

UNIVERSIDADE DE SÃO PAULO  
FACULDADE DE MEDICINA DE RIBEIRÃO PRETO

MARIANA AYA SUZUKI UCHIDA

TRABALHO DE CONCLUSÃO DE CURSO:  
Estudo de Modelo de U-net Bidimensional para Realizar  
Segmentação Automatizada de Corpos Vertebrais Fraturados  
em Exames de Ressonância Magnética

PROF. DR. PAULO MAZZONCINI DE AZEVEDO MARQUES

SÃO PAULO - RIBEIRÃO PRETO

2022



MARIANA AYA SUZUKI UCHIDA

## Trabalho de Conclusão de Curso:

Estudo de Modelo de U-net Bidimensional para Realizar  
Segmentação Automatizada de Corpos Vertebrais Fraturados  
em Exames de Ressonância Magnética

Trabalho de Conclusão de Curso para  
a disciplina RIB0114 - Trabalho de  
Conclusão de Curso para obtenção de  
aprovação.

Área de Concentração: Informá-  
tica em Imagens Médicas

Orientador: Prof. Dr. Paulo Mazzoncini  
de Azevedo Marques

SÃO PAULO - RIBEIRÃO PRETO

2022

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

# AGRADECIMENTOS

Agradeço primeiramente minha família e amigos por todo o apoio e amor recebidos por eles. Quero agradecer por todos os ensinamentos que me acompanham até hoje.

Ao Prof. Dr. Paulo Mazzoncini de Azevedo Marques, ao Mestre Lucas Lins e à Doutora Natália Santana Chiari Correia por todo o auxílio recebido para a conclusão deste projeto. Muito obrigada por todos o aprendizado e paciência.



# RESUMO

A ocorrência e diagnóstico de fraturas vertebrais muitas vezes podem ser subnotificados, seja por falta de sintomas ou profissionais disponíveis. Dado isso surge a opção de inclusão de sistemas de auxílio a diagnóstico através da implementação de algoritmos de aprendizado de máquina para atuarem como segunda opinião e permitindo um melhor diagnóstico, reduzindo também o desgaste do profissional. Para o desenvolvimento deste projeto, foi estudado o framework U-net para realização automatizada de segmentação de corpos vertebrais fraturados de exames de ressonância magnética provenientes de pacientes ( 36 homens e 55 mulheres, com idade média de  $64.24 \pm 11.75$  anos) atendidos pelo HCMFRP-USP. Utilizou-se do ambiente Jupyter e técnicas de aumento de base para o treino de modelos que depois tiveram suas performances generalizadas, obtendo uma performance de até 0.89 para o coeficiente de Jaccard e 0.99 para o coeficiente de Dice na etapa de treinamento e validação; e 0,74 e 0,62 respectivamente para os coeficientes de Jaccard e Dice na etapa de teste independente.

**Palavras-chave:** Corpos vertebrais, Ressonância Magnética, U-Net





# LISTA DE ILUSTRAÇÕES

Figura 1 – <b>Workflow de deep learning para detecção de fraturas</b> . . . . .	14
Figura 2 – <b>Diagrama de Venn sobre a relação entre Inteligência Artificial, Machine Learning e Deep Learning</b> . . . . .	15
Figura 3 – <b>Arquitetura da U-net</b> . . . . .	16
Figura 4 – <b>Representação de imagem digital para a forma de matriz de níveis de cinza</b> . . . . .	18
Figura 5 – <b>Representação de imagem digital para a forma de matriz de níveis de cinza</b> . . . . .	19
Figura 6 – <b>Desenvolvimento de Modelo de Machine Learning</b> . . . . .	21
Figura 7 – <b>Esquema das etapas da metodologia</b> . . . . .	23
Figura 8 – <b>Gráfico dos resultados originais e da generalização</b> . . . . .	28
Figura 9 – <b>Predição feita por modelo com aumento de base</b> . . . . .	29
Figura 10 – <b>Predição feita por modelo sem aumento de base</b> . . . . .	29
Figura 11 – <b>Coeficientes de Jaccard e Dice após aplicação de modelo com e sem aumento de base</b> . . . . .	30
Figura 12 – <b>Caso em que o corpo é detectado, porém segmentado incorretamente</b>	30
Figura 13 – <b>Caso em que estruturas vizinhas também são segmentadas</b> . . . . .	31
Figura 14 – <b>Caso em que a predição melhor se aproxima da máscara verdadeira</b>	31

# LISTA DE TABELAS

Tabela 1 – <b>Todos os modelos feitos e seus resultados</b> . . . . .	27
Tabela 2 – <b>Resultado da aplicação dos modelos em outras imagens</b> . . . . .	28

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>13</b>
<b>1.1</b>	<b>Contextualização</b>	<b>13</b>
<b>1.2</b>	<b>Machine Learning e Deep Learning</b>	<b>13</b>
1.2.1	U-net	15
<b>1.3</b>	<b>Justificativa</b>	<b>17</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>18</b>
<b>2.1</b>	<b>Fraturas Vertebrais por Compressão</b>	<b>18</b>
<b>2.2</b>	<b>Reconhecimento de Padrões em Imagens Médicas</b>	<b>19</b>
2.2.1	Imagem Médica Digital	19
2.2.2	Etapas do reconhecimento de padrões em imagens	19
2.2.2.1	Aquisição da Imagem	19
2.2.2.2	Segmentação	20
2.2.2.3	Extração de Atributos	20
2.2.2.4	Extração de Atributos Relevantes	20
2.2.2.5	Criação do Modelo	21
<b>3</b>	<b>OBJETIVO</b>	<b>22</b>
<b>4</b>	<b>MATERIAIS E MÉTODOS</b>	<b>23</b>
<b>4.1</b>	<b>Base de Dados</b>	<b>23</b>
<b>4.2</b>	<b>Pré-processamento dos dados</b>	<b>24</b>
<b>4.3</b>	<b>Aumento de Base</b>	<b>24</b>
<b>4.4</b>	<b>Construção da Rede</b>	<b>25</b>
<b>4.5</b>	<b>Treino e Teste da Rede</b>	<b>25</b>
<b>4.6</b>	<b>Avaliação dos Resultados</b>	<b>25</b>
<b>5</b>	<b>RESULTADOS E DISCUSSÃO</b>	<b>27</b>
<b>5.1</b>	<b>Generalização dos Resultados</b>	<b>27</b>
<b>5.2</b>	<b>Eficiência do Aumento de Base</b>	<b>29</b>
<b>5.3</b>	<b>Resultado das Predições</b>	<b>29</b>

<b>6</b>	<b>CONCLUSÃO . . . . .</b>	<b>33</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>34</b>
	 <b>APÊNDICES</b>	 <b>37</b>
	<b>APÊNDICE A – CÓDIGO DESENVOLVIDO . . . . .</b>	<b>38</b>
<b>A.1</b>	<b>Importação de Bibliotecas . . . . .</b>	<b>38</b>
<b>A.2</b>	<b>Carregar Base de Dados . . . . .</b>	<b>38</b>
<b>A.3</b>	<b>Pré-Processamento . . . . .</b>	<b>38</b>
<b>A.4</b>	<b>Aumento de base . . . . .</b>	<b>39</b>
<b>A.5</b>	<b>Construção da Rede . . . . .</b>	<b>39</b>
<b>A.6</b>	<b>Treinamento . . . . .</b>	<b>40</b>
<b>A.7</b>	<b>Carregar Modelo . . . . .</b>	<b>41</b>
<b>A.8</b>	<b>Teste . . . . .</b>	<b>41</b>
<b>A.9</b>	<b>Avaliação . . . . .</b>	<b>41</b>

# 1 INTRODUÇÃO

## 1.1 Contextualização

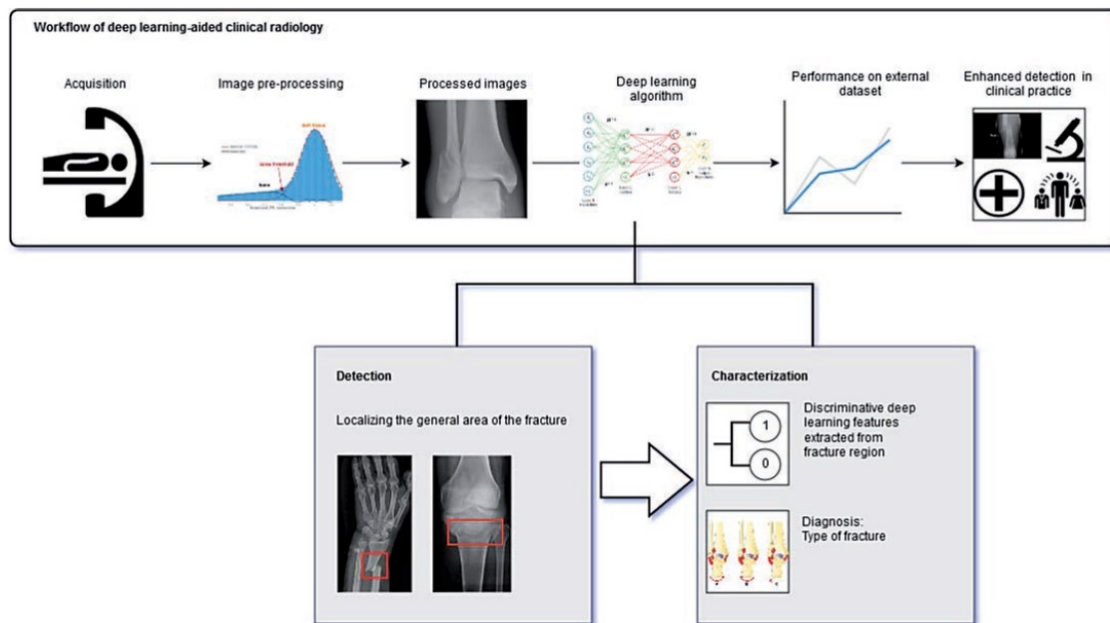
Fraturas vertebrais podem ocorrer de diversas formas, seja devido à osteoporose ou câncer; ao contrário da maior parte das fraturas, grande parte das fraturas vertebrais não são diagnosticadas no momento em que ocorrem. Apenas um quarto dos casos são clinicamente diagnosticados (ENSRUD; SCHOUSBOE, 2011). Esse tipo de fratura consta como um dos tipos de fratura mais prevalentes e a falta de diagnóstico se dá pelo fato de que fraturas vertebrais são assintomáticas ou causar incômodo tolerável, além de falta de uma rotina de detecções radiográficas (CHEN et al., 2021).

Na maioria das vezes, os diagnósticos podem precisar da atuação de diversos profissionais além de cirurgiões ortopédicos, como a opinião de radiologistas; estes profissionais, por sua vez, podem estar indisponíveis em horários de pico ou hospitais de menor escala. Dessa forma, a implementação de modelos de aprendizado de máquina é uma alternativa útil (LI et al., 2021) como um modo de 'segunda opinião'. De modo geral, técnicas de deep learning em conjunto com as práticas tradicionais da radiologia têm o potencial de aprimorar a velocidade e acurácia dos diagnósticos, reduzindo o desgaste do profissional (KALMET et al., 2020).

## 1.2 Machine Learning e Deep Learning

Um algoritmo de machine ou deep learning é um processo computacional que utiliza dados informados para realizar uma determinada tarefa sem que essa seja literalmente programada para obter um resultado em particular (NAQA; MURPHY, 2022). Deep learning é uma classe de algoritmos dentro do campo do machine learning, abrangendo análise de big data, segmentação de imagens e outros métodos. Observa-se na Figura 1 o workflow geral do uso de deep learning na detecção de fraturas.

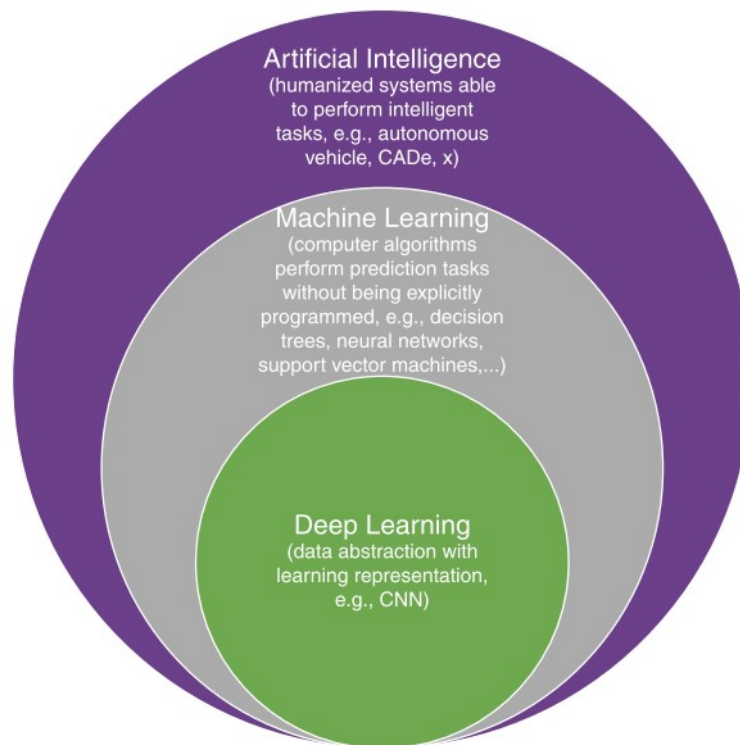
Figura 1 – Workflow de deep learning para detecção de fraturas



**Fonte:** (KALMET et al., 2020)

O ideal por trás do Machine Learning é simular o processo de pensamento humano. Tradicionalmente, um algoritmo de machine learning alimentaria características provenientes de dados crus obtidos através de padrões extraídos por computador. Esse processo contrasta com a sub-categoria Deep Learning, que possibilita a combinação de representação de dados e aprendizado de tarefas (NAQA; MURPHY, 2022). A Figura 2 ilustra a relação entre Inteligência Artificial, Machine Learning e Deep Learning, definindo Deep Learning como a coleta de dados com representação do aprendizado; Machine Learning como os algoritmos que performam tarefas de predição sem programação explícita e Inteligência Artificial como sistemas humanizados capazes de realizar tarefas inteligentes como veículos autônomos.

Figura 2 – Diagrama de Venn sobre a relação entre Inteligência Artificial, Machine Learning e Deep Learning

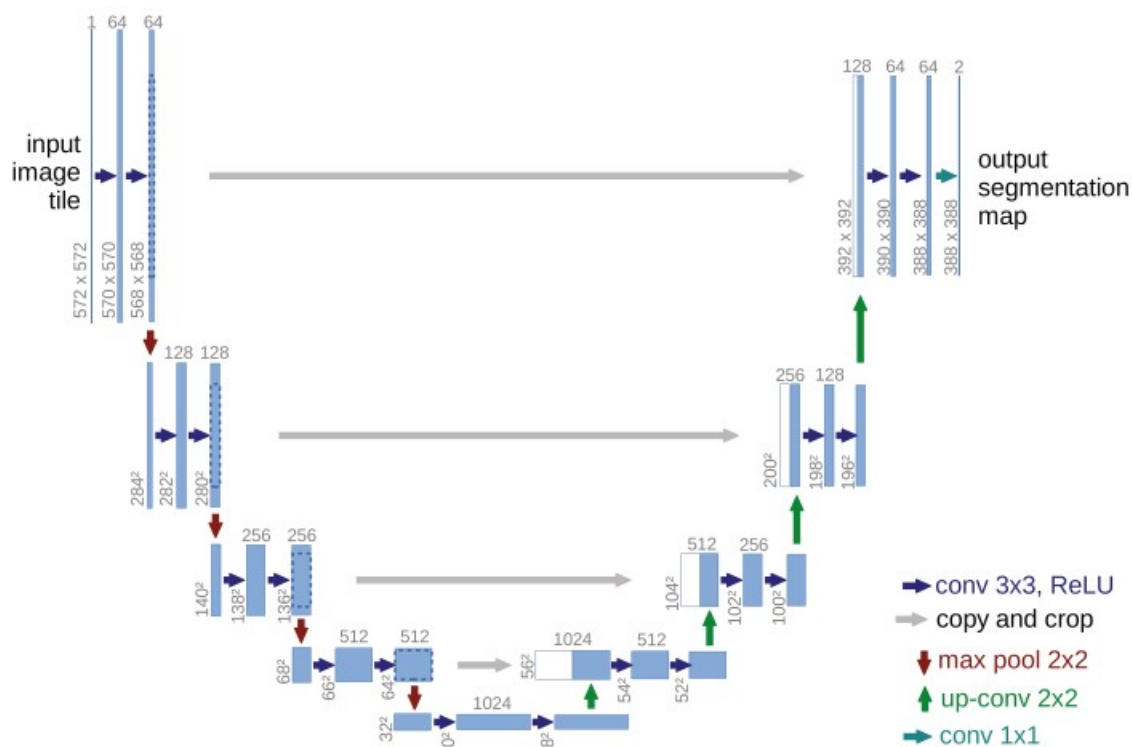


**Fonte:** (NAQA; MURPHY, 2022)

### 1.2.1 U-net

Desenvolvida por Olaf Ronneberger U-net é a arquitetura de deep learning mais popular no âmbito de segmentação de imagens médicas (NAQA; MURPHY, 2022); desenvolvida primariamente para segmentação de imagens. Tendo por estrutura básica uma passagem de análise, similar a redes de convolução, e uma outra passagem para classificação de informações. A rede que resulta dessas operações possui formato quase simétrico, atingindo assim uma forma de 'U'. (SIDDIQUE et al., 2021). O esquema ilustrado pela Figura 3 demonstra o funcionamento da rede com uma imagem de dimensões 572x572.

Figura 3 – Arquitetura da U-net



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

Cada quadrado azul corresponde a um multi-canal de características. O número de canais é registrado acima da caixa. O tamanho x-y é definido na parte inferior esquerda da caixa. As caixas brancas representam as características copiadas e as setas denotam as diferentes operações (RONNEBERGER; FISCHER; BROX, 2015). Uma das vantagens da arquitetura é que esta é mais rápida de treinar devido ao seu aprendizado baseado em contexto (SIDDIQUE et al., 2021).

A arquitetura consiste de uma passagem de contração (esquerda) e uma de expansão (direita). A passagem de contração segue a arquitetura típica de uma rede de convolução; aplicando repetidamente duas convoluções 3x3, cada uma seguida por uma Unidade Linear Retificada (ReLU) e uma operação de 2x2 max pooling com a segunda etapa para redução de resolução. A cada redução, o número de canais de características é dobrado. Cada etapa da passagem expansiva consiste em operações de convolução transposta do mapa de características seguida por convolução 2x2, reduzindo pela metade o número de canais de recursos. Na camada final, é feita uma convolução 1x1 adicional para mapear cada vetor de características. No total, a rede possui 23 camadas de convolução. (RONNEBERGER; FISCHER; BROX, 2015).

De acordo com SIDDIQUE et al., o que faz com que a U-net seja utilizada na comunidade de Imagens Médicas é que esta possibilita a criação de mapas altamente detalhados com amostras limitadas.



## 1.3 Justificativa

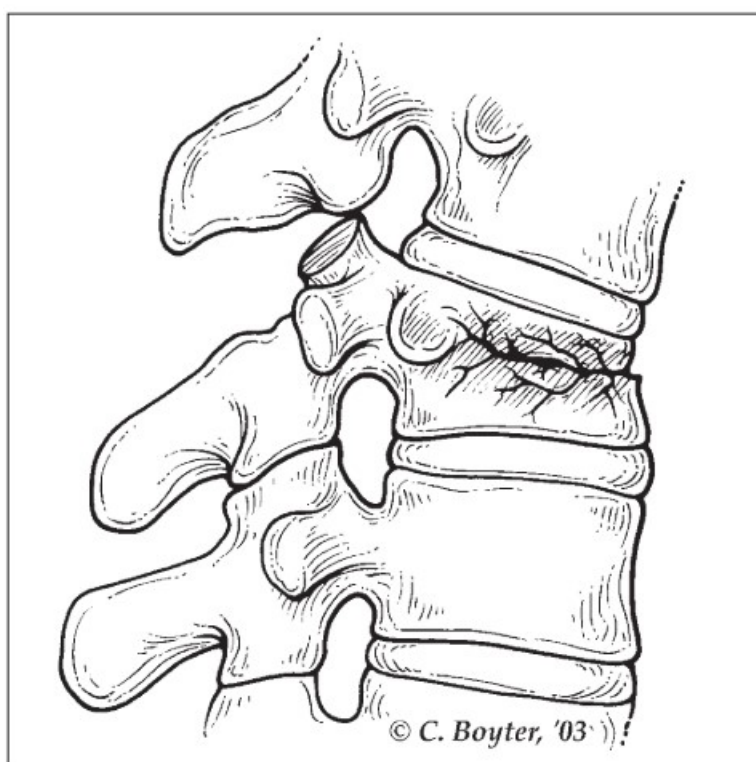
A segmentação de imagens médicas é uma etapa fundamental para diagnósticos, planejamento e intervenções de procedimentos cirúrgicos. No entanto, grande parte das técnicas utilizadas para executar essa etapa envolvem métodos manuais ou semi-automáticos, sendo dessa forma muito trabalhosos. Assim, surge o desafio de se implementar se métodos automáticos para realizar a segmentação de imagens médicas (BAE et al., 2020), com o auxílio desses métodos, promove-se a reprodutibilidade dos estudos feitos. Modelos de deep learning foram então criados para auxiliar nesse desafio.

## 2 FUNDAMENTAÇÃO TEÓRICA

### 2.1 Fraturas Vertebrais por Compressão

Fraturas vertebrais por compressão (fvc) podem ser definidas radiograficamente ou como um evento clínico. Como um evento clínico, são caracterizadas pela perda de peso e dor aguda; radiograficamente, fvc podem ser assintomáticas (SILVERMAN, 1992). Fvc ocorrem quando o corpo vertebral colapsa, o que pode fazer com que o paciente fique incapacitado de realizar atividades da vida diária (CORREIA, 2022). A Figura 4 apresenta uma ilustração de fratura vertebral por compressão.

Figura 4 – **Representação de imagem digital para a forma de matriz de níveis de cinza**



**Fonte:** (MCCARTHY; DAVIS, 2016)

Fraturas por compressão são tipicamente diagnosticadas por radiografias laterais da coluna vertebral. O critério radiográfico inclui redução de ao menos 20% da altura do corpo vertebral. Imagens obtidas por ressonância magnética ajudam a distinguir fraturas benignas das malignas e a determinar o tempo de cada fratura, dado que fraturas recentes apresentam edema (MCCARTHY; DAVIS, 2016).

O tratamento de fvc pode ser cirúrgico ou não, sendo este por repouso em um curto

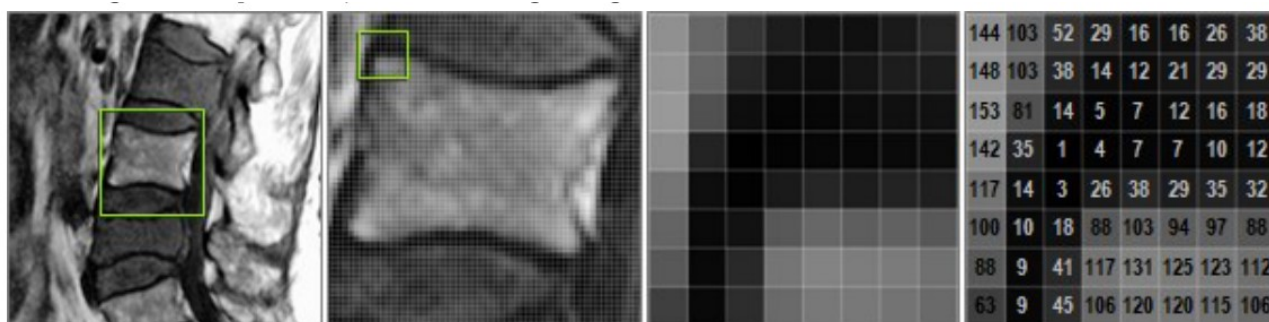
período, uso de analgésicos e fisioterapia (CORREIA, 2022). Os objetivos do tratamento são alívio da dor, restaurar a função e prevenção de fraturas futuras. O início do tratamento deve se iniciar com conversa com o paciente, discutindo riscos e benefícios de cada abordagem.

## 2.2 Reconhecimento de Padrões em Imagens Médicas

### 2.2.1 Imagem Médica Digital

A imagem médica digital é uma função  $f(x,y)$  em escala de cinza distribuída como coordenadas que pode ser representada por uma matriz, na qual o valor de cada pixel na matriz indica um nível de cinza no ponto  $(x,y)$  (SANTOS et al., 2019). A Figura 5 representa uma imagem digital na forma de matriz.

Figura 5 – **Representação de imagem digital para a forma de matriz de níveis de cinza**



Fonte: (CORREIA, 2022)

### 2.2.2 Etapas do reconhecimento de padrões em imagens

#### 2.2.2.1 Aquisição da Imagem

Para obter imagens de Ressonância Magnética, o paciente é colocado em um campo magnético e um pulso de ondas de rádio é gerado por antenas posicionadas no paciente. Assim, os prótons do paciente absorvem as ondas e reemitem essa energia após um período de tempo que depende das propriedades magnéticas do tecido. A leitura do sinal transmitido pelo paciente faz o mapeamento do sinal recebido para os níveis de cinza na imagem (BUSHBERG et al., 2003).

As variações no pulso de radiofrequência e na interpretação do sinal permitem a criação de imagens com diferentes tipos de contraste para um mesmo tecido, fazendo com que a ressonância magnética seja um dos exames mais sensíveis e detalhados (CORREIA, 2022).

### 2.2.2.2 Segmentação

É o processo de partição de uma imagem em seus objetos constituintes. Em geral, o reconhecimento é melhor conforme a acurácia da segmentação (GONZALES; WINTZ, 1987). Pode-se simplificar a segmentação utilizando formas definidas ou delineando na imagem uma região de interesse. Para imagens médicas, são utilizadas técnicas geralmente baseadas nos níveis de cinza, descontinuidade ou similaridade da imagem (SANTOS et al., 2019).

### 2.2.2.3 Extração de Atributos

As **características** são os valores numéricos que representam o exemplo dado ao modelo. No caso de imagens médicas, as características podem ser valores de pixels, variações dos valores de pixels em uma região, etc. (ERICKSON et al., 2017). Para obtê-los, algoritmos realizam o cálculo de valores numéricos que representam o conteúdo visual; são realizados procedimentos como a construção de histogramas, classificação de texturas, etc.. Por fim, os valores são armazenados em um vetor de atributos (SANTOS et al., 2019). De acordo com SANTOS et al. e SANTOS et al., os atributos são classificados nos seguintes grupos:

- **Cor (níveis de cinza):** os atributos podem ser obtidos diretamente ou por análise do histograma da imagem, que descreve a quantidade de níveis de cinza presentes na imagem;
- **Textura:** são atributos capazes de refletir detalhes dentro de uma lesão identificada na imagem, dando informações sobre a distribuição espacial do conteúdo;
- **Forma:** esses atributos descrevem a borda da imagem e características geométricas do objeto.

### 2.2.2.4 Extração de Atributos Relevantes

A extração de características consiste em fazer o algoritmo "ver" estruturas e traços nos dados recebidos, é melhor que o vetor de características seja pequeno, mas é crítico que este seja adequado para representar dados e compreender os objetivos (NAQA; MURPHY, 2022). Como muitos atributos podem ser extraídos das imagens, realiza-se uma seleção das características mais relevantes; para isso, foram criados algoritmos que têm por objetivo

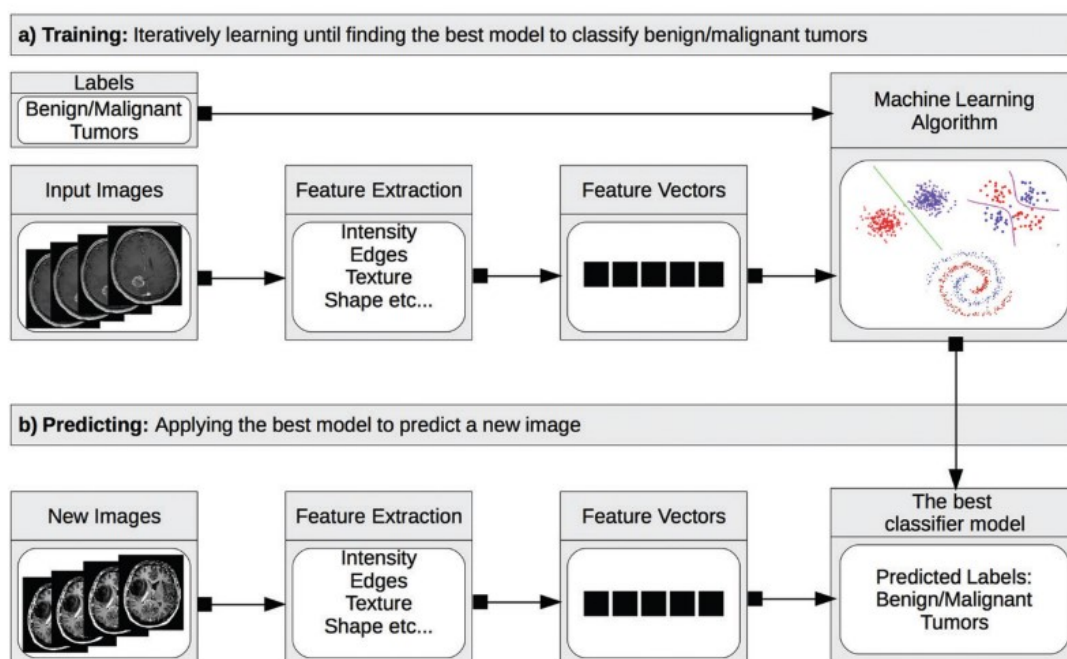
reduzir a dimensionalidade espacial do vetor. Boa parte desses algoritmos utilizam de aprendizado de máquina para executar essa tarefa (SANTOS et al., 2019).

### 2.2.2.5 Criação do Modelo

O modelo é conjunto de pesos ou pontos de decisão aprendido por um sistema de aprendizado de máquina. Uma vez aprendido, o modelo pode ser designado a um exemplo desconhecido para determinar a que classe tal exemplo pertence (ERICKSON et al., 2017). Para criar o modelo, são definidos os hiper-parâmetros **épocas** e **batch size**, que define o número de vezes que o algoritmo irá percorrer o dataset de treino durante a fase da aprendizagem e o que define o número de amostras utilizadas antes de atualizar o modelo interno dos parâmetros (BROWNLEE, 2018), respectivamente.

A Figura 6 a seguir ilustra o processo de desenvolvimento de um modelo treinado com machine learning no qual (a) apresenta a etapa de **treinamento**, em que são extraídas características da imagem para criação do vetor de características para o algoritmo identificar as propriedades dos exemplos (no caso da figura, diferenciar tumores benignos dos malignos). Em (b) é representada a etapa do **teste**, na qual o modelo resultante de (a) é aplicado em um novo conjunto de imagens para classificá-las e assim auxiliar o profissional a identificar o tumor.

Figura 6 – **Desenvolvimento de Modelo de Machine Learning**



Fonte: (ERICKSON et al., 2017)

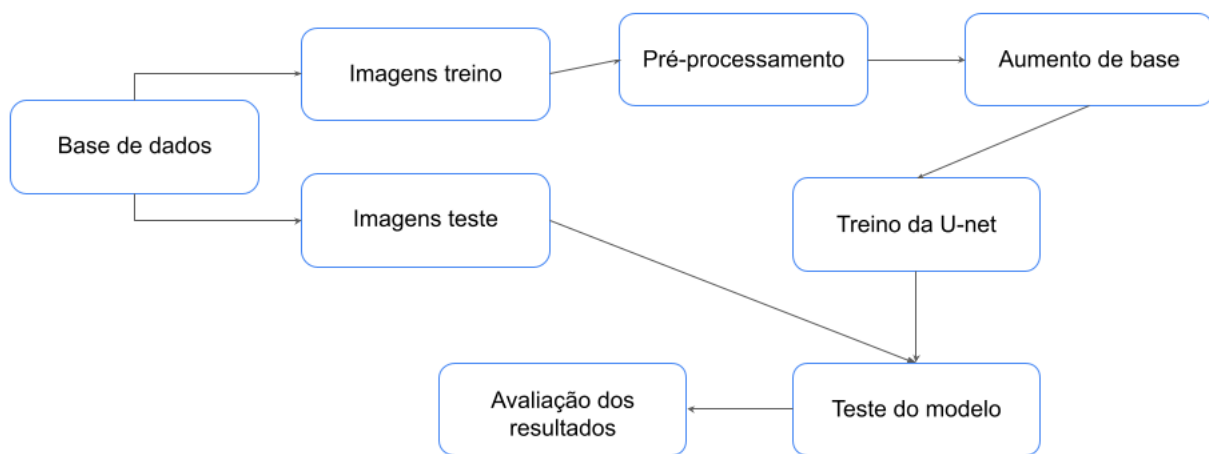
### 3 OBJETIVO

O presente Trabalho de Conclusão de Curso tem por objetivo estudar, aplicar, treinar e avaliar os resultados da aplicação de U-net para realizar a segmentação automatizada de corpos vertebrais fraturados de imagens provenientes de exames de ressonância magnética.

## 4 MATERIAIS E MÉTODOS

A figura 7 a seguir retrata o esquema de etapas seguidas para a execução do projeto. As imagens da base de dados foram redimensionadas, juntamente de suas máscaras. Como é necessário o mesmo número de máscaras e imagens, alguns exames foram duplicados para não prejudicar o treinamento do modelo.

Figura 7 – **Esquema das etapas da metodologia**



**Fonte:** Próprio autor

### 4.1 Base de Dados

A base de dados utilizada consiste de exames anonimizados e com uso aprovado pelo Comitê de Ética em Pesquisa (CEP) do Hospital das Clínicas da Faculdade de Medicina de Ribeirão Preto da Universidade de São Paulo - HCMFRP-USP, em concordância com o parecer de número 2.550.679 e o CAAE de número 84415318.8.0000.5440. Os exames provenientes de ressonância magnética compreendem sequências ponderadas em T1 de 91 pacientes ( 36 homens e 55 mulheres, com idade média de  $64.24 \pm 11.75$  anos) atendidos pelo HCMFRP-USP, sendo todos os pacientes diagnosticados com ao menos uma fratura

vertebral por compressão dentre os corpos vertebrais de L1 a L5 na sequência sagital (CORREIA, 2022).

## 4.2 Pré-processamento dos dados

Dado que as máscaras originais das vértebras fraturadas estavam em formato de arquivo .nrrd, as máscaras foram manualmente refeitas para o formato .png utilizando o programa Photoshop tendo por base as máscaras originais. Após a conversão, foi implementado algoritmo para redimensionar todas as imagens para dimensões de 256 x 256 pixels dado que as imagens originais apresentavam dimensões diferentes entre si.

Com o pré-processamento dos dados concluído, as imagens e máscaras foram separadas de modo que 80% do conteúdo fosse utilizado para treino e os outros 20% reservados para a etapa de teste. Dessa forma, 111 imagens passaram para a etapa de aumento de base e treinamento, enquanto as 19 restantes foram apenas redimensionadas e armazenadas para os testes futuros.

Para aprofundamento dos estudos, a base de dados original foi separada em 3 outras bases mantendo a proporção 80% para treino e 20% para teste. Estas 3 sub-bases foram feitas de modo que imagens diferentes fossem utilizadas para a validação dos modelos criados.

## 4.3 Aumento de Base

A acurácia de CNNs é melhorada conforme o tamanho do dataset, assim é aplicado o conceito de 'aumento de base', no qual o conjunto original é aumentado através de operações aleatórias de translação, rotação, giro e deformação (SEO et al., 2020).

Para realizar essa etapa, foi utilizada a biblioteca Albumentations (BUSLAEV et al., 2020), que possibilita a aplicação de diversas transformações às imagens. Neste trabalho, foram realizadas as seguintes transformações:

- Flip horizontal;
- Flip vertical;
- Distorção óptica;
- Transformação elástica;
- Distorção de grade;
- Center crop;
- Rotação de 90°;
- Transposição



Realizando todas essas transformações obteve-se 1110 imagens para auxiliar o treinamento do modelo.

O algoritmo de transformação elástica foi aplicado duas vezes ao aumento para melhorar o reconhecimento de corpos vertebrais com formato anormal.

## 4.4 Construção da Rede

A aplicação da U-net no presente trabalho foi feita com base no módulo 'Segmentação Semântica' do curso 'Segmentação de Imagens com Python de A a Z' de GRANATYR. Com uso do ambiente Jupyter, foi utilizada versão 2.10.0 do Tensorflow; nesse ambiente foi desenvolvido um algoritmo para aplicar o aumento de base e foi realizada a construção da rede com otimizador Adam.

## 4.5 Treino e Teste da Rede

Os modelos foram treinados todos com 50 épocas e batch size de 2. A cada rodada executada, foram salvos apenas os modelos que apresentavam melhor custo de função.

Foram criados diversos modelos para testá-los com diferentes datasets que mantivessem a proporção 80% para treino e 20% para teste; assim, modelos foram feitos executando certas combinações de técnicas de aumento de base. Foram treinados modelos usando apenas os flips vertical e horizontal, utilizando 5 transformações, e utilizando todas as transformações mencionadas anteriormente.

Após a criação dos modelos, os que apresentaram melhor desempenho foram utilizados nas outras sub-bases criadas a fim de obter generalização dos resultados.

## 4.6 Avaliação dos Resultados

Para a análise dos resultados, foram utilizadas duas das principais métricas de avaliação de performance em segmentação de imagens mencionadas por SIDDIQUE et al.:

- Coeficiente de Dice: compara a similaridade entre duas amostras.

$$Dice = \frac{2|GT \cap SR|}{|GT| + |SR|}$$

- Coeficiente de Jaccard (IoU): mede a sobreposição entre duas amostras.

$$IoU = \frac{|GT \cap SR|}{|GT \cup SR|}$$

Aqui, GT refere-se à máscara original (ground truth) e SR indica o resultado da segmentação. Um score elevado indica resultados com maior acurácia.

As métricas foram calculadas para cada modelo criado a fim de obter comparação entre os resultados obtidos.

## 5 RESULTADOS E DISCUSSÃO

A tabela a seguir demonstra os resultados da avaliação dos modelos. Para todos os testes foram efetuadas 50 épocas avaliando a melhora ou não da perda. Nota-se que todos os testes obtiveram média do IoU acima de 0,6.

Aqui, GT refere-se ao resultado de um especialista (ground truth) e SR indica o resultado da segmentação. Um score elevado indica resultados com maior acurácia.

Com a criação e aplicação de diversos modelos, foi possível obter os resultados apresentados na tabela 1

**Tabela 1 – Todos os modelos feitos e seus resultados**

modelo	dataset	épocas	descrição	IoU médio	Dice médio
1	1	50	2 transformações, 333 imagens	0.89993	0.99872
2	1	50	5 transformações, 666 imagens	0.71826	0.53
3	1	50	9 transformações, 1110 imagens	0.72228	0.548
4	2	50	2 transformações, 333 imagens	0.68618	0.49579
5	2	50	5 transformações, 666 imagens	0.71008	0.49016
6	2	50	9 transformações, 1110 imagens	0.71419	0.51532
7	3	50	2 transformações, 333 imagens	0.65139	0.42995
8	3	50	5 transformações, 666 imagens	0.66993	0.47089
9	3	50	9 transformações, 1110 imagens	0.69506	0.51389
10	1	50	sem aumento de base	0.66417	0.41432
11	2	50	sem aumento de base	0.830234	0.797
12	3	50	sem aumento de base	0.66467	0.41074

**Fonte:** Próprio autor

### 5.1 Generalização dos Resultados

Dos resultados exibidos na tabela 1, os cinco melhores modelos foram selecionados para serem testados em outras imagens a fim de analisar se este conseguiriam obter resultado similar aos seus primeiros testes. A tabela 2 apresenta o desempenho dos modelos 1, 2, 3, 6 e 11 nas 19 imagens do conjunto de teste das outras sub-bases criadas:

Tabela 2 – Resultado da aplicação dos modelos em outras imagens

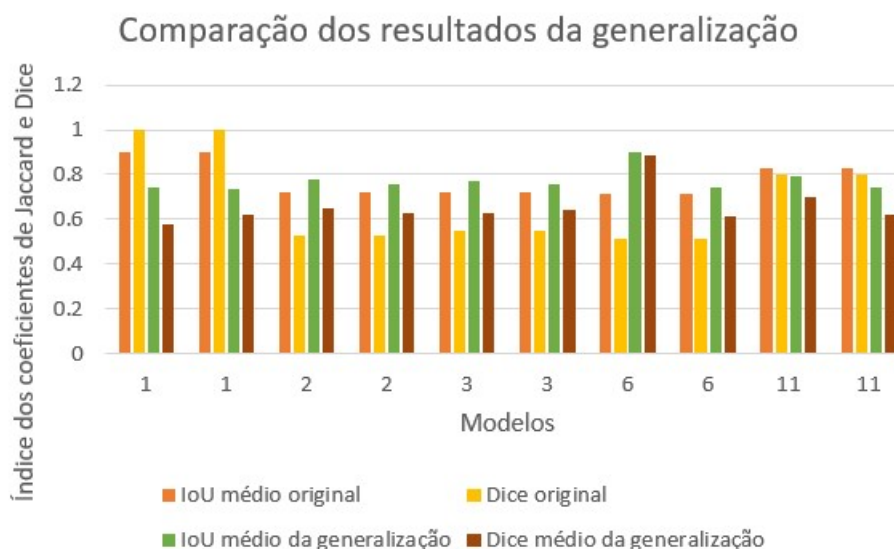
modelo	dataset	IoU médio	Dice médio	Comparação com o original
1	2	0.73912	0.57989	piora
1	3	0.73422	0.61889	piora
2	2	0.77632	0.65074	melhora
2	3	0.75372	0.62474	melhora
3	2	0.76931	0.63011	melhora
3	3	0.75943	0.64279	melhora
6	1	0.89993	0.88547	melhora
6	3	0.74546	0.61279	melhora
11	1	0.78935	0.698	piora
11	3	0.74083	0.62058	piora

Fonte: Próprio autor

Foi observado que os modelos cuja performance melhorou (aumento nos valores dos coeficiente de Dice e Jaccard) foram aqueles criados com o auxílio de aumento de base, enquanto os modelos 1 e 11 que passaram por dois e nenhum aumento respectivamente tiveram resultados piores que os originais; mostrando que os resultados que estes obtiveram podem ter sido consequências de viés nas imagens.

A Figura 8 a seguir exibe em forma de gráfico a diferença de performance após generalização dos resultados. Observa-se que, como anteriormente comentado, os modelos treinados com aumento de base (2, 4 e 6) apresentam crescimento nos coeficientes de Jaccard e Dice, enquanto houve redução nos resultados dos modelos restantes, treinados com poucos exemplos.

Figura 8 – Gráfico dos resultados originais e da generalização

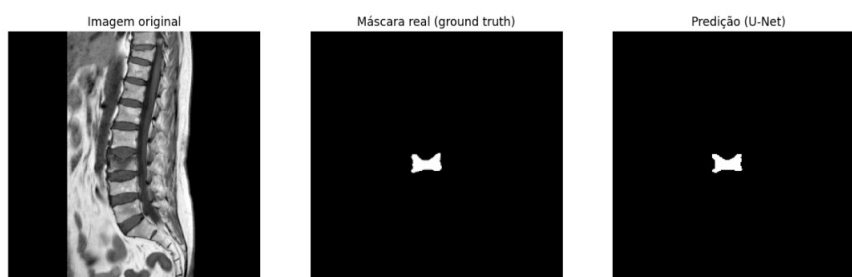


Fonte: Próprio autor

## 5.2 Eficiência do Aumento de Base

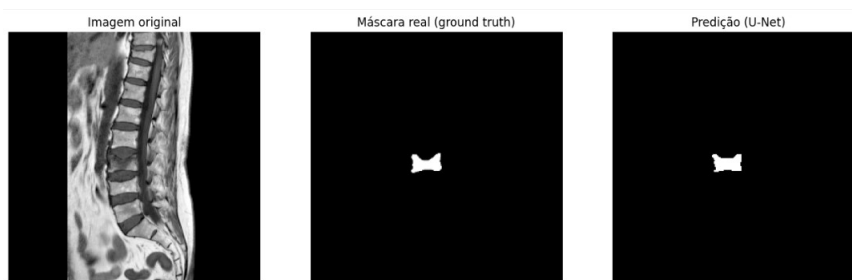
As imagens resultantes de modelos treinados com aumento de base apresentam mais detalhes quanto ao contorno dos corpos vertebrais. As figuras 9 e 10 a seguir retratam as diferenças entre as predições provenientes de modelos treinados com e sem operações de aumento de base.

Figura 9 – **Predição feita por modelo com aumento de base**



Fonte: Próprio autor

Figura 10 – **Predição feita por modelo sem aumento de base**



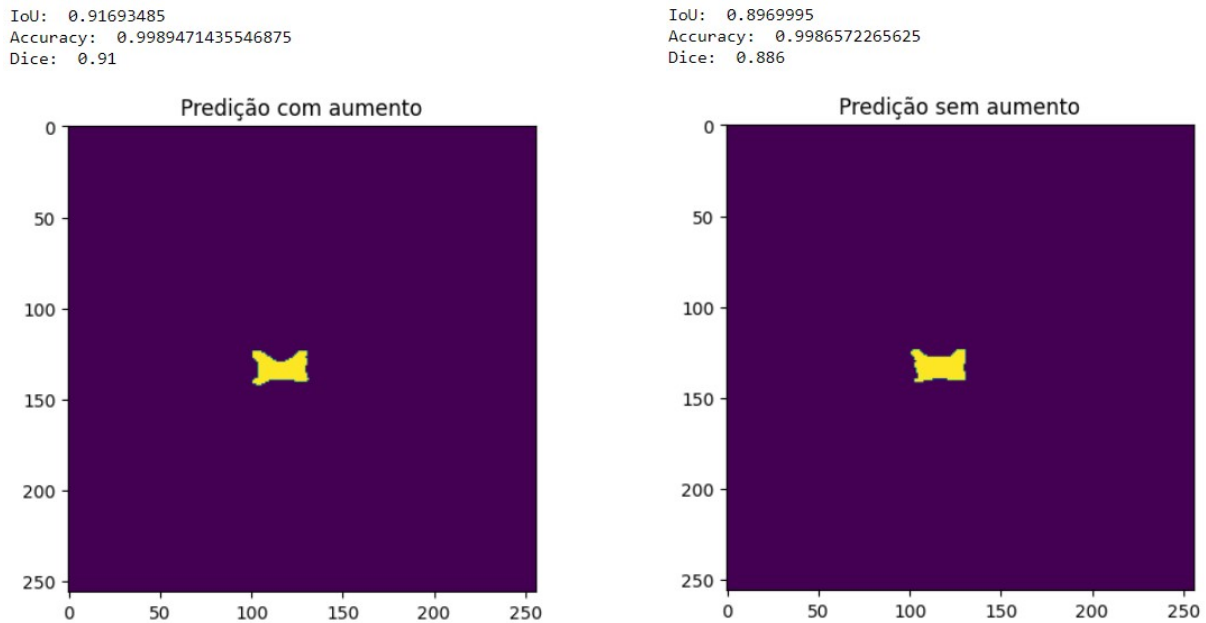
Fonte: Próprio autor

Percebe-se também que a média do IoU é menor quando comparada aos resultados de modelos treinados com aumento de base. A imagem 11 a seguir exibe os coeficientes de Jaccard e de Dice após aplicação de modelos com e sem aumento na mesma imagem.

## 5.3 Resultado das Predições

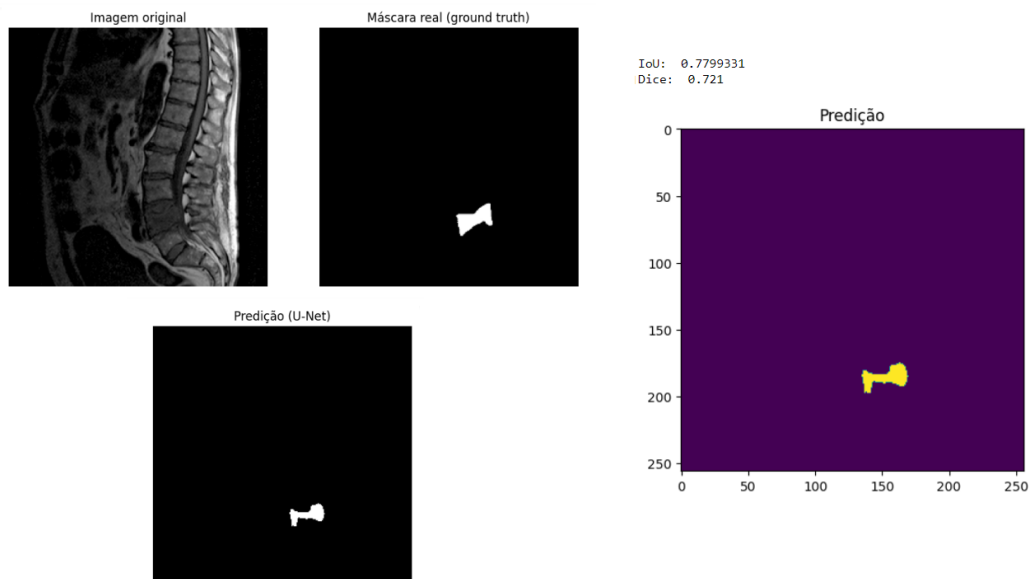
As figuras 12, 13 e 14 apresentam algumas predições e são acompanhadas de uma breve discussão sobre os resultados obtidos. As segmentações apresentadas foram obtidas a partir do modelo 6, uma vez que este foi treinado com o maior aumento de base, contendo nove operações; e este obteve o melhor resultado no teste de generalização. Cada caso está acompanhado de sua máscara verdadeira.

Figura 11 – Coeficientes de Jaccard e Dice após aplicação de modelo com e sem aumento de base



Fonte: Próprio autor

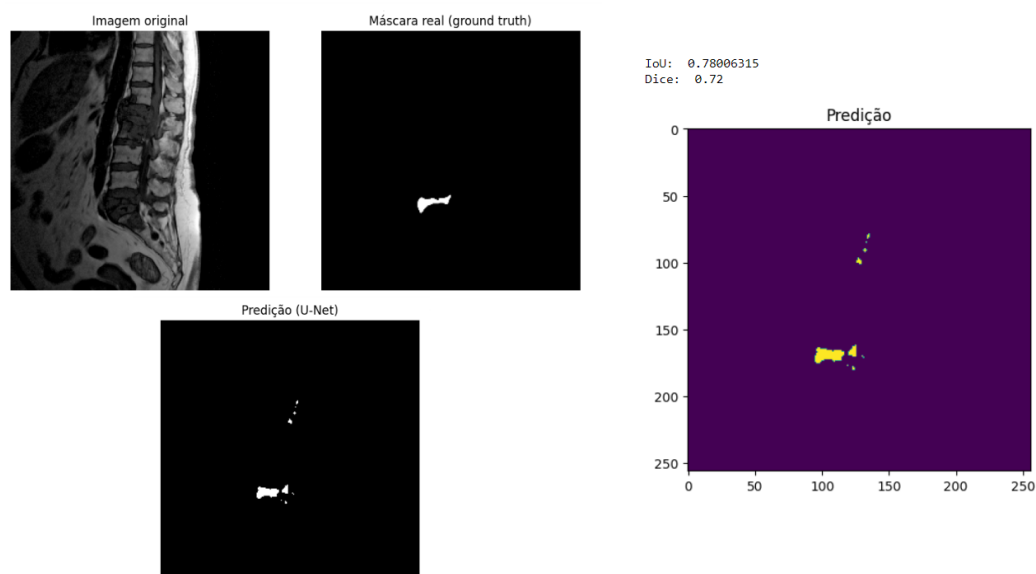
Figura 12 – Caso em que o corpo é detectado, porém segmentado incorretamente



Fonte: Próprio autor

A figura 12 demonstra que o modelo foi capaz de identificar que o corpo vertebral L4 estava fraturado, porém não conseguiu segmentá-lo corretamente. Isto pode ser devido ao padrão mais escuro da imagem original.

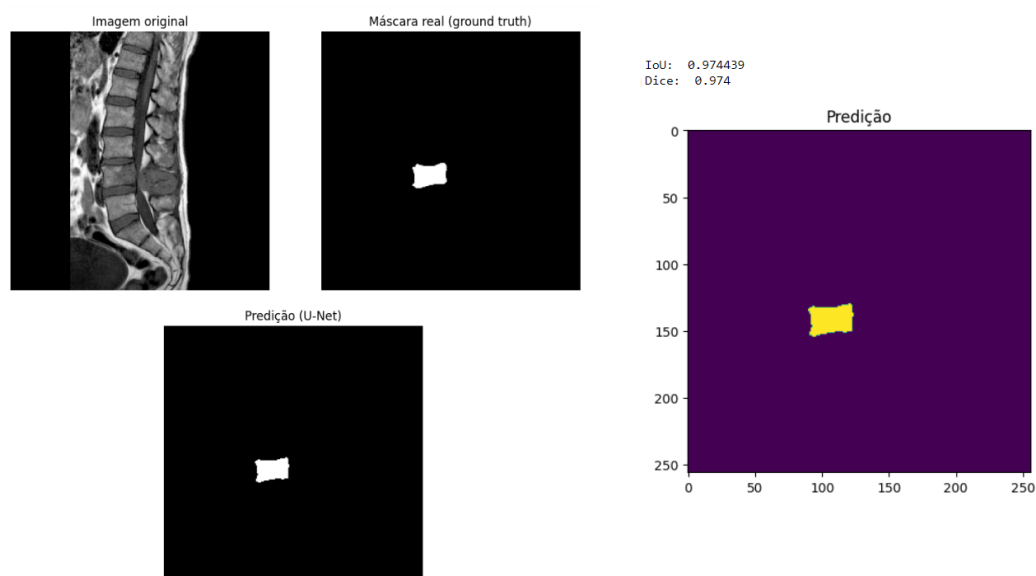
Figura 13 – Caso em que estruturas vizinhas também são segmentadas



Fonte: Próprio autor

Semelhante ao caso anterior, a figura 13 demonstra que o modelo foi capaz de identificar que o corpo vertebral L5 está fraturado, porém não conseguiu segmentá-lo corretamente, incluindo estruturas vizinhas na máscara gerada. Também neste caso é provável que o padrão da imagem original, na qual o corpo vertebral L2 possui tom de cinza similar ao corpo fraturado, tenha prejudicado o desempenho do modelo. Tal resultado pode prejudicar o trabalho do profissional, uma vez que haveria retrabalho para checar se os corpos segmentados incorretamente estão realmente fraturados ou não.

Figura 14 – Caso em que a predição melhor se aproxima da máscara verdadeira



Fonte: Próprio autor

A predição exibida na figura 14 apresentou o melhor resultado, tendo os melhores coeficientes. Esse resultado pode ser de maior interesse para o radiologista, já que exhibe as condições do corpo vertebral fraturado e tem maior utilidade como "segunda opinião".

Os resultados exibidos demonstram a variabilidade dos resultados e suas possíveis interpretações clínicas. No geral, os modelos foram capazes de executar predições satisfatórias para uma primeira implementação. Porém, fica evidente a necessidade de estudos futuros voltados para o melhorias no processo de treinamento.

O trabalho realizado por HWANG; KIM; JUNG foi realizado com metodologia similar, utilizando dados de ossos saudáveis e de ossos afetados por doenças hematológicas para treinar modelos de U-net, contando com aumento de base, batch size de 32 e 100 épocas para o treinamento. O modelo treinado apenas com dados de ossos adoecidos foi capaz de obter médias do coeficiente de Jaccard de 0.9128 e Dice de 0.9502, segmentando todos os corpos vertebrais presentes nas imagens. Nota-se que o modelo treinado a partir de dados de ossos tanto saudáveis como de ossos prejudicados teve performance um pouco melhor quanto ao que foi treinado apenas com ossos adoecidos.

Conseguindo segmentar os corpos vertebrais fraturados, mas não atingindo métricas de avaliação tão altas como o trabalho anteriormente mencionado; o modelo criado no presente trabalho foi capaz de realizar o objetivo de segmentar os corpos vertebrais fraturados.



## 6 CONCLUSÃO

O uso de técnicas de machine learning no ambiente clínico precisam gerar respostas precisas e os algoritmos também devem ser ágeis para que o profissional possa obter uma "segunda opinião" confiável. Dessa forma, pode-se concluir que a U-net possui alto potencial para a segmentação de corpos vertebrais, sejam esses fraturados ou não; havendo também diversos trabalhos na literatura focados na segmentação tridimensional de corpos vertebrais e outras estruturas do corpo.

Os modelos criados foram capazes de diferenciar corpos vertebrais fraturados dos saudáveis. Assim, atendendo ao objetivo inicial do trabalho de estudo de modelo de U-net para segmentação bidimensional. Com uma performance de até 0.89 para o coeficiente de Jaccard e 0.99 para o coeficiente de Dice considerando a etapa de treinamento e validação e 0,74 e 0,62 respectivamente para os coeficientes de Jaccard e Dice considerando a etapa de teste independente. Nota-se que modelos treinados de U-net são promissores para segmentação bidimensional de corpos vertebrais fraturados. Porém, deve-se ressaltar que tais resultados podem melhorar significativamente com uma base de dados maior, pois esta possibilitaria um aumento de base superior e, conseqüentemente, haveriam mais dados disponíveis para o treino de modelos.

Espera-se que o presente trabalho realizado possa servir como base para outros projetos a serem desenvolvidos e que futuramente o prognóstico e diagnóstico de fraturas vertebrais seja melhor e mais acessível aos pacientes.

# REFERÊNCIAS

- BAE, H.-J. et al. Fully automated 3d segmentation and separation of multiple cervical vertebrae in ct images using a 2d convolutional neural network. **Computer Methods and Programs in Biomedicine**, Elsevier, v. 184, p. 105119, 2020. Citado na página 17.
- BROWNLEE, J. What is the difference between a batch and an epoch in a neural network. **Machine Learning Mastery**, v. 20, 2018. Citado na página 21.
- BUSHBERG, J. et al. The essential physics of medical imaging. 2002. **Eur J Nucl Med Mol Imaging**, v. 30, p. 1713, 2003. Citado na página 19.
- BUSLAEV, A. et al. Alumentations: Fast and flexible image augmentations. **Information**, v. 11, n. 2, 2020. ISSN 2078-2489. Disponível em: <<https://www.mdpi.com/2078-2489/11/2/125>>. Citado na página 24.
- CHEN, H.-Y. et al. Application of deep learning algorithm to detect and visualize vertebral fractures on plain frontal radiographs. **Plos one**, Public Library of Science San Francisco, CA USA, v. 16, n. 1, p. e0245992, 2021. Citado na página 13.
- CORREIA, N. S. C. **Classificação computadorizada de imagens de ressonância magnética da coluna vertebral para diferenciação de fraturas vertebrais por compressão benignas e malignas utilizando uma abordagem 3D**. Tese (Doutorado) — Universidade de São Paulo, 2022. Citado 3 vezes nas páginas 18, 19 e 24.
- ENSRUD, K. E.; SCHOUSBOE, J. T. Vertebral fractures. **New England Journal of Medicine**, Mass Medical Soc, v. 364, n. 17, p. 1634–1642, 2011. Citado na página 13.
- ERICKSON, B. J. et al. Machine learning for medical imaging. **Radiographics**, Radiological Society of North America, v. 37, n. 2, p. 505–515, 2017. Citado 2 vezes nas páginas 20 e 21.
- GONZALES, R. C.; WINTZ, P. **Digital image processing**. [S.l.]: Addison-Wesley Longman Publishing Co., Inc., 1987. Citado na página 20.
- GRANATYR, J. **Segmentação de Imagens de A a Z**. 2022. <[https://www.udemy.com/share/105YrE3@YmpF7cPKDoFZQVn42UkgFmWzw1kzrHeZRNnZtB\\_mSK0z-qc151OUxGWOzSI9eln6lg==/](https://www.udemy.com/share/105YrE3@YmpF7cPKDoFZQVn42UkgFmWzw1kzrHeZRNnZtB_mSK0z-qc151OUxGWOzSI9eln6lg==/)>. Accessed: 2020–09-08. Citado na página 25.
- HWANG, E.-J.; KIM, S.; JUNG, J.-Y. Fully automated segmentation of lumbar bone marrow in sagittal, high-resolution t1-weighted magnetic resonance images using 2d u-net. **Computers in Biology and Medicine**, Elsevier, v. 140, p. 105105, 2022. Citado na página 32.
- KALMET, P. H. et al. Deep learning in fracture detection: a narrative review. **Acta orthopaedica**, Taylor & Francis, v. 91, n. 2, p. 215–220, 2020. Citado 2 vezes nas páginas 13 e 14.
- LI, Y.-C. et al. Can a deep-learning model for the automated detection of vertebral fractures approach the performance level of human subspecialists? **Clinical Orthopaedics and Related Research**, LWW, v. 479, n. 7, p. 1598–1612, 2021. Citado na página 13.
- MCCARTHY, J.; DAVIS, A. Diagnosis and management of vertebral compression fractures. **American family physician**, v. 94, n. 1, p. 44–50, 2016. Citado na página 18.

NAQA, I. E.; MURPHY, M. J. **Machine and Deep Learning in Oncology, Medical Physics and Radiology**. [S.l.]: Springer, 2022. Citado 4 vezes nas páginas 13, 14, 15 e 20.

RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: SPRINGER. **International Conference on Medical image computing and computer-assisted intervention**. [S.l.], 2015. p. 234–241. Citado na página 16.

SANTOS, M. K. et al. Inteligência artificial, aprendizado de máquina, diagnóstico auxiliado por computador e radiômica: avanços da imagem rumo à medicina de precisão. **Radiologia brasileira**, SciELO Brasil, v. 52, p. 387–396, 2019. Citado 3 vezes nas páginas 19, 20 e 21.

SEO, H. et al. Machine learning techniques for biomedical image segmentation: an overview of technical aspects and introduction to state-of-art applications. **Medical physics**, Wiley Online Library, v. 47, n. 5, p. e148–e167, 2020. Citado na página 24.

SIDDIQUE, N. et al. U-net and its variants for medical image segmentation: A review of theory and applications. **IEEE Access**, IEEE, 2021. Citado 3 vezes nas páginas 15, 16 e 25.

SILVERMAN, S. The clinical consequences of vertebral compression fracture. **Bone**, Elsevier, v. 13, p. S27–S31, 1992. Citado na página 18.



## Apêndices

# APÊNDICE A – CÓDIGO DESENVOLVIDO

A seguir, pseudocódigo da lógica seguida para a elaboração do código feito neste trabalho. Foram seguidas as aulas do curso "Segmentação de Imagens de A a Z" de Jones Granatyr na plataforma Udemy. O pseudocódigo foi dividido em blocos para facilitar a compreensão de cada etapa.

## A.1 Importação de Bibliotecas

---

```
1 import os, shutil
2 import numpy as np
3 import cv2
4 from tqdm import tqdm
5 from matplotlib import pyplot as plt
6 import tensorflow
7 import random
8 from glob2 import glob
```

---

## A.2 Carregar Base de Dados

---

```
1 def carregar_dataset(caminho do arquivo):
2     X_train = caminho das imagens de treino
3     y_train = caminho das mascaras de treino
4     x_test = caminho das imagens de teste
5     y_test = caminho das mascaras de teste
6
7 (X_train, y_train), (X_test, y_test) = carregar_dataset(caminho do arquivo)
```

---

## A.3 Pré-Processamento

---

```
1 largura = 256
2 altura = 265
```

---

```

3  canais = 3
4  dataset_imagens = imagens de X_train redimensionadas para largura e altura
5  mascaras = mascaras de y_test redimensionadas para largura, altura e canal
    de cor alterado para 1

```

---

## A.4 Aumento de base

---

```

1  import tensorflow as tf
2  import albumentations as A
3
4  def criar_diretorio(camimnho):
5      if diretorio nao existe:
6          criar diretorio novo
7
8  criar_diretorio('Diretorio para imagens e m scaras de treino com aumento de
    base')
9  criar_diretorio('Diretorio para armazenar imagens e m scaras de teste')
10
11 def aumento(imagens, mascaras, diretorio, altura, largura, aumento=True):
12     for imagens no diretorio informado:
13         if aumento = True:
14             aplicar transformacoes nas imagens
15         else:
16             nao aplicar transformacoes
17     for todas as imagens e mascaras:
18         redimesionar altura e largura para 256
19
20 aumento(X_train, y_train, 'Diretorio para imagens e mascaras de treino com
    aumento de base', aumento = True)
21 aumento(X_test, y_test, 'Diretorio para armazenar imagens e mascaras de
    teste', aumento = False)

```

---

## A.5 Construção da Rede

---

```

1  from tensorflow.keras.models import Model
2  from tensorflow.keras.layers import Input, Conv2D, MaxPooling2D,
    UpSampling2D, concatenate, Conv2DTranspose, BatchNormalization, Dropout,
    Lambda, ReLU

```

---

```

3 from tensorflow.keras.optimizers import Adam
4 from tensorflow.keras.layers import Activation, MaxPool2D, Concatenate
5 from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping,
    TensorBoard, ReduceLROnPlateau, CSVLogger
6 from tensorflow.keras.metrics import Recall, Precision4
7 from keras import backend as K
8
9 def bloco_cov(input, num_filtros):
10     criar bloco de convolucao
11
12 def bloco_encoder(input, num_filtros):
13     criar bloco bloco_encoder
14
15 def bloco_decoder(input, num_filtros):
16     criar bloco de bloco_decoder
17
18 def modelo_unet(formato):
19     montar modelo de u-net
20
21 def IoU(mascara_verdadeira, predicao, smooth = 1):
22     definir o calculo do IoU
23
24 def coef_Dice(mascara_verdadeira, predicao, smooth = 1):
25     definir o calculo do coeficiente de Dice
26
27 def perda_coef_dice(mascara_verdadeira, predicao):
28     definir a perda do coeficiente de Dice
29
30 epochs = 50
31 batch_size = 2
32 lr = 1e-4
33
34 modelo = modelo_unet((altura, largura, 3))
35 """compilar modelo com as metricas estabelecidas"""

```

---

## A.6 Treinamento

---

```

1 criar_diretorio(Criar diretorio para armazenar os modelos criados)
2 callbacks(implementar save_best_only para salvar apenas o melhor modelo)

```



```
3 history = treinar modelo
```

---

## A.7 Carregar Modelo

---

```
1 from tensorflow.keras.utils import CustomObjectScope
2 from tensorflow.keras.models import load_model
3 modelo_teste = carregar(carregar modelo salvo)
```

---

## A.8 Teste

---

```
1 def segmenta_imagem(imagem, modelo):
2     predicao = aplicar modelo_teste em imagem
3
4 carregar diretorio de imagens teste
5 for imagens em imagens_teste:
6     predicao = segmenta_imagem(imagem, modelo_teste)
```

---

## A.9 Avaliação

---

```
1 from sklearn.metrics import accuracy_score
2 from tensorflow.keras.metrics import MeanIoU
3
4 lista_scores = []
5
6 for imagens em teste_imagens:
7     segmenta_imagem(imagem, modelo_teste)
8
9     Iou = valor_iou(mascara, predicao)
10    dice = dice_coe(mascara, predicao)
11
12    lista_scores = recebe os valores de Iou e Dice para cada predicao
13
14 print(f"\nM dia do IoU: media do Iou")
15 print(f"\nM dia do coeficiente de Dice: media do coeficiente de Dice")
16
17 lista_scores #exibir dados de todas as predicoes
```

---