar na Figura 14.

A configuração básica de um equipamento de ressonância magnética exige três parâmetros: a intensidade do campo magnético, o tempo de repetição e o tempo de eco. Estas quatro informações estão armazenadas nos arquivos `ScanParams_TOF.txt` para imagens de exames de TOF angio-RM e `ScanParams_struct.txt` para imagens de exames de ressonância magnética do tipo T1, T2 ou FLAIR.

Figura 14 – Seções de imagem 3D do exame FLAIR do dado 10072F do desafio ADAM



Fonte: Autor.

Nota: O círculo vermelho artificialmente sobreposto à imagem localiza o aneurisma não tratado nem rompido e a posição do seu centro foi obtida por meio do arquivo de localização do mesmo dado.

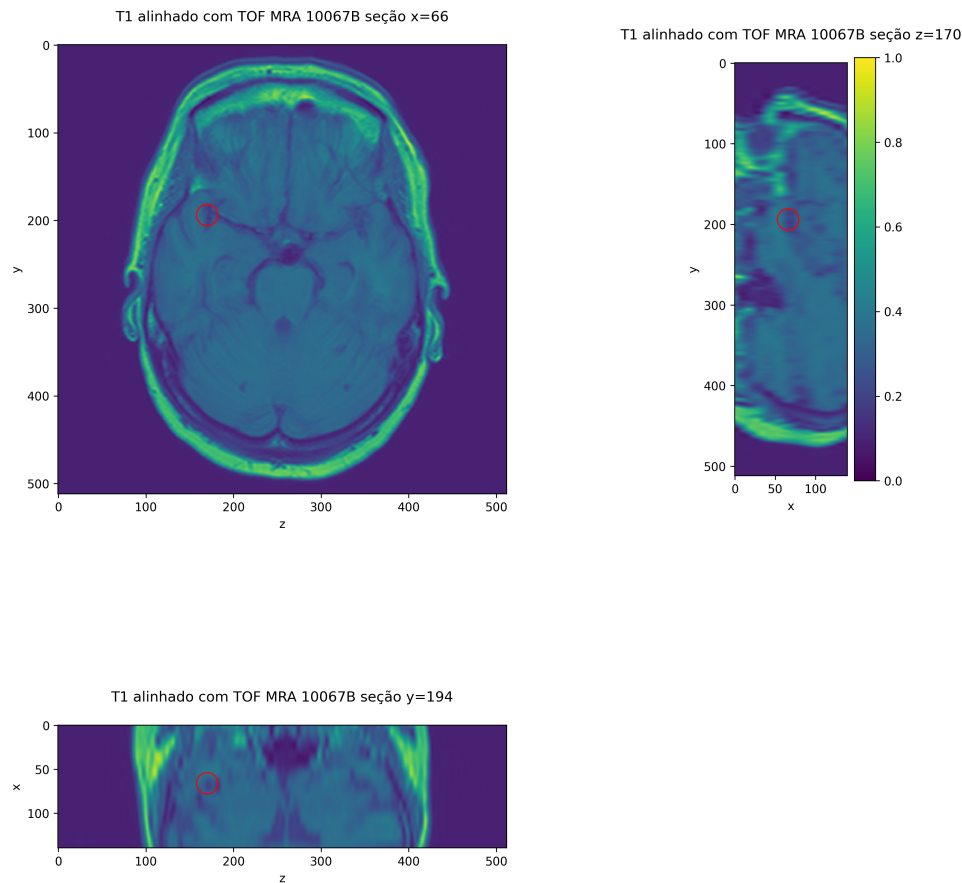
Nota: Os valores da imagem foram normalizados linearmente de tal forma que os valores mínimo e máximo da imagem original passam a valer 0 e 1 na imagem normalizada.

Como as imagens dos arquivos que começam por **struct** podem ser de três modalidades distintas, é importante estudá-las para avaliar se podemos utilizá-las indiscriminadamente na solução, se alguma adaptação específica para cada modalidade é necessária ou se é melhor descartar esta informação.

As Figuras 14, 15 e 16 contêm seções com aneurisma cerebral de exames das três modalidades FLAIR, T1 e T2 respectivamente. Nas três modalidades, o aneurisma aparece como uma região de baixa intensidade e os diferentes tecidos da cabeça apresentam mais detalhes e menos ruído quando comparado à imagem do exame de TOF angio-RM (Figura 13), o que sugere que de fato estas três imagens possam ser tratadas de forma indistinta e que possam fornecer informações relevantes da anatomia da cabeça para a segmentação de aneurismas. Todavia, os tecidos apresentam intensidades relativas diferentes em cada modalidade e uso destas imagens no treinamento pode introduzir

correlações artificiais entre as modalidades da imagem e as anotações. Para exemplificar o problema, apenas um dos 113 pares de exames de imagem é composto por imagens ponderadas em T1, e portanto, caso as imagens dos arquivos `struct_aligned.nii.gz` sejam usadas como entrada do sistema, a rede neural teria apenas um exemplo de imagem de ressonância magnética ponderada em T1 com anotação para aprender a lidar com uma entrada desta modalidade.

Figura 15 – Seções de imagem 3D do exame de IRM ponderado em T1 do dado 10067B do desafio ADAM



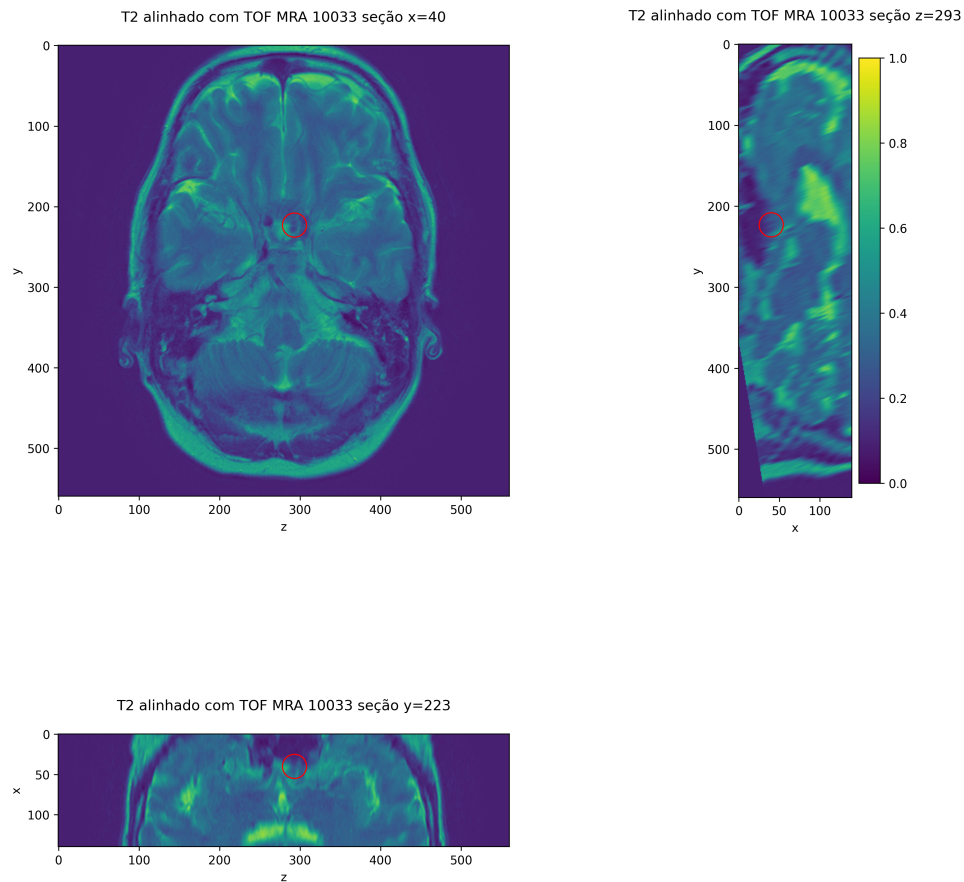
Fonte: Autor.

Nota: O círculo vermelho artificialmente sobreposto à imagem localiza o aneurisma não tratado nem rompido e a posição do seu centro foi obtida por meio do arquivo de localização do mesmo dado.

Nota: Os valores da imagem foram normalizados linearmente de tal forma que os valores mínimo e máximo da imagem original passam a valer 0 e 1 na imagem normalizada.

`aneurysms.nii.gz` é um arquivo que contém uma imagem 3D que representa a anotação de segmentação da imagem do exame TOF angio-RM, ou seja, cada voxel desta imagem contém um valor qualitativo que indica se o voxel representa uma região de um aneurisma ou não, conforme mostrado na Figura 17. Esta imagem também é chamada de máscara de segmentação e a informação qualitativa da máscara consta no Quadro 6.

Figura 16 – Seções de imagem 3D do exame de IRM ponderado em T2 do dado 10033 do desafio ADAM



Fonte: Autor.

Nota: O círculo vermelho artificialmente sobreposto à imagem localiza o aneurisma não tratado nem rompido e a posição do seu centro foi obtida por meio do arquivo de localização do mesmo dado.

Nota: Os valores da imagem foram normalizados linearmente de tal forma que os valores mínimo e máximo da imagem original passam a valer 0 e 1 na imagem normalizada.

Quadro 6: Legenda da máscara de segmentação do desafio ADAM

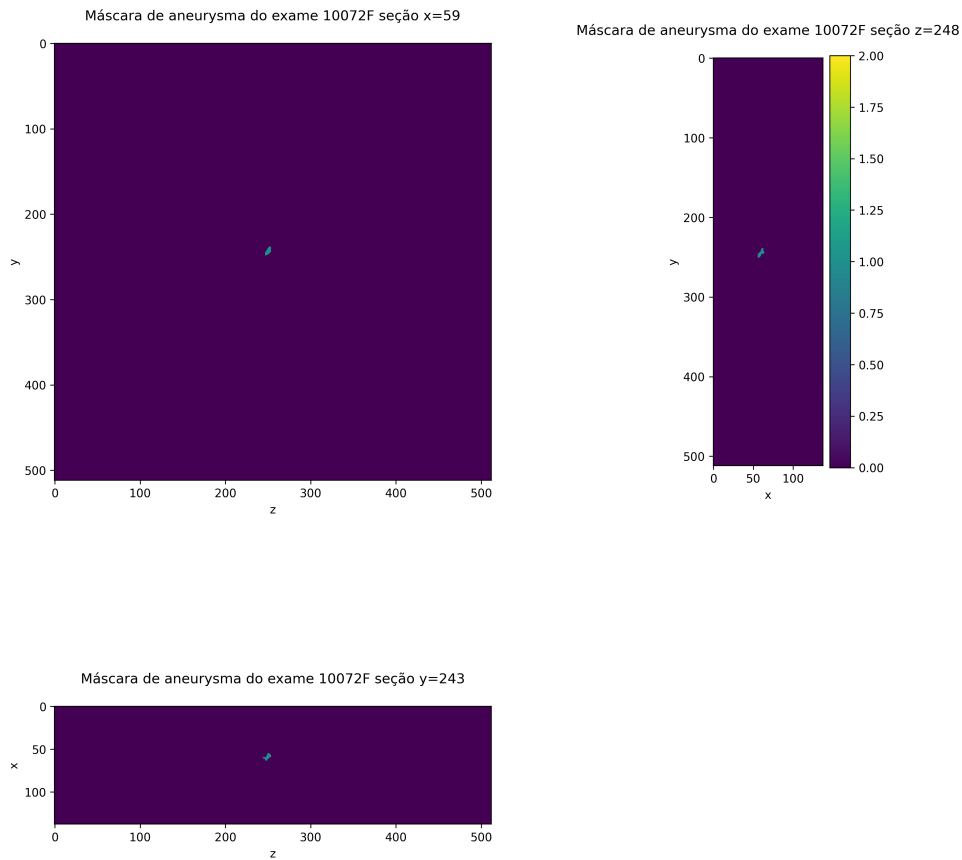
Valor	Descrição
0	Normal
1	Aneurisma não rompido nem tratado
2	Aneurisma tratado ou rompido

Fonte: Adaptado de (TIMMINS et al., 2020)

`location.txt` é um arquivo de texto que contém a posição em pixels do centro de todos os aneurismas não tratados nem rompidos nos exames de imagem e o raio tal que todos os pontos do aneurisma estão dentro da esfera com este raio e o centro anotado.

As informações do arquivo `location.txt` não podem ser utilizadas diretamente pelo modelo GLIA-Net. Entretanto, estes arquivo revelam informações referentes ao ta-

Figura 17 – Seções de máscara 3D do dado 10072F do desafio ADAM. Consulte o Quadro 6 para entender o mapa de cores.



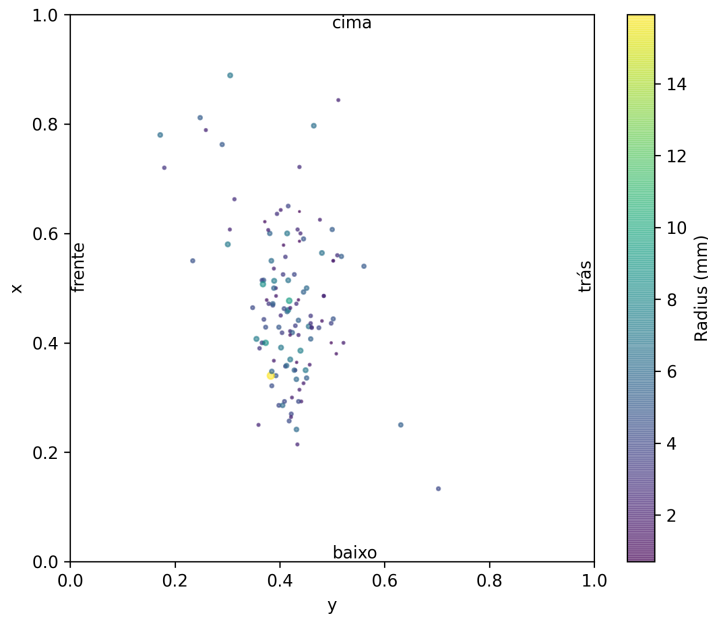
Fonte: Autor.

manho e à localização dos aneurismas já obtidas do banco de dados ADAM, que podem ser usados para sintetizar informações sobre o banco de dados completo. Utilizando estes arquivos e normalizando a informação de posição contida neles para que ela fique entre 0 e 1 de acordo com o tamanho da imagem, expõem-se a distribuição e o tamanho dos aneurismas não tratados nem rompidos no banco de dados nas Figuras 18, 19 e 20 e na Tabela 6.

A partir destas figuras, pode-se notar que os aneurismas não são uniformemente distribuídos na cabeça e há regiões da cabeça que concentram esta patologia. Logo, a exploração da localização de cada bloco da imagem pela rede GLIA-Net pode eliminar falsos positivos em regiões pouco prováveis de aparecerem aneurismas e eliminar falso-negativos em regiões que a patologia se concentra, melhorando a performance da segmentação em relação ao uso exclusivo de blocos locais, como é feito em redes neurais análogas a U-Net.

A Tabela 6 e o histograma do raio da Figura 21 revelam que, exceto o aneurisma anômalo de 15 mm de raio, os aneurismas não tratados nem rompidos possuem raios na mesma ordem de grandeza e que o eixo y (frente-trás) é o que mais concentra aneurismas já que este possui o menor desvio-padrão. É importante que os aneurismas possuam

Figura 18 – Projeção da localização de todos os aneurismas não tratados nem rompidos do banco de dados ADAM. Os eixos x e y correspondem, respectivamente, aos eixos da direita à esquerda do paciente e de frente para trás do paciente.



Fonte: Autor.

Tabela 6 – Estatísticas de localização e de tamanho dos 125 aneurismas não tratados nem rompidos do banco de dados disponibilizado aos participantes do desafio ADAM.

Estatísticas	x	y	z	Raio (mm)
Média	0,471	0,417	0,489	4,07
Desvio-padrão	0,140	0,070	0,109	2,20
Mínimo	0,133	0,172	0,145	0,70
25%	0,380	0,387	0,424	2,27
50%	0,457	0,420	0,492	3,96
75%	0,550	0,445	0,561	5,45
Máximo	0,889	0,703	0,700	15,92

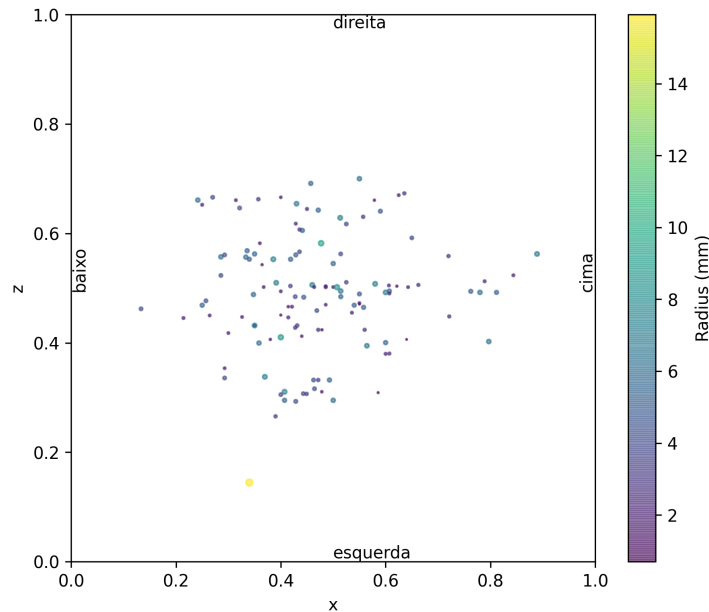
Fonte: Autor.

Nota: **x**, **y** e **z** correspondem à posição dos aneurismas nos respectivos eixos direita-esquerda, frente-trás e baixo-cima. O raio é corresponde ao raio da menor esfera centrada no baricentro do aneurisma que engloba todos os seus voxels.

tamanhos semelhantes, pois aneurismas maiores são tratados com um peso maior já que a função de custo local atua no nível dos voxels.

Ainda que os aneurismas possuam tamanhos em milímetros na mesma ordem de grandeza, é possível que eles apresentem tamanhos em ordens de grandezas diferentes se o espaçamento dos voxels for diferente. Observando a Tabela 7, pode-se notar que o espaçamento dos voxels não é constante ao longo do banco de dados e o espaçamento

Figura 19 – Projeção da localização de todos os aneurismas não tratados nem rompidos do banco de dados ADAM. Os eixos x e z correspondem, respectivamente, aos eixos da direita à esquerda do paciente e de baixo para cima do paciente.



Fonte: Autor.

em cada dimensão de uma imagem pode ser mais do que o dobro do espaçamento de outra imagem, comparando os espaçamentos mínimo e máximo em cada dimensão. Uma medida de mitigação possível para este problema é a fixação do espaçamento a um valor representativo do banco de dados, como a mediana do espaçamento em cada dimensão.

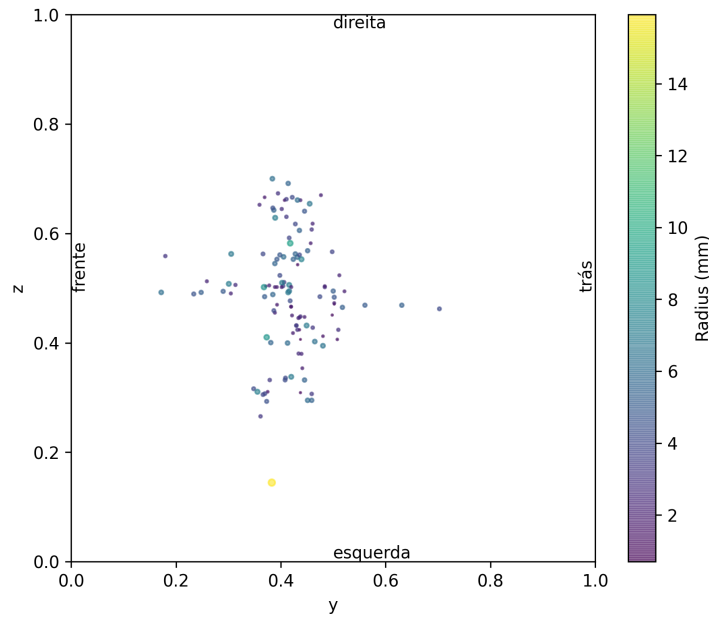
Tabela 7 – Estatísticas referentes ao espaçamento dos 113 pares de imagens disponíveis para treinamento e validação no desafio ADAM.

Estatísticas	Largura	Altura	Profundidade
Média	0,352362	0,352362	0,544688
Desvio-padrão	0,056936	0,056936	0,092424
Mínimo	0,195312	0,195312	0,400000
25%	0,312500	0,312500	0,499999
50%	0,357143	0,357143	0,500000
75%	0,390625	0,390625	0,550000
Máximo	0,585938	0,585938	1,000000

Fonte: Autor.

A fixação do espaçamento das imagens do banco de dados é realizada através da mudança no tamanho das imagens. Para viabilizar a operação, é necessário que estas imagens tenham um tamanho suficientemente pequeno para o processamento computacional. As estatísticas referentes ao tamanho das imagens médicas são mostradas na Tabela 8, todas as imagens foram capazes de serem processadas pelos recursos computacionais disponíveis.

Figura 20 – Projeção da localização de todos os aneurismas não tratados nem rompidos do banco de dados ADAM. Os eixos y e z correspondem, respectivamente, aos eixos de frente para trás do paciente e de baixo para cima do paciente.



Fonte: Autor.

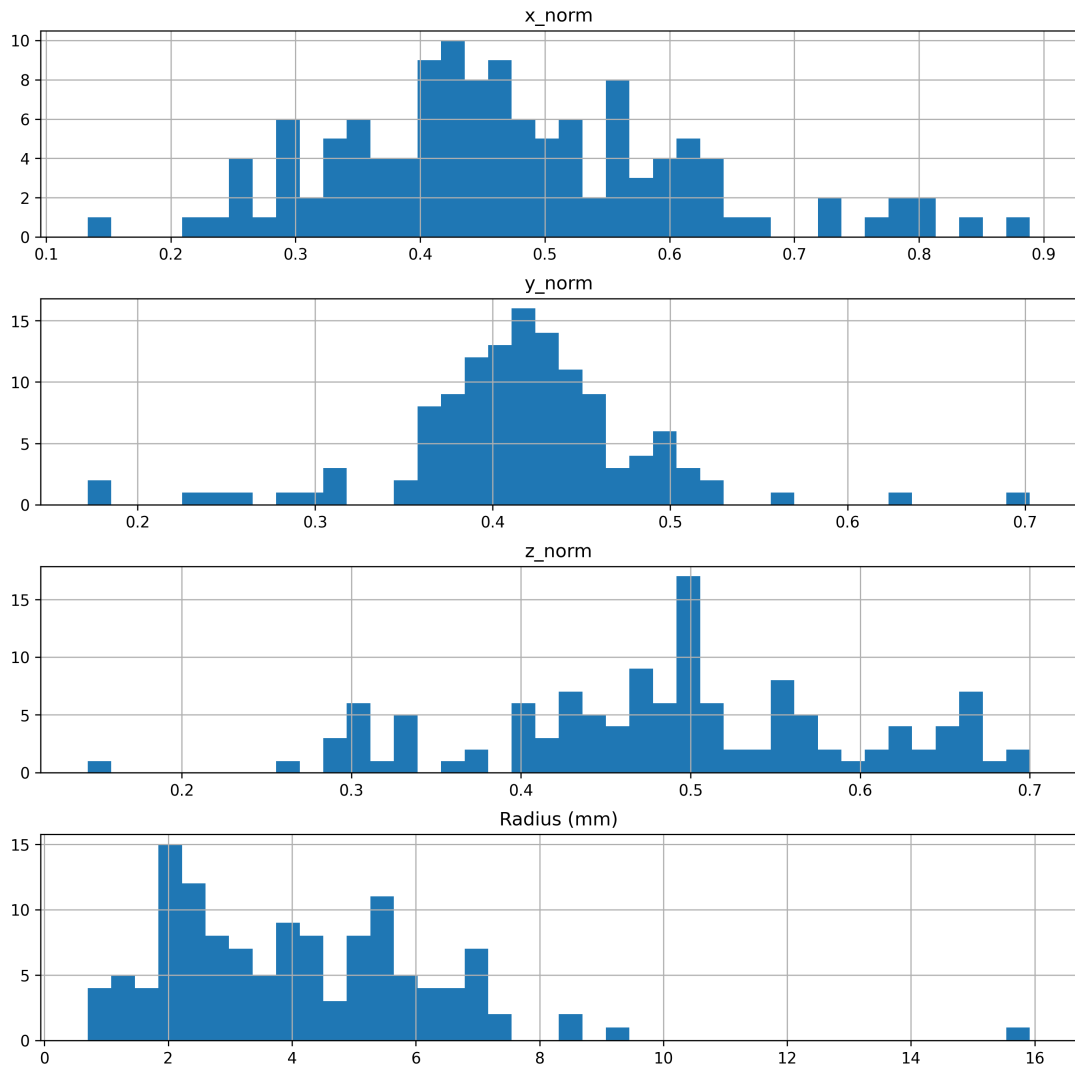
Tabela 8 – Estatísticas referentes ao tamanho dos 113 pares de imagens disponíveis para treinamento e validação no desafio ADAM.

Estatísticas	Largura	Altura	Profundidade
Média	556,9	556,9	131,9
Desvio-padrão	141,0	141,0	21,2
Mínimo	256	256	64
25%	512	512	138
50%	512	512	140
75%	560	560	140
Máximo	1024	1024	180

Fonte: Autor.

O banco de dados ADAM disponibiliza também a marcação de regiões de aneurismas tratados ou rompidos, como apresentado no Quadro 6 e permite o questionamento de como tratar esta nova classe. Há três opções: adaptar a solução para três classes independentes, tratar os voxels desta classe como voxels de aneurismas não tratados nem rompidos ou tratá-los como uma região normal da imagem. Para avaliar a melhor opção, calculou-se o volume em voxels de cada região sem mudar o espaçamento e o seu percentual, que constam na Tabela 9. Pela tabela, o volume de regiões normais é ordens de grandeza maior que o volume de regiões com aneurismas tratados ou rompidos e que este é ordens de grandeza maior que o volume de regiões com aneurismas não tratados nem rompidos. Logo, a classe 2 não pode ser tratada da mesma forma que a classe que deseja-

Figura 21 – Histograma da localização e do tamanho dos 125 aneurismas não tratados nem rompidos do banco de dados disponibilizado aos participantes do desafio ADAM.



Fonte: Autor.

se segmentar no desafio (1) pela diferença na quantidade de ambas as classes, a adaptação do projeto GLIA-Net para a segmentação de três classes é uma tarefa trabalhosa e que levanta questionamentos sobre o balanceamento das três classes. Portanto, a solução mais simples e que parece minimizar os efeitos desta terceira classe no treinamento da solução é renomear os voxels anotados com 2, de 2 para 0.

No projeto original GLIA-Net, as imagens de angiografia por tomografia computadorizada passavam por uma etapa de normalização antes de serem usadas pela rede neural. A normalização consistia no mapeamento de um intervalo de valores em unidades Hounsfield para um intervalo entre 0 e 1. As intensidades de 0 a 100 eram mapeadas no primeiro canal, as intensidades de 100 a 200 eram mapeadas no segundo canal e as intensidades de 200 a 800 eram mapeadas no terceiro canal. Este mapeamento é possível

Tabela 9 – Volume em voxels de cada classe nos 113 anotações com espaçamento original disponíveis para treinamento e validação no desafio ADAM.

Classe	0	1	2
Volume total	5135773829	59743	3394588
Volume relativo (%)	99,93279	0,00116	0,06605

Fonte: Autor.

Nota: O significado das classes é explicado no Quadro 6.

para uma imagem de ATC, pois a intensidade dos voxels desta modalidade está associada ao material que o voxel representa, à temperatura da região e à voltagem do equipamento de raio X, como apresentado na revisão bibliográfica. Enquanto para uma imagem de angiografia por ressonância magnética do tipo TOF, a intensidade dos voxels não depende só do material, da temperatura e da configuração do equipamento, o movimento interfere no resultado, vide revisão bibliográfica.

Tabela 10 – Estatísticas referentes à intensidade dos voxels de valor mínimo, mediano e máximo das 113 imagens de TOF angio-RM disponíveis para treinamento e validação no desafio ADAM.

Estatísticas	Voxel mínimo	Voxel mediano	Voxel máximo
Média	0,0	118,3	1647,6
Desvio-padrão	0,0	138,4	1259,8
Mínimo	0,0	3,0	128,8
25%	0,0	14,1	301,6
50%	0,0	66,4	1905,7
75%	0,0	155,7	2612,2
Máximo	0,0	620,7	6119,1

Fonte: Autor.

Observando a Tabela 10, nota-se que exceto que as imagens de TOF angio-RM possuem sempre valor mínimo igual a zero, mas os valores máximo e mediano não têm um limite bem definido. As imagens desta modalidade variam a ponto da mediana de uma imagem ser mais intensa do que o máximo de outra imagem. Como os aneurismas não tratados nem rompidos possuem um volume com ordens de grandeza menores que o tamanho da imagem, estas variações de valores máximos e medianos não relacionadas aos aneurismas nos exames, por isso é de interesse reduzir estas variações de modo a evitar correlações não representativas. Como apresentado na revisão bibliográfica, as soluções mais bem classificadas empregam uma normalização Z, que consiste em subtrair da imagem a sua média e dividir esta diferença pelo desvio-padrão da imagem. Este trabalho também usa esta normalização.

3.2 Treinamento do modelo GLIA-Net no banco de dados ADAM

3.2.1 Transferência de aprendizado

Como apresentado na revisão do estado da arte, a transferência de aprendizado pode agilizar a convergência da performance de um modelo durante o treinamento.

No trabalho original, a rede neural profunda GLIA-Net foi treinada para segmentar aneurismas em imagens de angiografia por tomografia computadorizada. Enquanto neste trabalho, deseja-se treinar a mesma rede GLIA-Net para segmentar aneurismas em imagens de ressonância magnética.

Assim, podemos formular uma transferência de dados onde temos um domínio fonte formado por imagens de ATC e suas anotações, um domínio alvo formado por imagens de RM e suas anotações e uma tarefa composta pelo espaço de anotações classificando voxels em aneurismas ou em regiões normais e pela função que realiza a segmentação semântica de uma imagem tomográfica da cabeça em regiões com aneurismas e normais. Desta forma, temos tarefas fonte e alvo idênticas e domínios fonte e alvo distintos.

A rede neural GLIA-Net emprega funções de ativação não lineares, como sigmoide (Equação 2.3), softmax (Equação 2.7) e Leaky ReLU (Equação 2.2), e apresenta um comportamento não linear, por conseguinte a transferência de aprendizado é uma transferência de aprendizado profundo.

Das quatro possibilidades de aplicação da transferência de aprendizado profundo descritas na literatura, opta-se neste trabalho por empregar a transferência de aprendizado profundo baseada em rede. No projeto GLIA-Net original, está disponível uma rede já treinada no banco de dados de exames de ATC disponível através do *link* <<https://github.com/MeteorsHub/GLIA-Net/releases/download/v1.0/checkpoint-0245700.pt>>.

Por conta das normalizações diferentes aplicadas às imagens de ATC e TOF angio-RM, as entradas *global inputs* e *local inputs* no caso de cada modalidade possuem um número diferente de canais. Isto implica que o número de parâmetros dos primeiros filtros convolucionais possuem um número de parâmetros diferentes. Por esta razão, todos os parâmetros da nova rede neural, exceto aqueles cuja entrada é a entrada da rede, são inicializados com os parâmetros correspondentes da rede neural treinada com imagens de ATC para segmentar imagens de TOF angio-RM. De forma mais explícita, apenas as duas primeiras convoluções tridimensionais com *kernels* de tamanho unitário do primeiro encoder do componente *feature generator* e do primeiro encoder da rede *local segmentation network* não recebem os parâmetros do modelo já treinado.

3.2.2 Protocolo de treinamento

O desenvolvimento de uma solução de aprendizado supervisionado exige geralmente três banco de dados: de treino, de validação e de teste. O banco de dados de treino é usado para adaptar os parâmetros da solução de forma a minimizar uma função de custo que deve estar associada à performance do algoritmo. Já o banco de dados de validação é usado para avaliar ao longo do treinamento se os resultados obtidos são reproduzidos em dados diferentes dos empregados no treinamento. Finalmente, o banco de dados de teste é empregado para verificar a performance do algoritmo para dados diferentes dos usados no treinamento da rede.

Neste contexto, a adaptação do modelo GLIA-Net para a tarefa de segmentar aneurismas cerebrais em imagens de TOF angio-RM emprega três bancos de dados sem intersecção: os bancos de dados de treino, de validação e de teste. O banco de dados de treino é composto por 90 imagens de TOF angio-RM anotadas e selecionadas aleatoriamente do banco de dados disponibilizado do desafio ADAM. O banco de dados de validação é composto pelas 23 imagens de TOF angio-RM anotadas que não foram selecionadas do banco de dados disponibilizado do desafio ADAM para o treinamento. O banco de dados de teste do desafio é composto por 140 imagens de TOF angio-RM anotadas do banco de dados secreto de teste do desafio ADAM. Apesar do desafio disponibilizar o banco de dados secreto para uma comparação de resultados mais confiável, neste trabalho, assumi-se o banco de dados de validação como banco de dados de teste. Em particular, a composição dos bancos de dados disponibilizados aos participantes é mostrada nos Quadros 7 e 8.

Quadro 7: Composição do banco de dados de treino.

10001.nii.gz	10002.nii.gz	10004.nii.gz	10006.nii.gz	10007.nii.gz
10008.nii.gz	10009.nii.gz	10011.nii.gz	10013.nii.gz	10014.nii.gz
10015.nii.gz	10016.nii.gz	10017.nii.gz	10018.nii.gz	10019.nii.gz
10020.nii.gz	10023.nii.gz	10024.nii.gz	10025.nii.gz	10026.nii.gz
10027.nii.gz	10028.nii.gz	10030.nii.gz	10031.nii.gz	10032.nii.gz
10033.nii.gz	10034.nii.gz	10036.nii.gz	10037.nii.gz	10038.nii.gz
10039.nii.gz	10041.nii.gz	10042.nii.gz	10043.nii.gz	10044F.nii.gz
10045B.nii.gz	10045F.nii.gz	10046B.nii.gz	10046F.nii.gz	10047B.nii.gz
10047F.nii.gz	10048B.nii.gz	10048F.nii.gz	10049B.nii.gz	10049F.nii.gz
10050B.nii.gz	10050F.nii.gz	10051B.nii.gz	10051F.nii.gz	10052B.nii.gz
10053B.nii.gz	10053F.nii.gz	10054B.nii.gz	10054F.nii.gz	10055B.nii.gz
10055F.nii.gz	10056B.nii.gz	10056F.nii.gz	10057B.nii.gz	10057F.nii.gz
10058B.nii.gz	10059B.nii.gz	10059F.nii.gz	10060B.nii.gz	10060F.nii.gz
10061B.nii.gz	10061F.nii.gz	10062F.nii.gz	10063B.nii.gz	10064B.nii.gz
10064F.nii.gz	10065B.nii.gz	10065F.nii.gz	10067B.nii.gz	10067F.nii.gz
10068B.nii.gz	10068F.nii.gz	10069B.nii.gz	10070F.nii.gz	10072B.nii.gz
10072F.nii.gz	10073B.nii.gz	10073F.nii.gz	10074B.nii.gz	10075B.nii.gz
10076B.nii.gz	10076F.nii.gz	10077B.nii.gz	10078B.nii.gz	10078F.nii.gz

Quadro 8: Composição do banco de dados de validação.

10003.nii.gz	10005.nii.gz	10010.nii.gz
10012.nii.gz	10021.nii.gz	10022.nii.gz
10029.nii.gz	10035.nii.gz	10040.nii.gz
10044B.nii.gz	10052F.nii.gz	10058F.nii.gz
10062B.nii.gz	10063F.nii.gz	10066B.nii.gz
10066F.nii.gz	10069F.nii.gz	10070B.nii.gz
10071B.nii.gz	10071F.nii.gz	10074F.nii.gz
10075F.nii.gz	10077F.nii.gz	-

O treinamento da rede GLIA-Net é realizado ao longo de épocas. A cada época, a rede recebe a entrada de cada imagem do banco de dados de treinamento, calcula uma saída, calcula a função de custo em função desta saída e, por meio de um algoritmo de retro-propagação, os parâmetros da rede neural são atualizados de forma a tentar reduzir a função de custo. A função de custo empregada no treinamento é apresentada na subseção 2.2.2.1.1 e os pesos de cada não termo não foram modificados em relação ao trabalho anterior, onde $\omega_{Global} = 0,1$, $\omega_{Local} = 1,0$, $\omega_{Dice} = 0,2$, $\omega_{Cross} = 0,8$, $\gamma_{Dice} = 0,3$ e $\gamma_{Cross} = 0,3$.

Quanto ao algoritmo de retro-propagação, utiliza-se o otimizador Adam (KINGMA; BA, 2015) com uma taxa de aprendizado inicial igual a 0,000058398, a taxa de aprendizado final usada no treinamento do modelo pré-treinado para modalidade ATC. A taxa de aprendizado é reduzida em 5% a cada 10000 passos de treinamento da rede, onde um passo de treinamento corresponde a uma etapa de adaptação dos parâmetros da rede neural por meio do algoritmo de retro-propagação, da mesma forma que o trabalho original GLIA-Net desenvolvido para a modalidade ATC. Esta atualização dos parâmetros da rede ocorre após o modelo ser aplicado a três blocos (*patches*) de imagens, ou seja, o tamanho do lote (*batch size*) é igual a três, análogo ao trabalho GLIA-Net original para a modalidade ATC.

A atualização dos parâmetros do modelo não é aplicada a todas as camadas da rede neural, atualiza-se apenas os componentes *global localizer*, *localizer loss*, *global positioning adaptors*, *output conv* e os *decoders* da rede *local segmentation network*, ou seja, apenas os *encoders* da rede *local segmentation network* estão fixos. Na literatura, estas camadas cujos parâmetros estão fixados, mas poderiam ser adaptados ao longo do treinamento são consideradas de camadas congeladas (*frozen layers*). Este método de treinar o modelo foi baseado no artigo (KARIMI; WARFIELD; GHOLIPOUR, 2020), mencionado na seção 2.3, em que mostra-se que o congelamento dos *encoders* e o treinamento apenas dos *decoders* de uma rede do tipo 3D U-Net pode obter resultados equiparáveis ao treino da rede completa.

Como já mencionado na subseção 2.2.2.1, o modelo GLIA-Net é aplicado a cada

bloco (*patch*) da imagem. No treinamento e na validação, para cada imagem completa do banco de dados o número de blocos selecionados para treinar/validar o modelo é igual a duas vezes a razão entre o tamanho da imagem completa e o tamanho do bloco (*patch*). Para lidar com o desbalanceamento de dados exposto na Tabela 9, metade destes blocos é selecionada com aneurismas. Caso não haja aneurismas intracranianos no exame, apenas metade dos blocos é selecionada e nenhuma imposição é imposta para a seleção das regiões. Este treinamento em *patches* com balanceamento de dados já era feito no projeto GLIA-Net aplicado a imagens de ATC (BO et al., 2021), porém os *patches* sem aneurismas eram restritos àqueles cuja média é maior do que 0,05, ou seja, regiões pretas do plano de fundo de imagens da modalidade ATC. No treinamento desenvolvido neste trabalho para a modalidade TOF angio-RM, esta restrição não é imposta.

A seleção dos blocos é aleatória, porém na etapa de validação a semente do gerador de números aleatórios é fixada para garantir que a avaliação do algoritmo em cada época seja aplicada nas mesmas entradas. O tamanho dos blocos é fixado em $96 \times 96 \times 96$, o mesmo do projeto GLIA-Net para a modalidade ATC e para que um bloco seja considerado com aneurisma, seu centro deve estar a uma distância máxima de 28 voxels em cada direção do centro de um aneurisma.

O pré-processamento dos dados é composto pelas etapas sequenciais: mudança de espaçamento, seleção de *patches*, aumento de dados e normalização. A mudança de espaçamento é realizada conforme explicado na seção 3.1, todas as imagens são redimensionadas de forma a apresentarem o espaçamento mediano $0,357143 \times 0,357143 \times 0,500000 \text{ mm}^3$ (Tabela 7 linha 50%). Para cada exame duas imagens devem ser redimensionadas: a imagem de entrada de TOF angio-RM e a anotação desta imagem. Para as imagens de entrada da rede, que são contínuas no espaço, usa-se a interpolação polinomial do tipo *spline* cúbica visando preservar a forma da imagem original, enquanto para as anotações associadas, que são imagens binárias, usa-se a interpolação de vizinhos mais próximos, que preserva os valores da anotação. Com o novo tamanho após o dimensionamento, todas as imagens são pequenas o suficiente para serem armazenadas na memória RAM e processadas pela GPU empregada. Contudo, a profundidade de algumas imagens é inferior a 96 voxels – a profundidade do *patch*. Visando solucionar este problema, estas imagens são completadas artificialmente com o valor de um dos voxels da ponta que faz contato com a região estendida.

A etapa de aumento artificial de dados (*data augmentation*) consiste em aplicar transformações aos dados de treino que não afetam as anotações ou que as afetam de maneira conhecida e buscando a simular uma quantidade maior de dados distintos. Em particular, aplica-se as seguintes transformações: ruído Gaussiano aditivo com média e desvio-padrão iguais a, respectivamente, 0 e 1, correção *gamma* aleatória da imagem elevando os valores dos pixels a um expoente normalmente distribuído com média igual a 1

e desvio-padrão igual a 0,1, espelhamento aleatório da imagem nos três planos normais aos eixos de indexação da imagem e permutação dos eixos da imagem. As duas primeiras transformações aumentam a diversidade da intensidade dos voxels entre cada época, enquanto as duas últimas transformações aumentam a variedade de poses do paciente na imagem, e portanto, do aneurisma na imagem. A normalização aplicada à entrada é a normalização Z, já discutida na seção 3.1.

Como discutido na seção 3.1, a classe 2 é ordens de grandeza mais numerosa que a classe 1, que gostaríamos de detectar, e ordens de grandeza inferior à classe 0 e que a representação dos voxels da classe 2 como 0 deve ser a estratégia que melhor mitiga a influência desta classe no treinamento do modelo. Por esta razão, o modelo só considera duas classes: 0 e 1, sendo os voxels da classe 2 anotados como membros da classe 0.

Concernindo as métricas de avaliação do modelo na tarefa, são considerados dois cenários: avaliação por *patches* e avaliação por exames.

No cenário de avaliação por *patches*, a avaliação da máscara de segmentação é realizada selecionando apenas *patches* específicos da imagem de forma a mitigar artificialmente o desbalanceamento das classes. Esta avaliação ocorre durante o treinamento da rede no banco de dados de treino e no banco de dados de validação e as métricas usadas são as mesmas empregadas no trabalho original GLIA-Net (BO et al., 2021): precisão média (AP), AUC (ROC), coeficiente de similaridade de Sorensen-Dice (DSC), precisão, sensibilidade (*recall*), distância de Hausdorff modificada para considerar o percentil 95 das menores distâncias (HD95) no nível dos voxels e precisão por alvo e sensibilidade por alvo no nível dos conjuntos conectados de voxels anotados como aneurismas. Assim como em no trabalho original GLIA-Net aplicado na modalidade ATC, as métricas de performance do algoritmo em uma época é calculada em função do número de , verdadeiro-positivo, verdadeiro-negativos, falso-positivos e falso-negativos total considerando todos os *patches* empregados no treinamento ou na validação.

3.3 Segmentação de exames de TOF angio-RM

Na avaliação por exames, as imagens completas são segmentadas e comparadas às respectivas anotações. Esta avaliação é mais fiel à aplicação concebida da solução GLIA-Net, pois os *patches* são selecionados de forma a cobrir todas as regiões do exame e o balanceamento das classes é preservado. Esta avaliação ocorre durante a inferência de exames que se deseja detectar/segmentar aneurismas em imagens e as métricas usadas no desafio ADAM são empregadas, incluindo o coeficiente de similaridade de Sorensen-Dice (DSC), a distância de Hausdorff modificada para considerar o percentil 95 das menores distâncias (HD95), a similaridade volumétrica (VS) no nível dos voxels e a sensibilidade e o número de falso-positivos (FP) no nível dos conjuntos conectados de voxels anotados como

aneurismas. A implementação destas métricas é baseada nos exemplos já desenvolvidos no repositório de avaliação do desafio ADAM, disponível em <https://github.com/hjkuijf/ADAMchallenge> (KUIJF, 2020). Diferentemente da avaliação por *patches*, as métricas do algoritmo em um banco de dados são as médias das métricas calculadas em cada exame individualmente do banco de dados em questão.

O processo de inferência das regiões com aneurismas dos exames de TOF angio-RM funciona da seguinte maneira: redimensiona-se as imagens originais completas de forma que estas apresentem o espaçamento $0,357143 \times 0,357143 \times 0,500000$ – o mesmo usado no treinamento da rede –, seleciona-se os *patches* dos exames de imagem cobrindo a imagem inteira, aplica-se a rede neural GLIA-Net a cada *patch* gerando uma máscara de segmentação para cada *patch*, compõem-se um máscara de segmentação da imagem completa a partir das máscaras geradas a partir de cada *patch* e redimensiona-se esta máscara de segmentação da imagem completa resituindo o espaçamento da imagem de TOF angio-RM original.

Na etapa de seleção dos *patches*, *patches* de tamanho $96 \times 96 \times 96$ são sobrepostos de tal forma que dois *patches* lado a lado possuam 48 superfícies 96×96 de voxels, ou seja, dois *patches* lado a lado possuem metade dos seus voxels compartilhados entre si. Se um *patch* não está na borda da imagem completa, em cada direção este patch está sobreposto a dois outros *patches* compartilhando 48 superfícies 96×96 de cada lado, e portanto, todos os seus voxels são compartilhados com outros *patches*. Assim, apenas os cubos $48 \times 48 \times 48$ dos cantos da imagem são representados por apenas um *patch*.

Considerando que cada *patch* passa pela rede neural gerando uma máscara de segmentação na mesma posição, levanta-se a questão de como unificar a informação de todos os *patches* sobrepostos. No projeto original GLIA-Net voltado para a modalidade ATC, a unificação das máscaras dos *patches* sobrepostos é realizada pela média aritmética simples, a intensidade final de um voxel é a média aritmética simples da intensidade de todas as máscaras que ocupam aquele voxel. Entretanto, esta solução é problemática no sentido que os *patches* possuem o mesmo peso na geração da intensidade de um voxel independentemente da posição do voxel nos *patches*. Lembrando que no treinamento a seleção dos *patches* é realizada buscando minizar o desbalanceamento das classes e que para um *patch* ser considerado com aneurisma este deve conter um centro de aneurisma a uma distância menor ou igual a 28 voxels em cada direção, conclui-se que aneurismas cujo centro está na borda de espessura 20 voxels de um *patch* não foram representados no treinamento da rede neural, e logo a performance do algoritmo não é garantida pelo processo de treinamento.

Neste trabalho, propõe-se o uso de um tensor de pesos linear para realizar a média ponderada dos *patches*, onde o peso de um voxel é a distância de Manhattan (Equação 3.1)

deste voxel em relação ao voxel do canto mais próximo do *patch* acrescentado de um.

$$d(p, q) = ||p - q||_1 = \sum_{i=1}^{i=N} |p_i - q_i| \quad (3.1)$$

onde p e q são dois pontos de N coordenadas e p_i e q_i são suas i -ésimas coordenadas. Para um *patch* de tamanho $96 \times 96 \times 96$, estes os pesos deste tensor aumentam linearmente do canto para o centro do *patch*, sendo os valores mínimo e máximo do tensor iguais a 1 nos cantos e 142 nos voxels centrais.

3.3.1 Determinação do limiar de segmentação

Todas as métricas usadas para avaliação de um algoritmo de segmentação e de detecção no desafio ADAM, coeficiente de similaridade de Sorensen-Dice, distância de Hausdorff considerando o percentil 95 das menores distâncias, coeficiente de similaridade volumétrica, sensibilidade e quantidade de falso-positivos por caso, variam com o limiar de segmentação. Assim, é necessário fixar um limiar de segmentação para avaliar o modelo em um banco de dados.

Dentre as métricas usadas, apenas o coeficiente de similaridade de Sorensen-Dice, a distância de Hausdorff modificada e o coeficiente de similaridade volumétrica consideram tanto os erros falso-positivos quanto os erros falso-negativos. Logo, é possível otimizar o modelo usando apenas uma dessas métricas. Como já explicado na subseção 2.2.1.2.3, uma similaridade volumétrica melhor possível, igual a 1, não implica necessariamente em uma segmentação boa. Por esta razão, esta métrica não é usada para a determinação do limiar. Restam apenas duas métricas: a distância de Hausdorff modificada e o coeficiente de similaridade de Sorensen-Dice. Neste trabalho, o coeficiente de Sorensen-Dice é empregado para a determinação do limiar de segmentação.

Para isso, itera-se no banco de dados de validação varrendo a imagem completa sem distorcer o balanceamento de classes e variando o limiar de segmentação. São escolhidos 100 limiares com passo constante $1/101$ entre 0 e 1. Para cada limiar e para cada exame, uma máscara de segmentação binária é gerada a partir da máscara de segmentação prevista pelo modelo GLIA-Net contendo a probabilidade de cada voxel pertencer às classes 0 (normal) e 1 (aneurisma não tratado nem rompido). Em seguida, esta máscara binária é comparada com a máscara de segmentação anotada e as métricas coeficiente de Sorensen-Dice, distância de Hausdorff modificada, similaridade volumétrica, sensibilidade e quantidade de falso-positivos por caso são calculadas. Para cada limiar, este processo é realizado em todas as instâncias do banco de dados de validação e as métricas para cada limiar são as médias aritmética de cada métrica calculada para cada exame. O limiar de segmentação escolhido é aquele que maximiza a média do coeficiente de Sorensen-Dice em todas as imagens do banco de dados de validação.

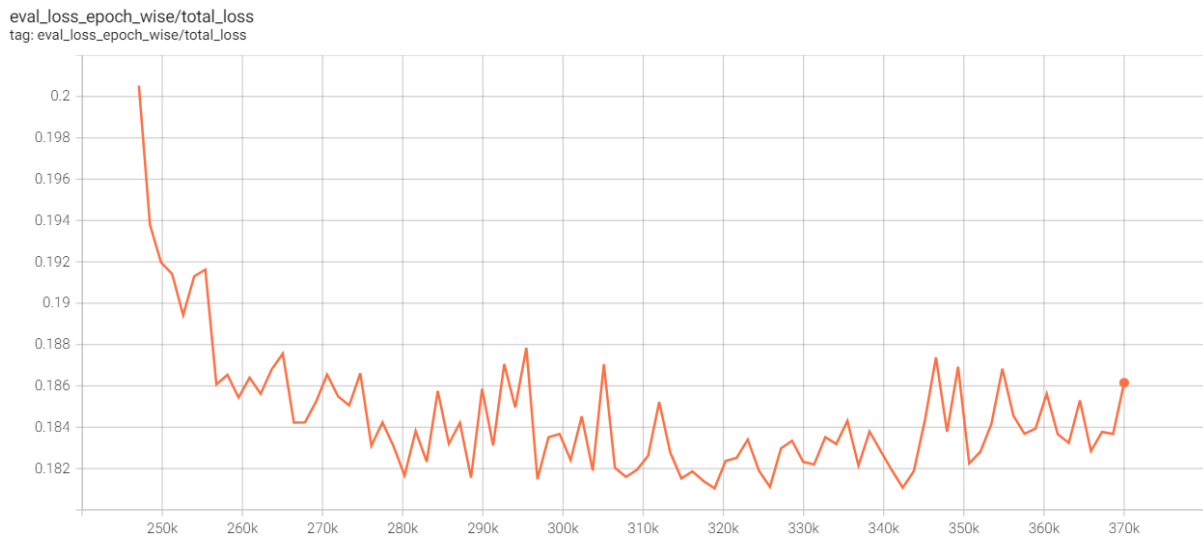
4 Resultados e Discussão

4.1 Treinamento de GLIA-Net no banco de dados ADAM

4.1.1 Função de custo

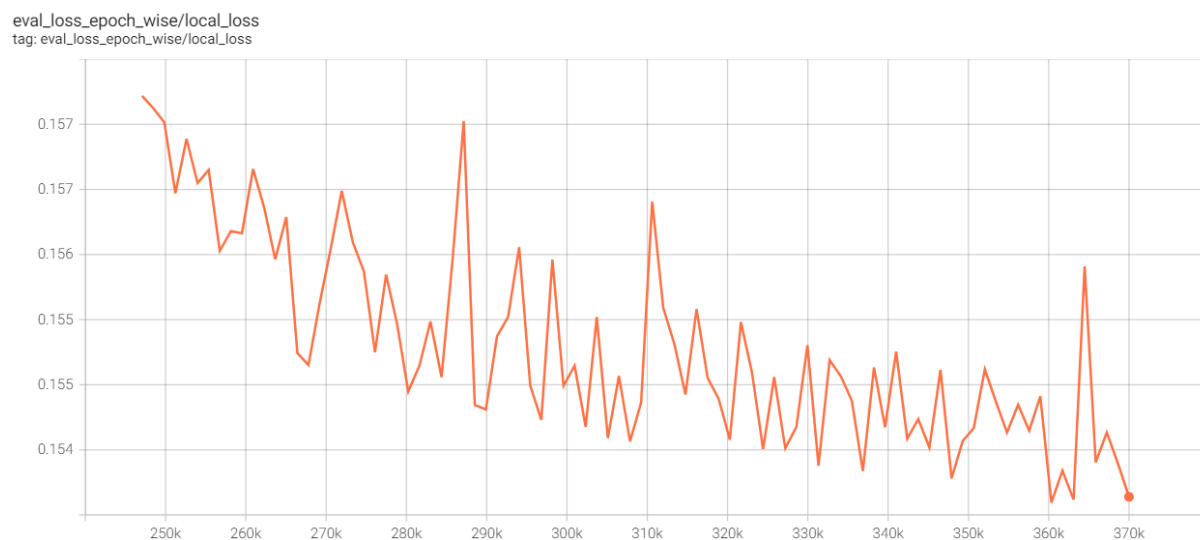
A função de custo da rede GLIA-Net no banco de dados de validação consta nas Figuras 22, 23 e 24, enquanto a função de custo da rede GLIA-Net no banco de dados de treinamento consta nas Figuras 25, 26 e 27.

Figura 22 – Função de custo total no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.



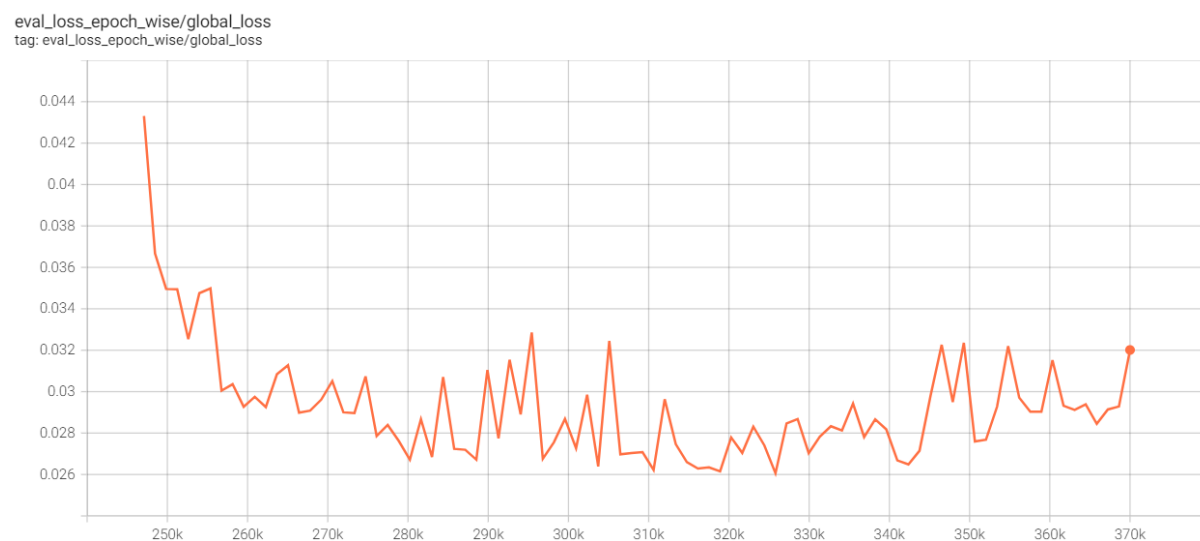
Fonte: Autor.

Figura 23 – Função de custo local no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.



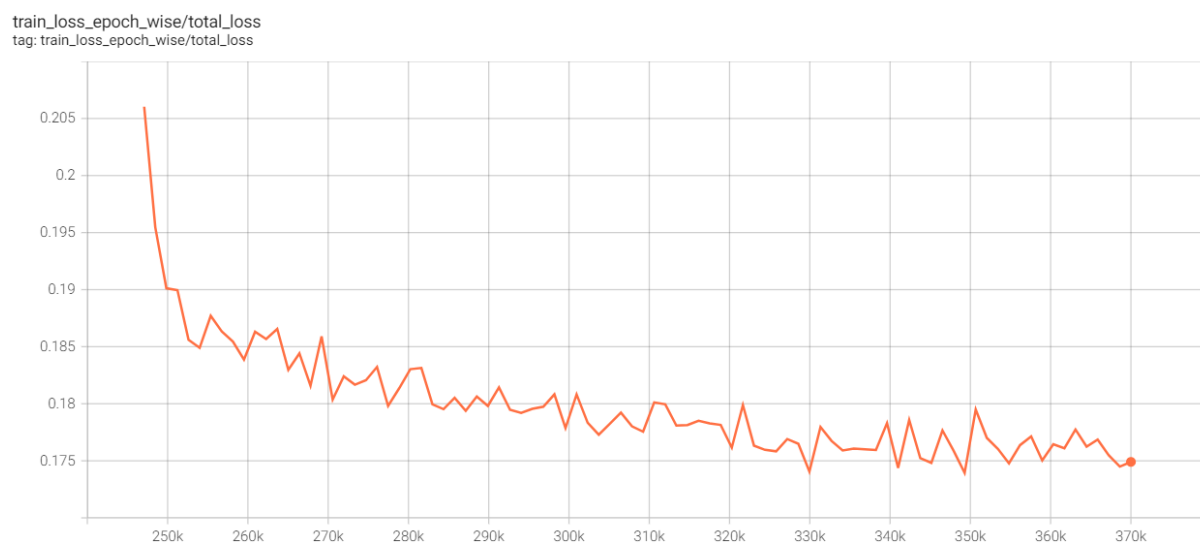
Fonte: Autor.

Figura 24 – Função de custo global no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.



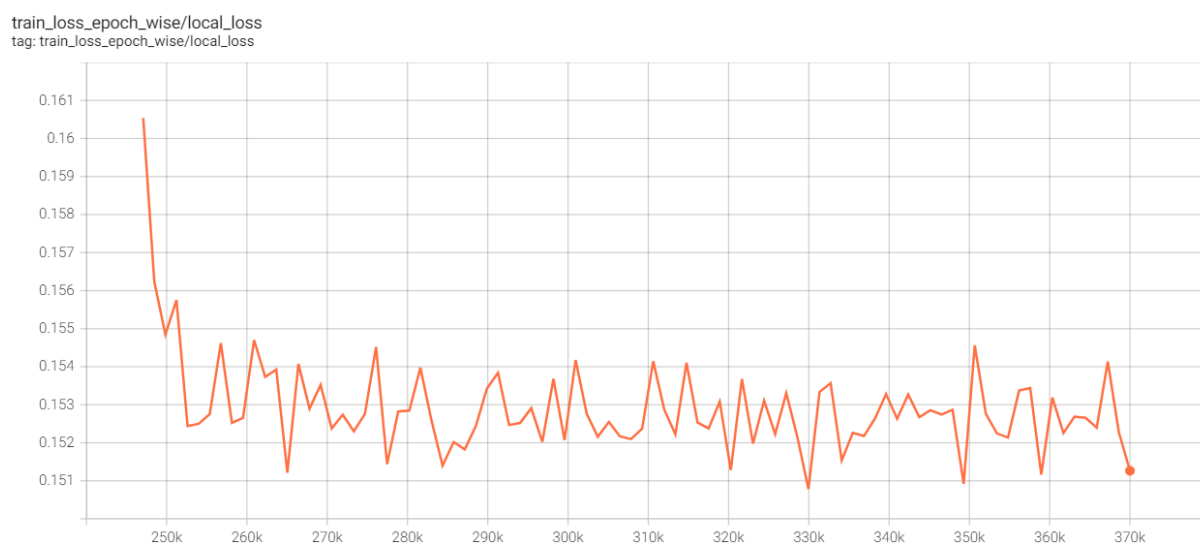
Fonte: Autor.

Figura 25 – Função de custo total no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.



Fonte: Autor.

Figura 26 – Função de custo local no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.



Fonte: Autor.

Figura 27 – Função de custo global no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.



Fonte: Autor.

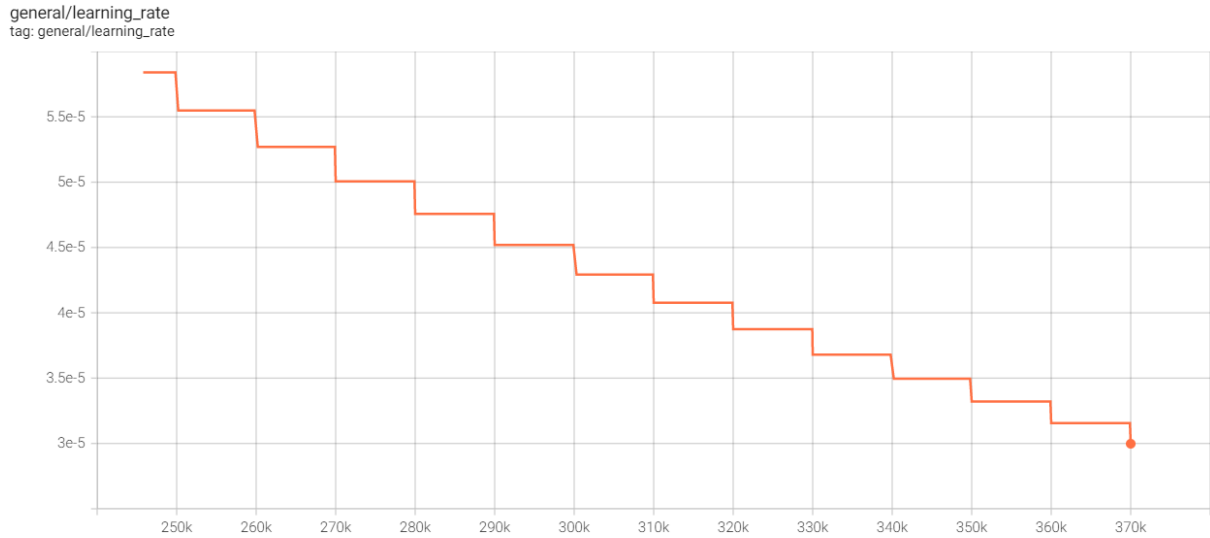
Com base nas Figuras 25 e 22, nota-se que a função de custo total do modelo GLIA-Net reduziu ao longo do treinamento tanto no banco de dados de treino como de validação, mas nos últimos 30 passos de treinamento a função de custo total diminui apenas no banco de dados de treino. Este cenário indica que ocorreu *overfitting* nos últimos passos de treinamento, ou seja, que o treinamento otimizou os parâmetros do modelo visando aumentar o seu desempenho no banco de dados de treino usando características particulares dos dados de treinamento, que não são generalizáveis para outros bancos de dados da mesma tarefa, como o banco de dados de validação.

Analisando as Figuras 24, 23, 27 e 26, que contém os componentes local e global da função de custo, nota-se que apenas a função de custo global sofre *overfitting*. A função de custo local continua a diminuir no banco de dados de validação ao longo do treinamento. Relembrando que a função de custo global é calculada a partir da saída *global localizer logits* da rede neural, enquanto a função de custo local é calculada a partir da saída *net* da rede neural e que a máscara de segmentação e as métricas de segmentação e de detecção são calculadas usando a saída *net* da rede neural, não podemos afirmar que a continuação do treinamento da rede GLIA-Net induza o modelo aprender características específicas do banco de dados de treino e não generalizáveis no banco de dados de validação, mas sim que a continuação do treinamento pode induzir a uma melhora na performance do algoritmo em ambos os bancos de dados. Ainda sim, *overfitting* na função de custo global pode em épocas futuras atrapalhar a performance do modelo nas tarefas de segmentação e de detecção já que os parâmetros da camada *global localizer* são afetados por ambas componentes da função de custo total.

Uma alternativa para mitigar o problema de *overfitting* da rede *global localiza-*

tion network é adicionar e intensificar os processos de aumento artificial de dados (*data augmentation*). Em particular, como esta rede utiliza apenas a imagem completa dimensionalmente reduzida (*global inputs*) e a posição do *patch* nesta imagem (*patch location box*), a adição de uma rotação com um ângulo aleatório pode reduzir a correlação particular de orientação do banco de dados de treino.

Figura 28 – Taxa de aprendizado ao longo do treinamento do modelo GLIA-Net no desafio ADAM.



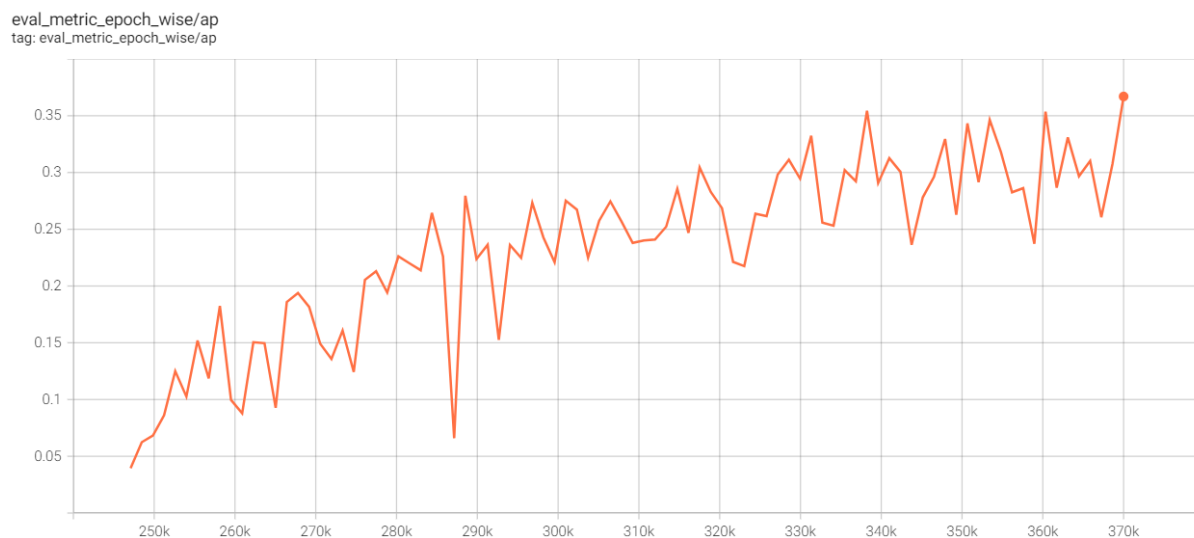
Fonte: Autor.

É perceptível nas Figuras de 22 a 25 uma alta variabilidade nas curvas de função de custo ao longo do treinamento. Uma possível justificativa para este fenômeno é o emprego de uma taxa de aprendizado inadequada. O treinamento do modelo GLIA-Net no artigo (BO et al., 2021) começou com uma taxa de aprendizado igual a $2 \cdot 10^{-4}$, enquanto a taxa de aprendizado inicial no treinamento de imagens de TOF angio-RM começou em $5,8398 \cdot 10^{-5}$. É possível que uma redução da taxa de aprendizado deste nível seja inadequada mesmo usando transferência de aprendizado. Testes exploratórios adicionais são necessários para verificar esta hipótese.

4.1.2 Métricas

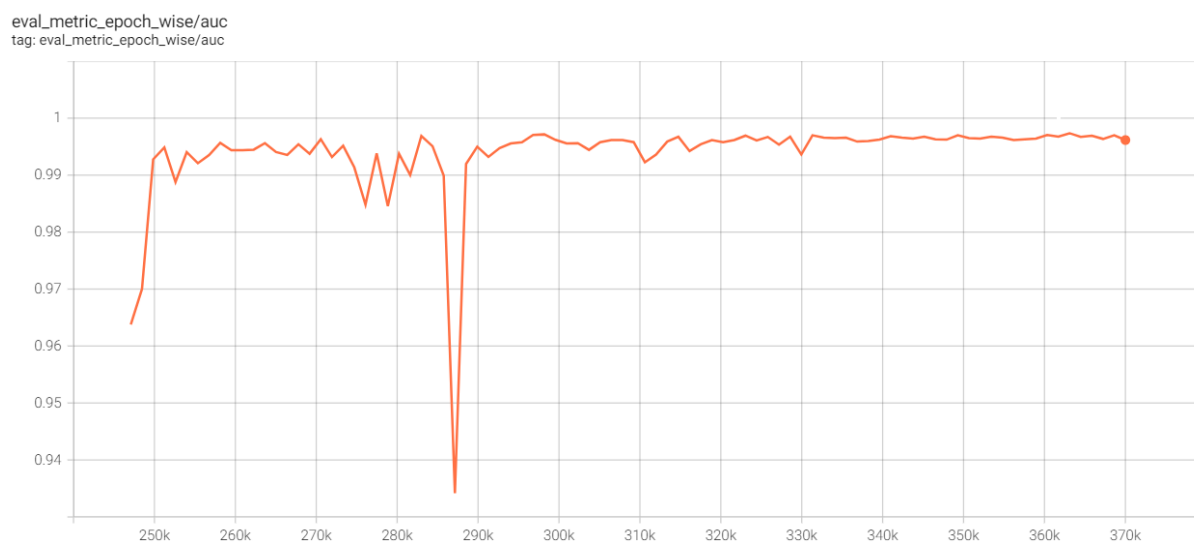
As métricas da rede GLIA-Net no banco de dados de validação constam nas Figuras 29, 30, 31, 32, 33, 34, 35 e 36, enquanto as métricas da rede GLIA-Net no banco de dados de treinamento consta nas Figuras 37, 38, 39, 40, 41, 42, 43 e 44.

Figura 29 – Métrica AP (precisão média) obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.



Fonte: Autor.

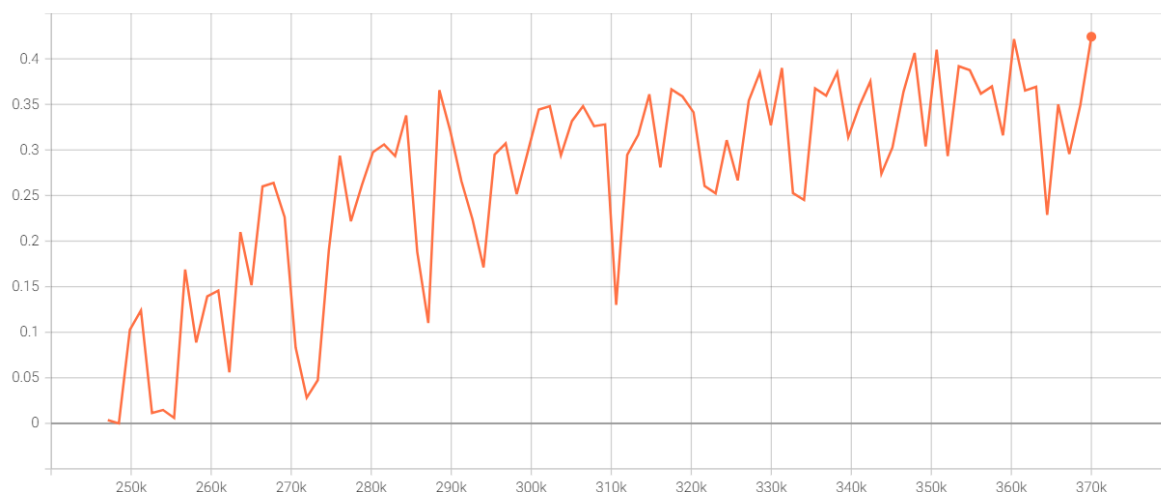
Figura 30 – Métrica AUC (área embaixo da curva ROC) obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.



Fonte: Autor.

Figura 31 – Métrica DSC (coeficiente de similaridade de Sorensen-Dice) obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.

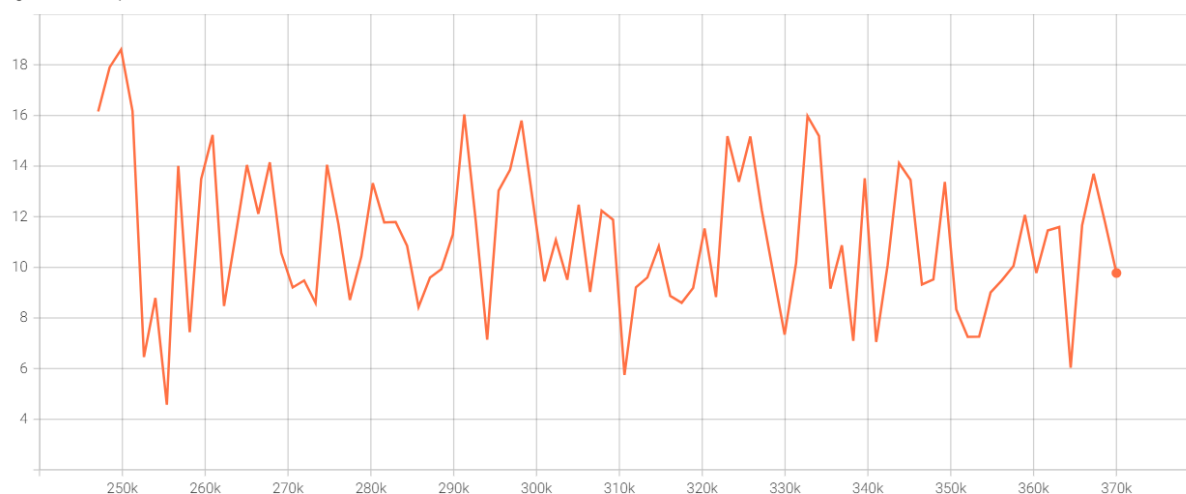
eval_metric_epoch_wise/dsc
tag: eval_metric_epoch_wise/dsc



Fonte: Autor.

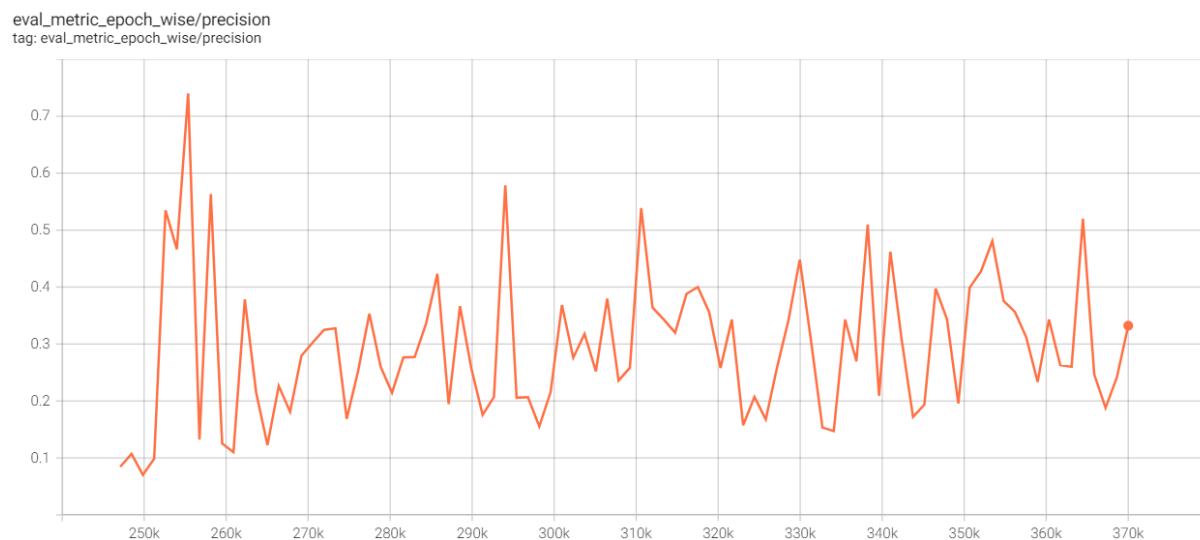
Figura 32 – Métrica HD95 (distância de Hausdorff considerando o percentil 95) obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.

eval_metric_epoch_wise/hd95
tag: eval_metric_epoch_wise/hd95



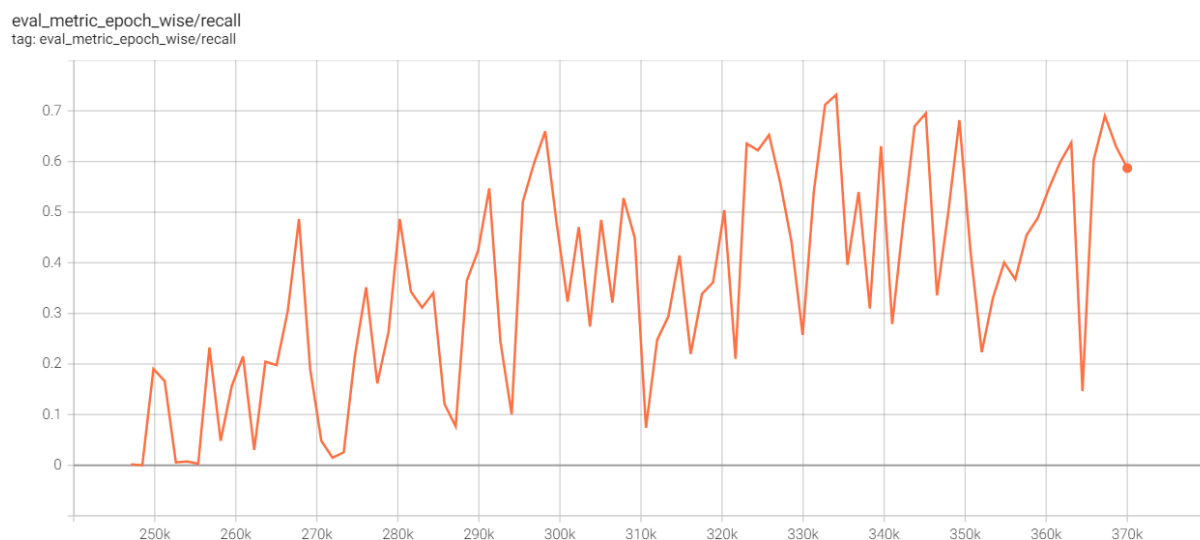
Fonte: Autor.

Figura 33 – Métrica precisão por voxel obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.



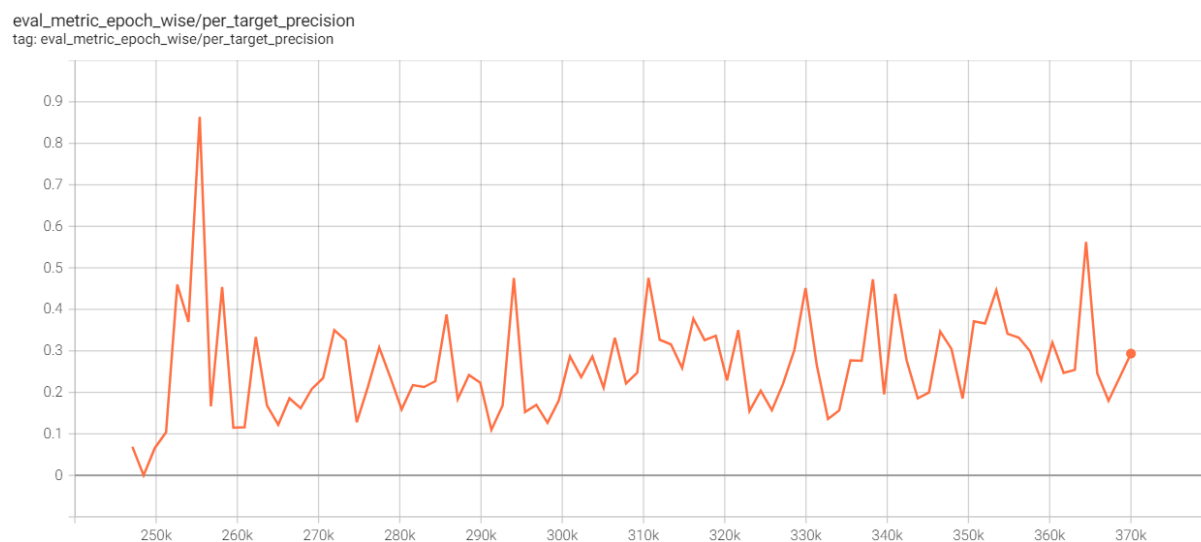
Fonte: Autor.

Figura 34 – Métrica sensibilidade por voxel obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.



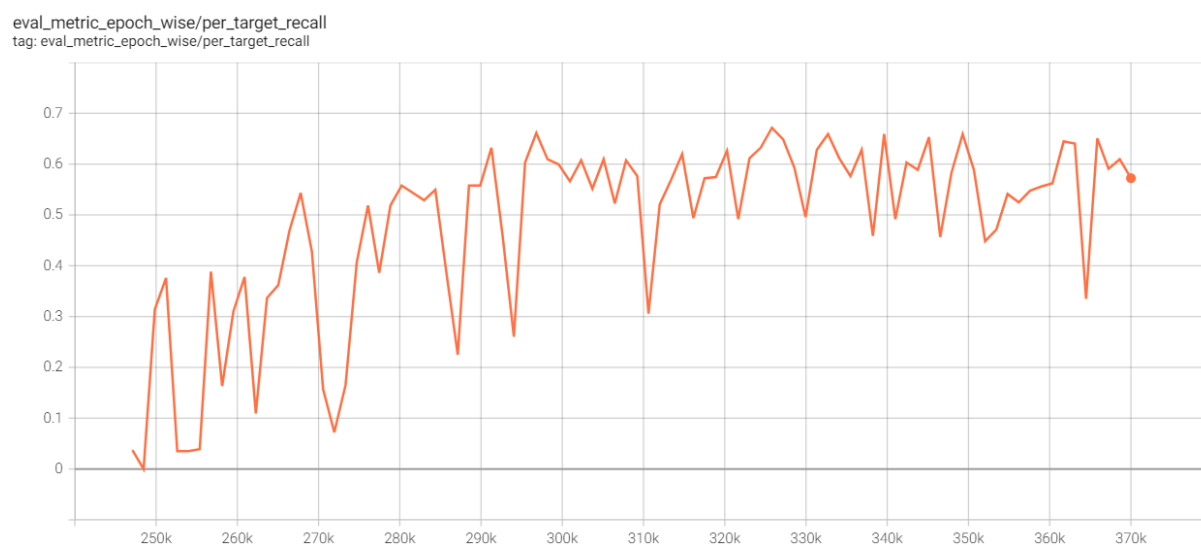
Fonte: Autor.

Figura 35 – Métrica precisão por exame obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.



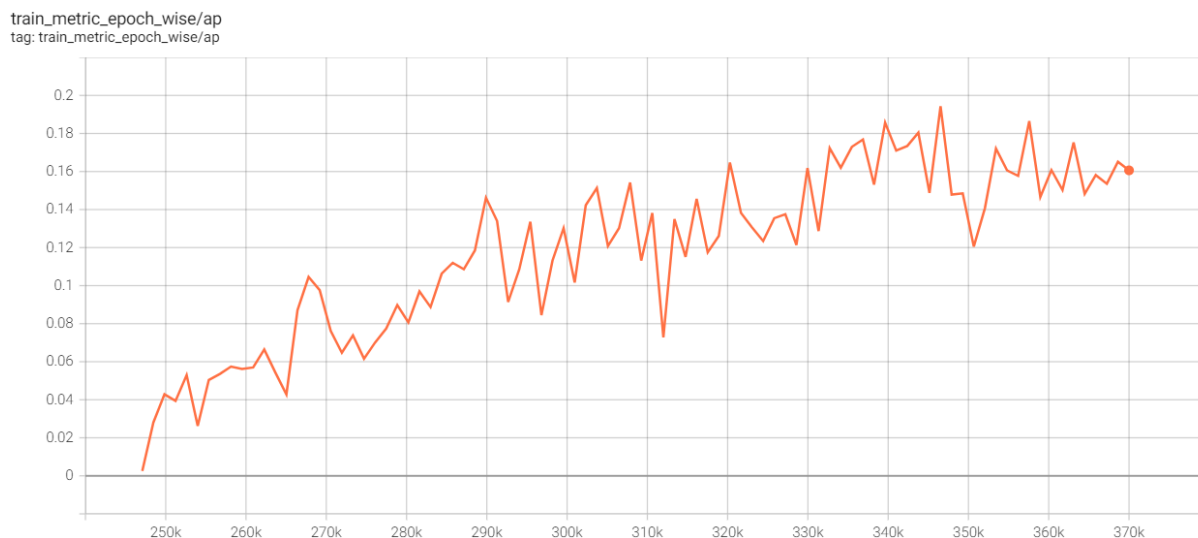
Fonte: Autor.

Figura 36 – Métrica sensibilidade por exame obtida no banco de dados de validação do desafio ADAM ao longo do treinamento da rede GLIA-Net.



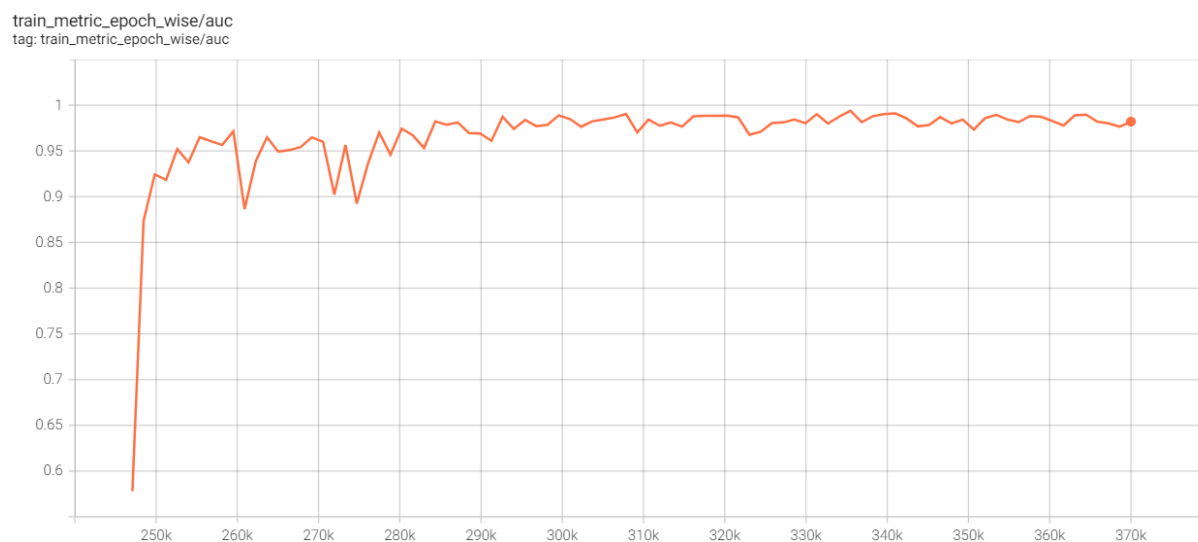
Fonte: Autor.

Figura 37 – Métrica AP (precisão média) obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.



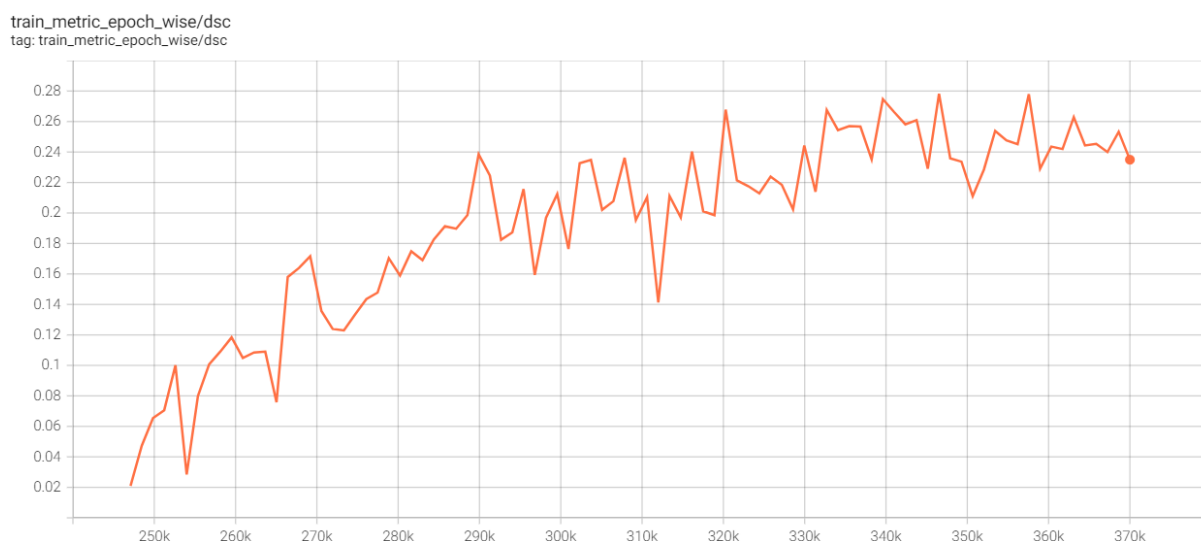
Fonte: Autor.

Figura 38 – Métrica AUC (área embaixo da curva ROC) obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.



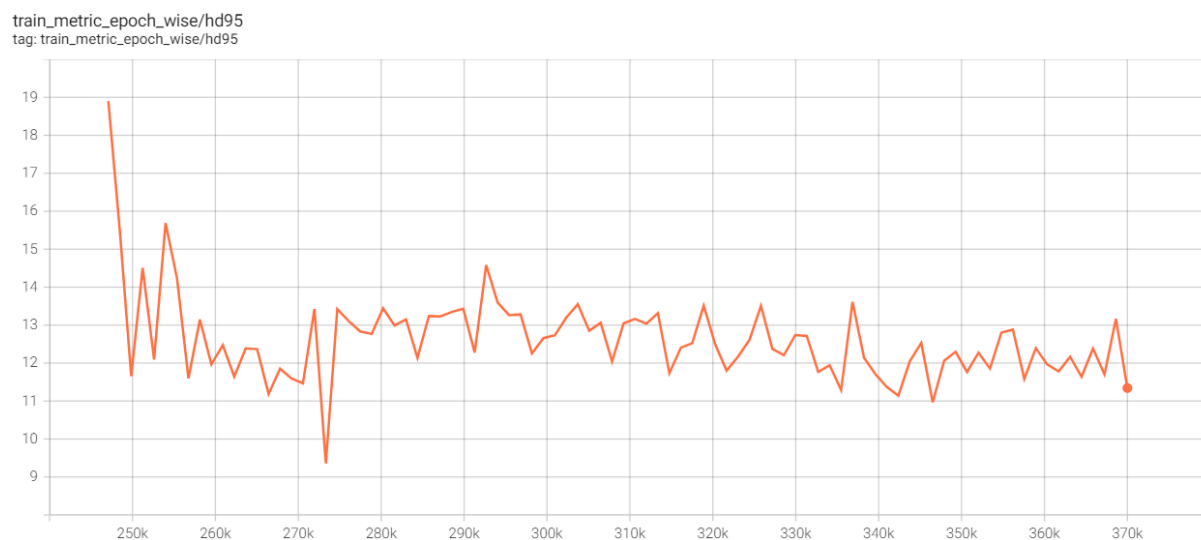
Fonte: Autor.

Figura 39 – Métrica DSC (coeficiente de similaridade de Sorensen-Dice) obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.



Fonte: Autor.

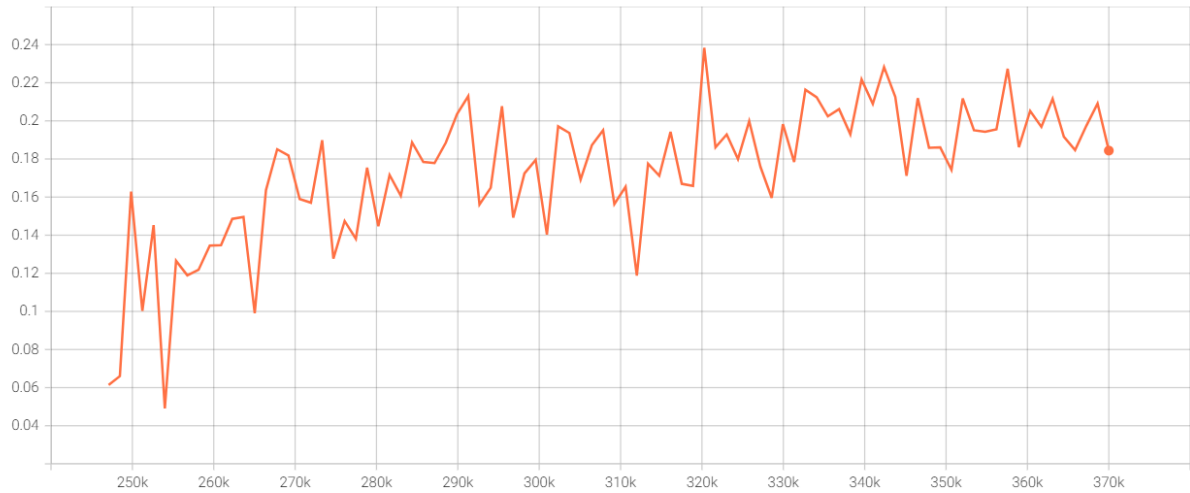
Figura 40 – Métrica HD95 (distância de Hausdorff considerando o percentil 95) obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.



Fonte: Autor.

Figura 41 – Métrica precisão por voxel obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.

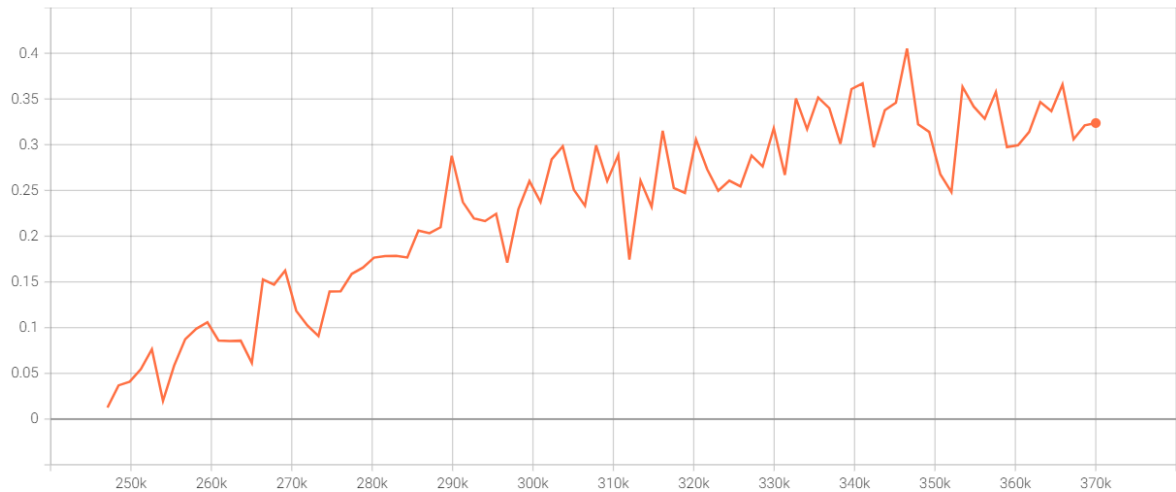
train_metric_epoch_wise/precision
tag: train_metric_epoch_wise/precision



Fonte: Autor.

Figura 42 – Métrica sensibilidade por voxel obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.

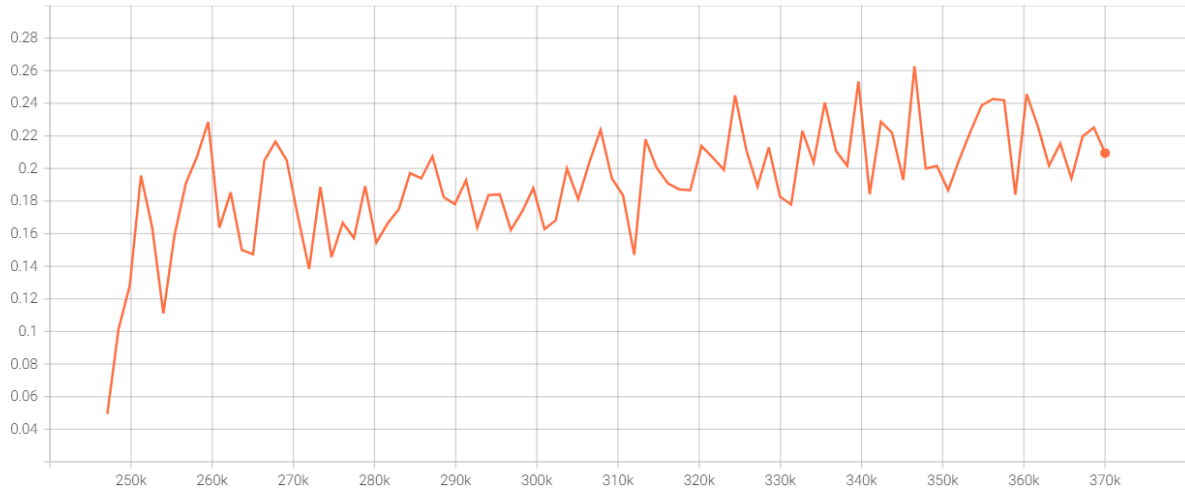
train_metric_epoch_wise/recall
tag: train_metric_epoch_wise/recall



Fonte: Autor.

Figura 43 – Métrica precisão por exame obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.

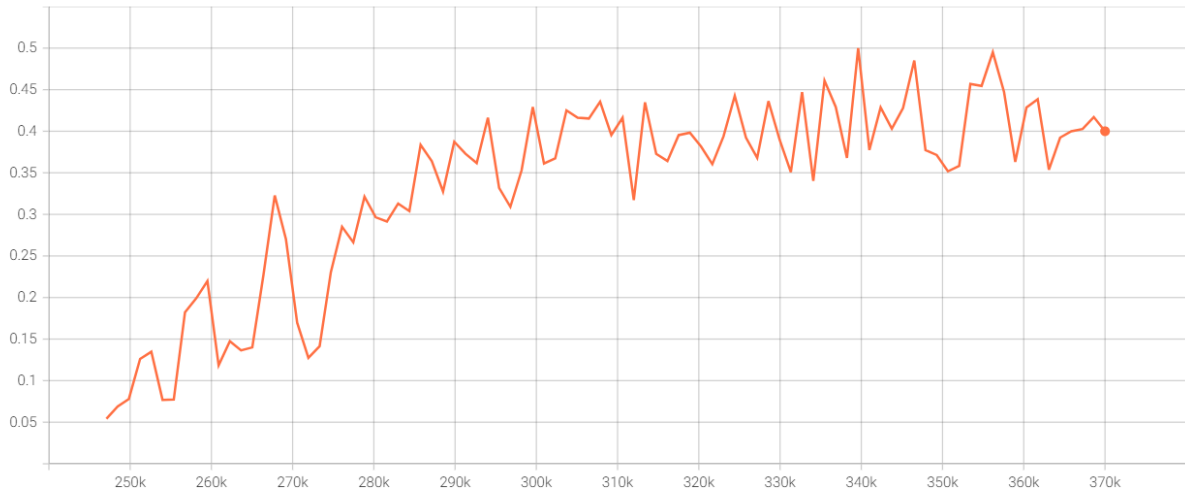
train_metric_epoch_wise/per_target_precision
tag: train_metric_epoch_wise/per_target_precision



Fonte: Autor.

Figura 44 – Métrica sensibilidade por exame obtida no banco de dados de treinamento do desafio ADAM ao longo do treinamento da rede GLIA-Net.

train_metric_epoch_wise/per_target_recall
tag: train_metric_epoch_wise/per_target_recall



Fonte: Autor.

Como já mencionado na subseção 2.2.1, a métrica precisão média (**AP**) é a mais representativa da performance do algoritmo, pois considera os erros falso-positivos e falso-negativos comparando com a classe menos numerosa (a positiva ou 1) e não depende do limiar de segmentação. A mesma subseção explica que o valor esperado de **AP** para um classificador no qual a previsão não depende da entrada vale a quantidade de voxels positivos (aneurismas não tratados nem rompidos) dividida pela quantidade total de vo-

xels no banco de dados. Abaixo esta precisão média é calculada para o banco de dados disponibilizado no desafio ADAM:

$$AP_{base_ADAM} = \frac{59743}{5135773829 + 59743 + 3394588} = 1,162 \cdot 10^{-5} \quad (4.1)$$

Logo, analisando as Figuras 29 e 37, que expressam a evolução da métrica **AP** ao longo do treinamento, observa-se que o desempenho do algoritmo é aparentemente ordens de grandeza superior que o desempenho esperado de um algoritmo de classificação aleatório. Um fator que dificulta esta comparação é que o cálculo feito na Equação 4.1 considera as imagens inteiras sem mudança de espaçamento nem rebalanceamento de dados, enquanto a métrica **AP** das Figuras 29 e 37 consideram o banco de dados selecionando prioritariamente *patches* com aneurismas e exames redimensionados para padronizar o espaçamento. Ainda sim, a quantidade máxima de *patches* observada em um exame redimensionado foi de 363 com sobreposição, então estas diferenças de pré-processamento sozinhas não são capazes de explicar a diferença de performance entre o algoritmo treinado e um algoritmo aleatório, implicando que o modelo treinado apresenta um desempenho melhor que um algoritmo aleatório.

Analogamente podemos comparar a performance do modelo GLIA-Net treinado com um classificador aleatório usando a métrica **AUC**. Nas Figuras 30 e 38, que exibem a evolução da métrica **AUC** ao longo do treinamento nos bancos de dados de treino e de teste, esta métrica é consistentemente superior a 0,95 nas últimos passos de treinamento, enquanto o **AUC** esperado de um classificador independente das entradas vale 0,5. Assim, a performance do modelo é aparentemente superior a um classificador aleatório segundo a métrica **AUC**. Nota-se que esta pouco varia ao longo do treinamento e está na maior parte do treinamento próxima do seu melhor valor possível (1), enquanto as outras métricas variam relativamente bastante e estão mais longe do seu melhor valor. Estes resultados confirmam a falta de expressividade da métrica **AUC** para bancos de dados com classes desbalanceadas, explicada na subseção 2.2.1.

Analisando as métricas anteriores **AP** e **AUC**, independentes do limiar de segmentação, nota-se que ambas melhoram ao longo do treinamento como previsto pela análise da função de custo local. Usando como critério exclusivamente a métrica **AP**, o modelo GLIA-Net performa melhor no banco de dados de validação do desafio ADAM ao longo do treinamento no final da última época do treinamento no passo de treinamento 370001. As métricas obtidas no banco de dados de validação estão mostradas na Tabela 11.

Tabela 11 – Métricas do melhor resultado da solução GLIA-Net no banco de dados de validação do desafio ADAM ao longo do treinamento.

AP	AUC	DSC	HD95 [mm]	PV	RV	PT	RT
0,3670	0,9962	0,4243	9,7767	0,3322	0,5869	0,2934	0,5723

Nota: **AP**, **AUC**, **DSC**, **HD95**, **PV**, **RV**, **PT** e **RT** representam, respectivamente, a precisão média na curva precisão-revocação, área abaixo da curva ROC, o coeficiente de similaridade de Sorensen-Dice, distância de Hausdorff adaptada para 95% das distâncias, precisão no nível dos voxels, sensibilidade no nível dos voxels, precisão no nível dos conglomerados de voxels conexos e sensibilidade no nível dos conglomerados de voxels conexos.

Nota: O melhor resultado é o resultado do modelo GLIA-Net na época em que a métrica **AP** calculada no banco de dados de validação é a mais elevada durante todo treinamento.

4.2 Segmentação de exames de TOF angio-RM

Os resultados apresentados na seção 4.1 foram obtidos durante o treinamento, e portanto, os *patches* foram selecionados de forma a mitigar o desbalanceamento de classes. É necessário analisar os resultados em imagens completas já que na prática clínica os dados estão desbalanceados. Nesta subseção todos os resultados são calculados sem modificar o balanceamento dos dados e percorrendo com *patches* todas as partes das imagens.

Seguindo os passos descritos na seção 3.3, as métricas de segmentação e de detecção em função do limiar de segmentação são mostradas nas Figuras 45 e 46. Maximizando a métrica **DSC** (coeficiente de similaridade de Sorensen-Dice), obtemos um limiar igual a 0,624 e as métricas da Tabela 12.

Tabela 12 – Métricas do modelo GLIA-Net no passo 370001 no banco de dados de validação do desafio ADAM considerando a varredura completa da imagem dos exames e um peso maior aos voxels centrais na composição da máscara de segmentação a partir dos *patches* da imagem e escolhendo o limiar de segmentação que maximiza a métrica **DSC**.

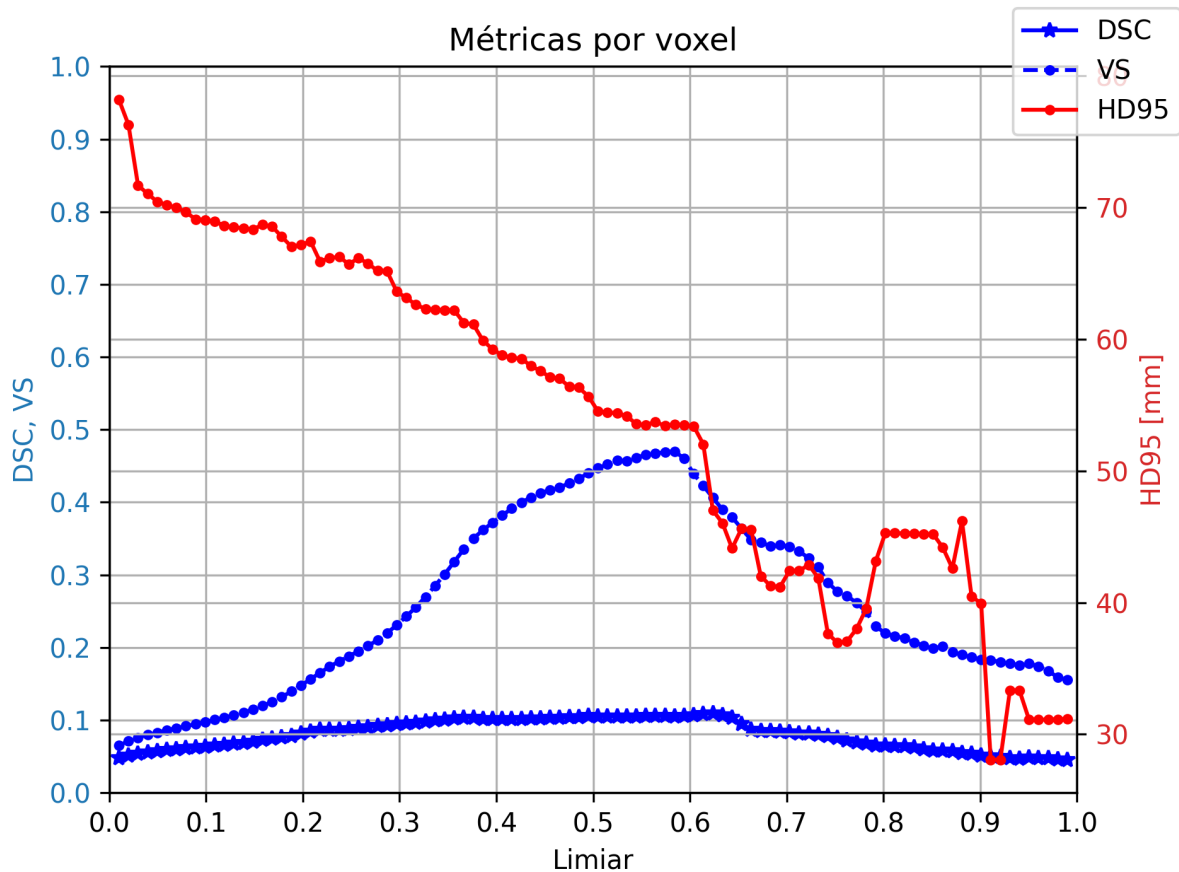
Limiar	FP/exame	RT	DSC	HD95 [mm]	VS
0,624	8,2	0,447	0,110	47,006	0,406

Nota: **DSC**, **HD95**, **VS**, **RT** e **FP/exame** representam, respectivamente, o coeficiente de similaridade de Sorensen-Dice, distância de Hausdorff adaptada para 95% das distâncias, a similaridade volumétrica, sensibilidade no nível dos conglomerados de voxels conexos e número de falso-positivos por exame no nível dos conglomerados de voxels conexos.

Observe que os resultados apresentados na Tabela 12 são equiparáveis aos resultados dos últimos colocados no desafio ADAM (Tabela 2) e são inferiores aos resultados do modelo GLIA-Net treinado em imagens de ATC (Tabela 3). As razões para estas diferenças são que o modelo ainda está treinando, performances elevadas exigem não só redes neurais do estado da arte como também da adaptação do configurações dos hiper-

parâmetros do modelo e do fluxo de dados ao banco de dados específico da tarefa, como descrito em (ISENSE et al., 2019) e na subseção 2.2.2. Além disso, a presença de aneurismas cerebrais tratados ou rompidos considerados como regiões normais pode aumentar a dificuldade do desafio em relação ao treinamento da rede na modalidade ATC. Outra possibilidade é que o congelamento das camadas dos *encoders* e a inicialização aleatória do primeiro *encoder* tenha adicionado um ruído na rede reduzido a capacidade de aprendizado do modelo.

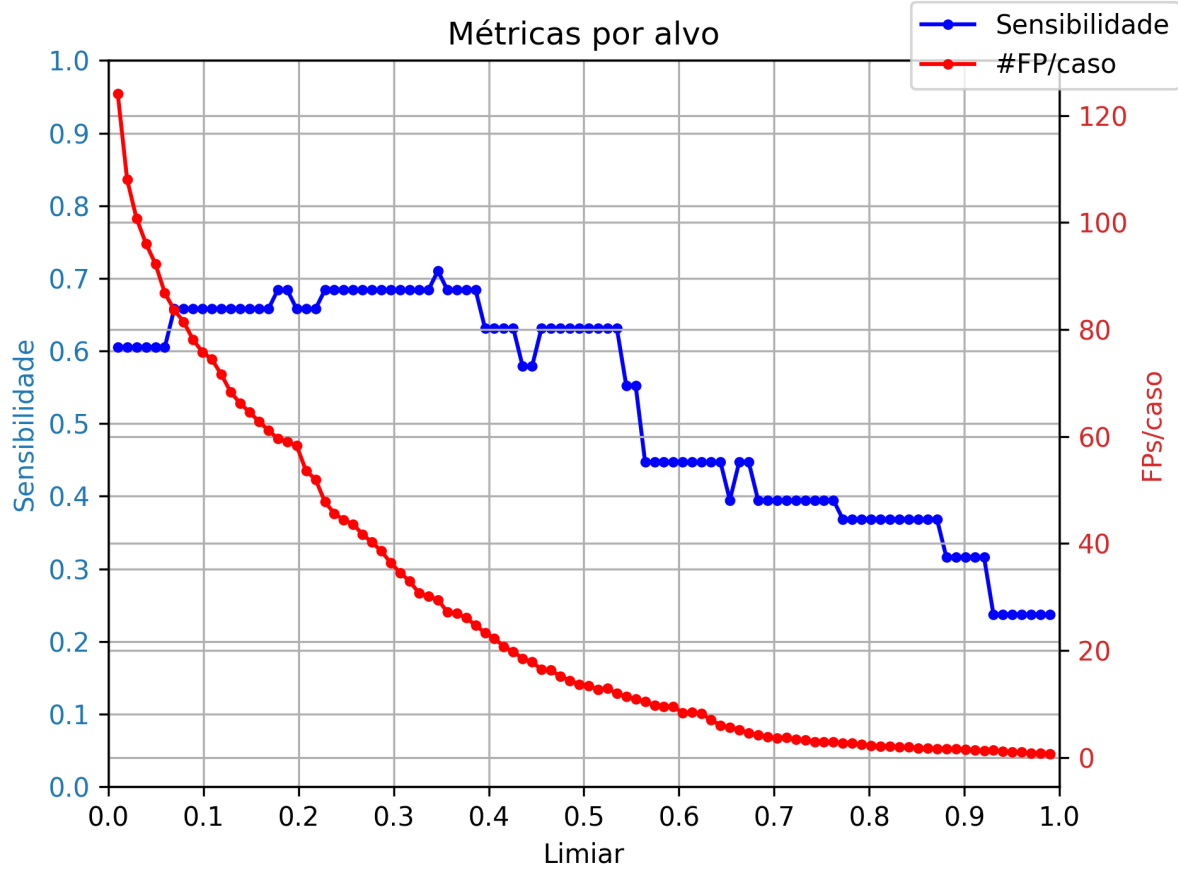
Figura 45 – Métricas de segmentação do modelo GLIA-Net no passo 370001 para diferentes valores de limiar de segmentação calculadas no banco de dados de validação do desafio ADAM dando um peso maior aos voxels centrais na composição da máscara de segmentação a partir dos *patches* da imagem.



Fonte: Autor.

Nota: **DSC**, **HD95** e **VS** representam, respectivamente, o coeficiente de similaridade de Sorensen-Dice, distância de Hausdorff adaptada para 95% das distâncias e a similaridade volumétrica.

Figura 46 – Métricas de detecção do modelo GLIA-Net no passo 370001 para diferentes valores de limiar de segmentação calculadas no banco de dados de validação do desafio ADAM dando um peso maior aos voxels centrais na composição da máscara de segmentação a partir dos *patches* da imagem.



Fonte: Autor.

Nota: **FP/caso** representa o número de falso-positivos por exame no nível dos conglomerados de voxels conexos.

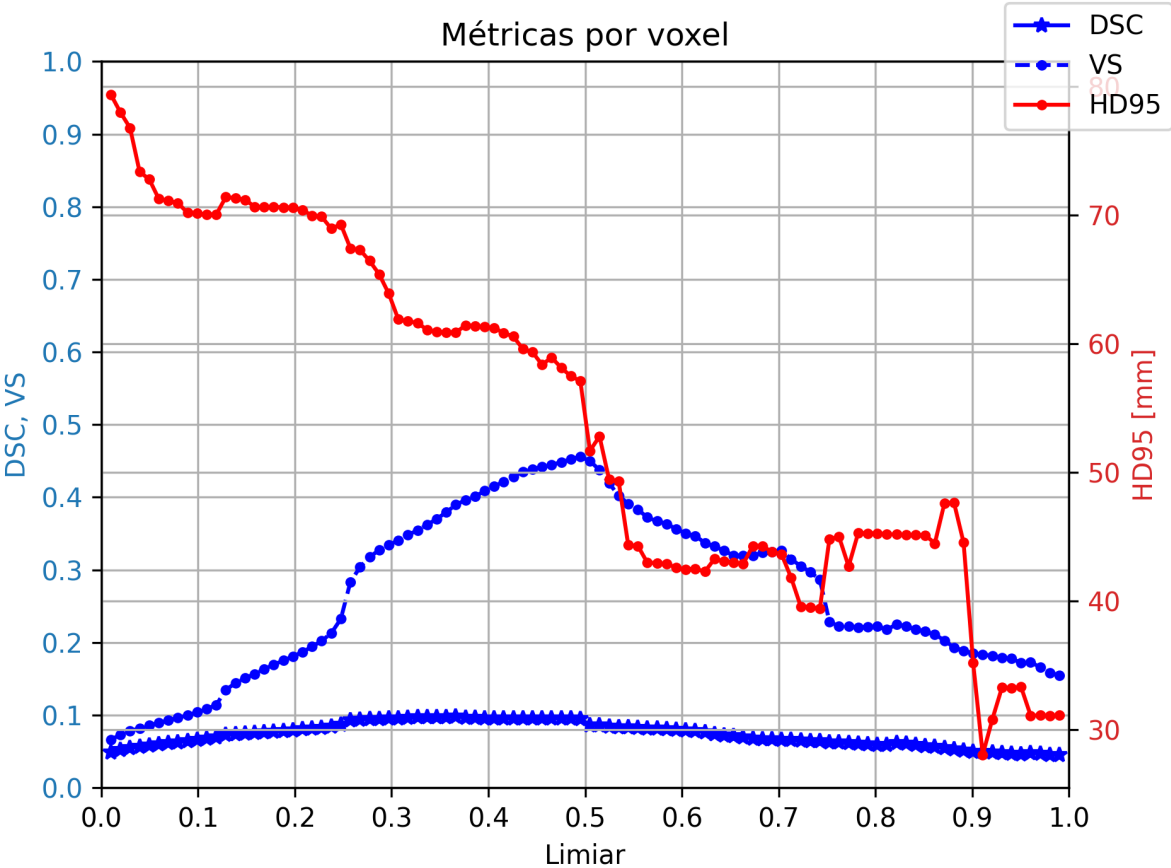
Na Tabela 13 e nas Figuras 47 e 48, um processo análogo à geração da Tabela 12 e as Figuras 45 e 46 é realizado. A única diferença é que nas primeiras imagens e e na primeira tabela apresentada a composição da máscara de segmentação considera um tensor de pesos para os *patches* que varia linearmente com a posição do voxel no *patch* e que se concentra no centro do *patch* enquanto nas imagens posteriores e na tabela posterior a composição da máscara de segmentação considera um tensor de pesos uniforme, como no projeto GLIA-Net original (BO et al., 2021).

Tabela 13 – Métricas do modelo GLIA-Net no passo 370001 no banco de dados de validação do desafio ADAM considerando a varredura completa da imagem dos exames e um peso constante para todos os voxels na composição da máscara de segmentação a partir dos *patches* da imagem e escolhendo o limiar de segmentação que maximiza a métrica **DSC**.

Limiar	DSC	HD95 [mm]	VS	RT	FP/exame
0,366	0,098	60,906	0,390	0,605	19,7

Nota: **DSC**, **HD95**, **VS**, **RT** e **FP/exame** representam, respectivamente, o coeficiente de similaridade de Sorensen-Dice, distância de Hausdorff adaptada para 95% das distâncias, a similaridade volumétrica, sensibilidade no nível dos conglomerados de voxels conexos e número de falso-positivos por exame no nível dos conglomerados de voxels conexos.

Figura 47 – Métricas de segmentação do modelo GLIA-Net no passo 370001 para diferentes valores de limiar de segmentação calculadas no banco de dados de validação do desafio ADAM dando um peso igual para todos os voxels na composição da máscara de segmentação a partir dos *patches* da imagem.



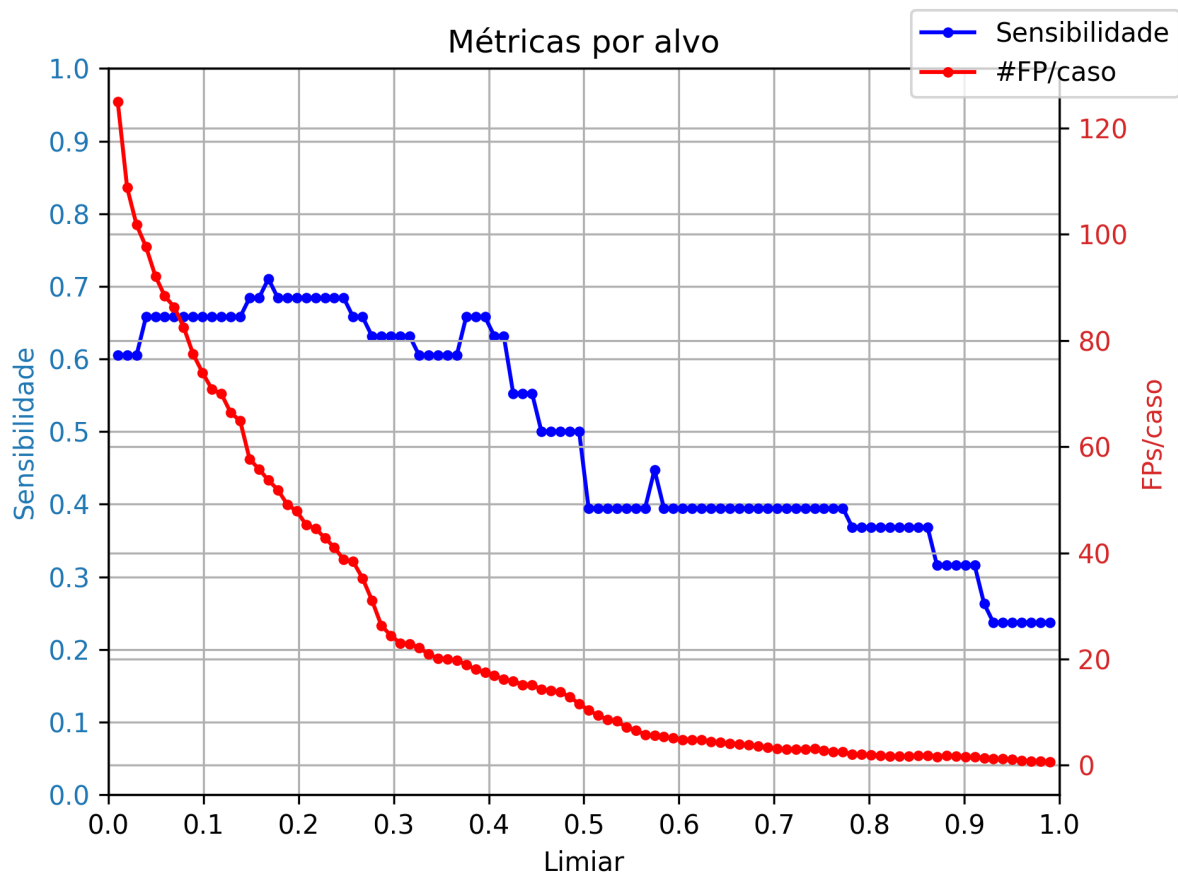
Fonte: Autor.

Nota: **DSC**, **HD95** e **VS** representam, respectivamente, o coeficiente de similaridade de Sorensen-Dice, distância de Hausdorff adaptada para 95% das distâncias e a similaridade volumétrica.

Comparando as Tabelas 12 e 13, nota-se que o emprego de uma interpolação com

pesos variando linearmente da máscara de segmentação dos *patches* permite a obtenção de um coeficiente de similaridade de Sorensen-Dice máximo superior aos resultados com uma interpolação com pesos uniformes. A Figura 45 aponta uma certa regularidade deste fenômeno, pois entre os limiares 0,37 e 0,62 a métrica **DSC** para pesos variando linearmente é consistentemente superior ao maior maior desta métrica para pesos uniformemente distribuídos. Além disso, todas as métricas usadas no desafio exceto a métrica sensibilidade por alvo (**RT**) para o limiar que maximiza a métrica **DSC** são superiores para pesos variando linearmente quando comparados a pesos uniformemente distribuídos.

Figura 48 – Métricas de detecção do modelo GLIA-Net no passo 370001 para diferentes valores de limiar de segmentação calculadas no banco de dados de validação do desafio ADAM dando um peso constante para todos os voxels na composição da máscara de segmentação a partir dos *patches* da imagem.



Fonte: Autor.

Nota: **FP/caso** representa o número de falso-positivos por exame no nível dos conglomerados de voxels conexos.

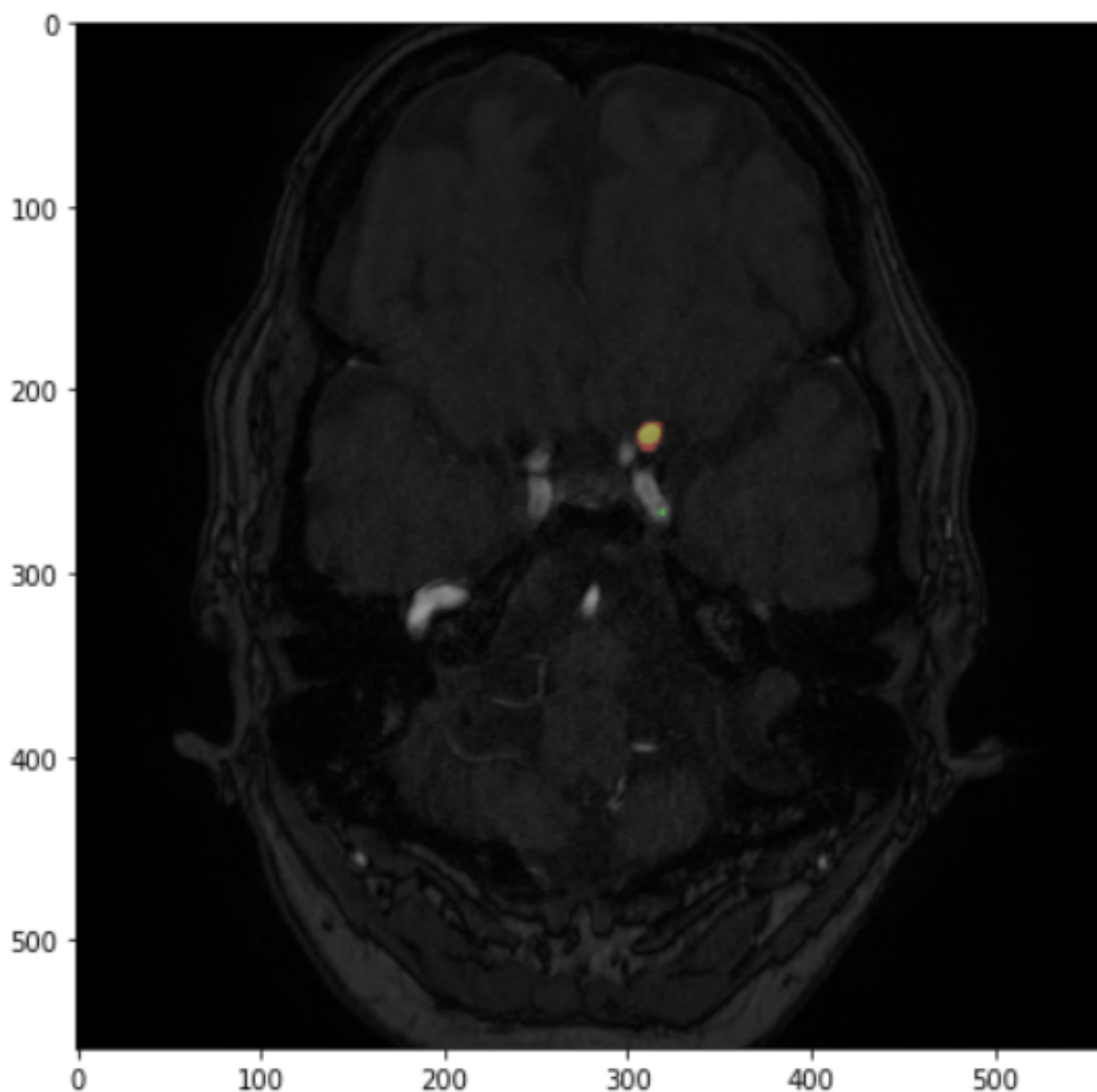
Ao comparar as Figuras 46 e 48, é perceptível que o uso de um tensor de pesos distribuídos linearmente eleva o número de conjuntos de voxels vizinhos conectados previstos como aneurismas não tratados nem rompidos quando comparado a um tensor homogêneo nos mesmo valores de limiar de segmentação, pois tanto a sensibilidade quanto a quan-

tidade de falso-positivos são elevadas com a variação linear dos pesos para os mesmos limiares.

Nota-se também que os limiares que maximizam a métrica **DSC** apresentam uma diferença superior a 0,2. Uma possível justificativa para o limiar maior do 0,5 para tensor linearmente variável é que o modelo foi treinado com uma proporção de casos positivos (aneurismas) em relação a casos negativos (normais) superior à proporção real dos exames, logo o modelo é exageradamente confiante nas previsões dos aneurismas e um limiar de segmentação mais alto filtra indica uma redução na confiança do modelo na classificação de aneurismas. Já para o tensor homogêneo, uma justificativa é que para todos voxels que não pertencem aos *patches* da borda da imagem, que são a maioria, oito *patches* são ponderados igualmente para obter a previsão total do voxel. Lembrando que no treinamento o centro dos aneurismas está a uma distância máxima de 28 voxels do centro do *patch* em cada direção, o voxel em questão deve cair a uma distância acima desta distância de 28 voxels do centro do *patch*, e portanto, estes *patches* possuem um viés de reduzir a probabilidade do voxel ser classificado como um aneurisma, logo o modelo é pouco confiante em classificar voxels como aneurismas e o limiar de segmentação escolhido é menor que 0,5 para compensar este viés.

Nas Figuras 49 e 50, estão exibidos um exame de TOF angio-RM do banco de dados de validação com a anotação do aneurisma não tratado nem rompido em vermelho e as máscaras de segmentação previstas pelo algoritmo. Observe que há um aneurisma cerebral corretamente detectado na Figura 49 e uma outra região que o algoritmo previu erroneamente como aneurismas. Além disso, nota-se que a segmentação não é perfeita e que o aneurisma é mais cauteloso que o necessário, pois deixa alguns voxels do aneurisma previsto como falso-negativos no nível dos voxels. Na Figura 50, este método de gerar máscaras de segmentação não é capaz de prever com detalhes a região do aneurisma, pois só recebe como entrada a imagem global reduzida dimensionalmente *global inputs*. Todavia, isto permite compreender a atuação da rede *global localizer network* de maneira distinta da rede *local segmentation network*. Note que a região destacada com maior probabilidade de apresentar aneurismas na Figura 50 segue uma distribuição semelhante a da Figura 20, o que levanta o questionamento se as máscaras de segmentação geradas pela rede *global localizer network* são individualizadas ou se capturam a distribuição do banco de dados como um todo.

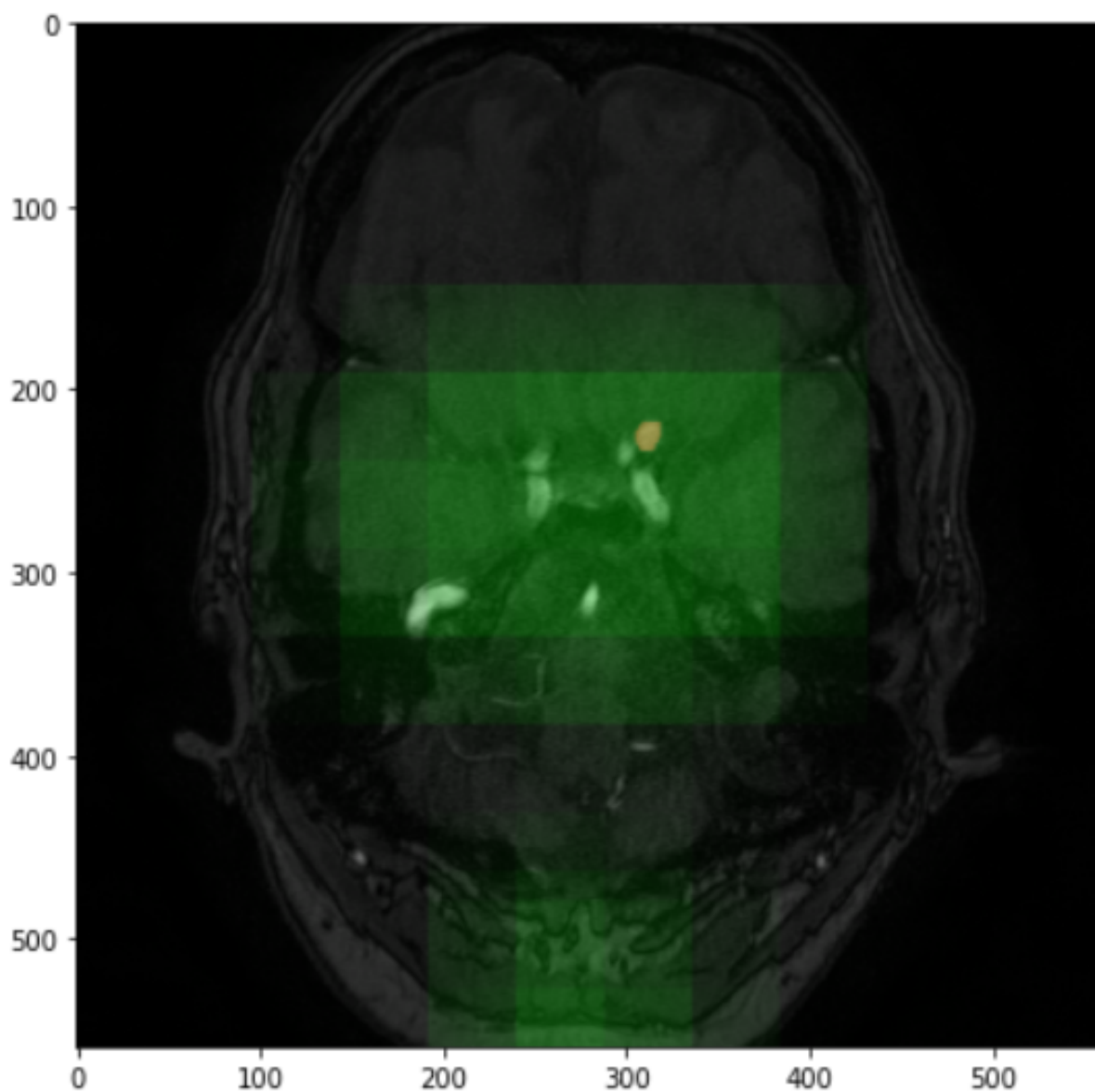
Figura 49 – Imagem de TOF angio-RM do exame 10063F do banco de dados de validação do desafio ADAM anotada e segmentada pelo modelo GLIA-Net treinado usando a saída *net* e o limiar de segmentação 0,624.



Fonte: Autor.

Nota: As regiões verdes são falso-positivos, as regiões vermelhas são falso-negativos, as regiões amarelas são verdadeiro-positivos e as regiões incolores são verdadeiro-negativos.

Figura 50 – Imagem de TOF angio-RM do exame 10063F do banco de dados de validação do desafio ADAM anotada e segmentada pelo modelo GLIA-Net treinado usando a saída *global localizer logits*.



Fonte: Autor.

Nota: As regiões verdes são falso-positivos, as regiões vermelhas são falso-negativos e as regiões amarelas são verdadeiro-positivos. regiões incolores são verdadeiro-negativos.

5 Conclusão

Neste trabalho, o projeto GLIA-Net desenvolvido para segmentar aneurismas cerebrais na modalidade ATC foi adaptado para realizar a mesma tarefa na modalidade TOF angio-RM. Este modelo foi treinado em um banco de dados com 113 imagens e 124 aneurismas craniais não tratados nem rompidos e os resultados apontam que a performance do algoritmo ainda está melhorando, porém a performance já é equiparável aos algoritmos com as piores notas no desafio.

A proposta de uma nova forma de compor a máscara de segmentação mostrou ser capaz de melhorar a performance geral do modelo em todas as métricas exceto na sensibilidade por vizinhança de voxels conectados e classificados como aneurismas não tratados nem rompidos.

Os resultados de segmentação do modelo treinado em imagens completas de TOF angio-RM foram realizados em um banco de dados de validação com 23 imagens. A escolha do passo de treinamento do modelo treinado e do limiar de segmentação foram baseadas em resultados neste banco de dados. Para aumentar a robustez e permitir a comparação deste modelo com outras soluções do desafio, deve-se testar a solução GLIA-Net adaptada para imagens de ressonância magnética no banco de dados secreto do desafio ADAM.

Como sugestões para projetos futuros, recomenda-se continuar o treinamento do modelo GLIA-Net nas mesmas configurações deste trabalho até atingir a estabilização das métricas no banco de dados de treinamento e obter *overfitting* para a função de custo local e para a função de custo global. Visando mitigar o *overfitting* da rede *global localizer network*, deve-se adicionar novas transformações aleatórias à etapa de aumento de dados, como mudança de espaçamento e rotação. Durante o treinamento do modelo GLIA-Net neste trabalho todas as *encoders* da rede *local segmentation network* foram congelados, seria interessante treinar o modelo com estas camadas descongeladas e avaliar se há um diferença de performance. A implementação nnU-Net foi capaz de adaptar o fluxo de dados e os hiperparâmetros da rede U-Net de modo a atingir o estado da arte em diversas tarefas, logo deve-se estudar a metodologia usada nesta solução e adaptá-la para a rede GLIA-Net. Por último, para aumentar a robustez do modelo sugere-se separar o banco de dados disponibilizado em cinco grupos e realizar um treinamento por validação cruzada. Ao final do treinamento dos cinco modelos, deve-se implementar um algoritmo de votação majoritária no nível dos voxels.

Referências

- ABC. *A Brief Technical Report on Our Submitted Model*. 2020. <<https://adam.isi.uu.nl/wp-content/uploads/2020/10/JunMa-Method-ADAM-2020.pdf>>. Acessado: 2021-11-07.
- ALLENBY, M. C. et al. Detection of clustered anomalies in single-voxel morphometry as a rapid automated method for identifying intracranial aneurysms. *bioRxiv*, Cold Spring Harbor Laboratory, 2020.
- BACH, F. R.; JORDAN, M. I. *A probabilistic interpretation of canonical correlation analysis*. [S.l.], 2005.
- BO, Z.-H. et al. Toward human intervention-free clinical diagnosis of intracranial aneurysm via deep neural network. *Patterns*, v. 2, n. 2, p. 100197, 2021. ISSN 2666-3899.
- ÇİÇEK, Ö. et al. 3d u-net: Learning dense volumetric segmentation from sparse annotation. *CoRR*, abs/1606.06650, 2016.
- DANCE, D. R. et al. *Diagnostic Radiology Physics*. Vienna: International Atomic Energy Agency, 2014. 259-261 p. (Non-serial Publications). ISBN 978-92-0-131010-1.
- ETMINAN, N. et al. Worldwide Incidence of Aneurysmal Subarachnoid Hemorrhage According to Region, Time Period, Blood Pressure, and Smoking Prevalence in the Population: A Systematic Review and Meta-analysis. *JAMA Neurology*, v. 76, n. 5, p. 588–597, 05 2019. ISSN 2168-6149.
- FARON, A. et al. Performance of a deep-learning neural network to detect intracranial aneurysms from 3d tof-mra compared to human readers. *Clinical Neuroradiology*, v. 30, n. 3, p. 591–598, 2020.
- FAWCETT, T. An introduction to roc analysis. *Pattern Recognition Letters*, v. 27, n. 8, p. 861–874, 2006. ISSN 0167-8655. ROC Analysis in Pattern Recognition.
- HAJNAL, J. V. et al. Use of fluid attenuated inversion recovery (FLAIR) pulse sequences in MRI of the brain. *J Comput Assist Tomogr*, v. 16, n. 6, p. 841–844, 1992.
- HE, K. et al. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.
- HOWARD, B. et al. Comprehensive review of imaging of intracranial aneurysms and angiographically negative subarachnoid hemorrhage. *Neurosurgical Focus*, v. 47, n. 6, p. 1–13, 2019. Cited By 5.
- IOFFE, S.; SZEGEDY, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.
- ISENSEE, F. et al. nnu-net: Breaking the spell on successful medical image segmentation. *CoRR*, abs/1904.08128, 2019.
- JAEGER, P. F. et al. Retina u-net: Embarrassingly simple exploitation of segmentation supervision for medical object detection. *CoRR*, abs/1811.08661, 2018.

- KAKA, H.; ZHANG, E.; KHAN, N. Artificial intelligence and deep learning in neuroradiology: Exploring the new frontier. *Canadian Association of Radiologists Journal*, v. 72, n. 1, p. 35–44, 2021. PMID: 32946272.
- KAMNITSAS, K. et al. Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. *Medical Image Analysis*, v. 36, p. 61–78, 2017. ISSN 1361-8415.
- KARIMI, D.; WARFIELD, S. K.; GHOLIPOUR, A. Critical assessment of transfer learning for medical image segmentation with fully convolutional neural networks. *CoRR*, abs/2006.00356, 2020.
- KAVUR, A. E. et al. Chaos challenge - combined (ct-mr) healthy abdominal organ segmentation. *Medical Image Analysis*, Elsevier BV, v. 69, p. 101950, Apr 2021. ISSN 1361-8415. Disponível em: <<http://dx.doi.org/10.1016/j.media.2020.101950>>.
- KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. In: BENGIO, Y.; LECUN, Y. (Ed.). *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. [S.l.: s.n.], 2015.
- KUIJF, H. J. *hjkuijf/ADAMChallenge*. 2020. <<https://adam.isi.uu.nl/>>. Acessado: 2021-09-20.
- KUO, A. H. et al. Vascular magnetic resonance angiography techniques. *Cardiovascular Diagnosis and Therapy*, v. 0, n. 0, 2019. ISSN 2223-3660.
- LI, J.; CHEN, B. M.; LEE, G. H. So-net: Self-organizing network for point cloud analysis. *CoRR*, abs/1803.04249, 2018.
- LI, Y. et al. Pointcnn. *CoRR*, abs/1801.07791, 2018.
- LIU, X. et al. Deep neural network-based detection and segmentation of intracranial aneurysms on 3d rotational dsa. *Interventional Neuroradiology*, v. 0, n. 0, p. 15910199211000956, 2021. PMID: 33715500.
- MA, J. *Loss Ensembles for Intracranial Aneurysm Segmentation: An Embarrassingly Simple Method*. 2020. <<https://adam.isi.uu.nl/wp-content/uploads/2020/10/JunMa-Method-ADAM-2020.pdf>>. Acessado: 2021-11-07.
- MAIER, A. et al. *Medical Imaging Systems An Introductory Guide*. 1st ed. 2018.. ed. Cham: Springer International Publishing, 2018. (Image Processing, Computer Vision, Pattern Recognition, and Graphics ; 11111). ISBN 3-319-96520-4.
- MAURER, M. et al. Time requirement and feasibility of a systematic quality peer review of reporting in radiology. *RoFo Fortschritte auf dem Gebiet der Rontgenstrahlen und der Bildgebenden Verfahren*, v. 193, n. 2, p. 160–167, 2021.
- MCDONALD, R. et al. The effects of changes in utilization and technological advancements of cross-sectional imaging on radiologist workload. *Academic Radiology*, v. 22, n. 9, p. 1191–1198, 2015.

- NAKAO, T. et al. Deep neural network-based computer-assisted detection of cerebral aneurysms in mr angiography. *Journal of Magnetic Resonance Imaging*, v. 47, n. 4, p. 948–953, 2018.
- NIEUWKAMP, D. J. et al. Changes in case fatality of aneurysmal subarachnoid haemorrhage over time, according to age, sex, and region: a meta-analysis. *The Lancet Neurology*, v. 8, n. 7, p. 635–642, 2009. ISSN 1474-4422.
- NOTO, T. D. et al. *Unil-CHUV Team, Lausanne, Switzerland, Task 1*. 2020. <https://adam.isi.uu.nl/results/results-live-leaderboard/unil_chuv2/>. Acessado: 2021-11-07.
- NOTO, T. D. et al. *Unruptured Intracranial Aneurysm Segmentation from TOF-MRA Images Using Cascaded 3D Convolutional Neural Networks*. 2020. <https://adam.isi.uu.nl/wp-content/uploads/2020/10/stronger_poster.pdf>. Acessado: 2021-11-07.
- PAN, S. J.; YANG, Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, v. 22, n. 10, p. 1345–1359, 2010.
- PARK, A. et al. Deep Learning-Assisted Diagnosis of Cerebral Aneurysms Using the HeadXNet Model. *JAMA Network Open*, v. 2, n. 6, p. e195600–e195600, 06 2019. ISSN 2574-3805.
- QI, C. R. et al. Pointnet: Deep learning on point sets for 3d classification and segmentation. *CoRR*, abs/1612.00593, 2016.
- QI, C. R. et al. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *CoRR*, abs/1706.02413, 2017.
- RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: NAVAB, N. et al. (Ed.). *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Cham: Springer International Publishing, 2015. p. 234–241. ISBN 978-3-319-24574-4.
- ROY, A. G. et al. Quicknat: Segmenting MRI neuroanatomy in 20 seconds. *CoRR*, abs/1801.04161, 2018.
- SAILER, A. M. et al. Diagnosing intracranial aneurysms with mr angiography. *Stroke*, v. 45, n. 1, p. 119–126, 2014.
- SAITO, T.; REHMSMEIER, M. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE*, Public Library of Science, v. 10, n. 3, p. 1–21, 03 2015.
- SETTECASE, F.; RAYZ, V. Advanced vascular imaging techniques. *Handbook of Clinical Neurology*, v. 176, p. 81–105, 2021.
- SHIMIZU, A. et al. Automated measurement of bone scan index from a whole-body bone scintigram. *International Journal of Computer Assisted Radiology and Surgery*, v. 15, Mar 2020.

- SICHTERMANN, T. et al. Deep learning–based detection of intracranial aneurysms in 3d tof-mra. *American Journal of Neuroradiology*, American Journal of Neuroradiology, v. 40, n. 1, p. 25–32, 2019. ISSN 0195-6108.
- SUINESIAPUTRA, A. et al. Quantification of lv function and mass by cardiovascular magnetic resonance: Multi-center variability and consensus contours. *Journal of Cardiovascular Magnetic Resonance*, v. 17, n. 1, 2015. Cited By 89.
- TAHA, A. A.; HANBURY, A. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC Medical Imaging*, v. 15, 2015.
- TAN, C. et al. A survey on deep transfer learning. *CoRR*, abs/1808.01974, 2018.
- TAN, M.; LE, Q. V. Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019.
- TIMMINS, K. et al. *Intracranial Aneurysm Detection and Segmentation Challenge*. [S.l.]: Zenodo, 2020.
- UEDA, D. et al. Deep learning for mr angiography: Automated detection of cerebral aneurysms. *Radiology*, v. 290, n. 1, p. 187–194, 2019. PMID: 30351253.
- UTRECHT, U. M. C. *ADAM Aneurysm Detection And segMentation Challenge*. 2020. <<https://adam.isi.uu.nl/>>. Acessado: 2021-05-06.
- van Gijn, J.; KERR, R. S.; RINKEL, G. J. Subarachnoid haemorrhage. *The Lancet*, v. 369, n. 9558, p. 306–318, 2007. ISSN 0140-6736.
- WANG, Y. et al. Dynamic graph CNN for learning on point clouds. *CoRR*, abs/1801.07829, 2018.
- WU, Y.; HE, K. Group normalization. *CoRR*, abs/1803.08494, 2018.
- XU, B. et al. Empirical evaluation of rectified activations in convolutional network. *CoRR*, abs/1505.00853, 2015.
- XU, Y. et al. Spidercnn: Deep learning on point sets with parameterized convolutional filters. *CoRR*, abs/1803.11527, 2018.
- YANG, X. et al. Intra: 3d intracranial aneurysm dataset for deep learning. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2020. p. 2653–2663.
- YANG, Y. L. Y. *3D U-Net Ensembles*. 2020. <https://adam.isi.uu.nl/wp-content/uploads/2020/10/Joker_description.pdf>. Acessado: 2021-11-07.
- ZHAO, J. et al. Current treatment strategies for intracranial aneurysms: An overview. *Angiology*, v. 69, n. 1, p. 17–30, 2018. PMID: 28355880.