



Universidade Federal do ABC
Centro de Matemática, Computação e Cognição
Bacharelado em Ciência da Computação

Detecção e Classificação Automática de Osteoartrite de Joelho em Radiografias Utilizando Visão Computacional

Guilherme de Sousa Santos

Santo André - SP, 17 de dezembro de 2024

Guilherme de Sousa Santos

Deteccção e Classificação Automática de Osteoartrite de Joelho em Radiografias Utilizando Visão Computacional

Projeto de Graduação apresentado ao Programa de Graduação em Ciência da Computação (área de concentração: Visão Computacional), como parte dos requisitos necessários para a obtenção do título de Bacharel em Ciência da Computação.

Universidade Federal do ABC – UFABC

Centro de Matemática, Computação e Cognição

Bacharelado em Ciência da Computação

Orientador: Hugo Puertas de Araújo

Santo André - SP

17 de dezembro de 2024

Resumo

A osteoartrite de joelho (OA) é uma das condições articulares mais comuns e incapacitantes no mundo, sendo caracterizada como uma doença progressiva que afeta principalmente a cartilagem do joelho. Embora não tenha cura, a detecção precoce é fundamental para prevenir sua progressão. A radiografia é a principal técnica utilizada para o diagnóstico da OA e para sua classificação com base na escala de Kellgren/Lawrence (KL). No entanto, o diagnóstico radiológico depende da experiência, interpretação e tempo do profissional, o que pode gerar inconsistências ou erros. Nesse contexto, técnicas de aprendizado profundo oferecem uma alternativa mais rápida e eficiente, permitindo a automação da detecção e classificação da OA de joelho. Este estudo propõe uma comparação entre modelos de redes neurais convolucionais (RNCs) e vision transformers (ViTs) na tarefa de classificar a severidade da OA de joelho, abrangendo os modelos ResNet34, ResNet50, ResNet101, VGG16, VGG19, DenseNet121, DenseNet169, Inception, ViT-B/16, DeiT e Swin Transformer. A análise comparativa considera tanto métricas de performance, após o uso de *transfer learning*, quanto o consumo computacional envolvido no treinamento dos modelos. Após a realização dos experimentos, observou-se que as arquiteturas ResNet-50 e DenseNet-169 obtiveram os melhores desempenhos, com acurácias de 72,48% e 73,19% na classificação da OA de joelho em cinco classes, respectivamente.

Palavras-chaves: Classificação. osteoartrite de joelho. radiografias. redes neurais convolucionais. transfer-learning. vision transformers.

Abstract

Knee osteoarthritis (OA) is one of the most common and debilitating joint conditions worldwide, characterized as a progressive disease that primarily affects the knee cartilage. Although there is no cure, early detection is crucial to prevent its progression. Radiography is the main technique used to diagnose OA and classify it based on the Kellgren/Lawrence (KL) scale. However, radiological diagnosis depends on the professional's experience, interpretation, and time, which can lead to inconsistencies or errors. In this context, deep learning techniques offer a faster and more efficient alternative, enabling the automation of knee OA detection and classification. This study proposes a comparison between convolutional neural network (CNN) models and vision transformers (ViTs) for the task of classifying knee OA severity, including the models ResNet34, ResNet50, ResNet101, VGG16, VGG19, DenseNet121, DenseNet169, Inception, ViT-B/16, DeiT, Swin Transformer, and ResNet50-ViT-B/16. The comparative analysis considers both performance metrics, following the application of transfer learning, and the computational resources required to train the models. It is expected that the dense networks (DenseNet121 and DenseNet169), along with the hybrid architecture ResNet50-ViT-B/16, will get the best results.

Keywords: Classification. convolutional neural networks. knee osteoarthritis. radiographs. transfer-learning. vision transformers.

Lista de ilustrações

Figura 1 –	Imagens de recuperação por inversão sagital (A–C) e eco de spin rápido coronal (D–F) ilustrando os achados da ressonância magnética na osteoartrite. (A) Sinovite reativa (seta branca espessa), (B) Formação de cistos subcondrais (seta branca), (C) Edema da medula óssea (setas brancas finas), (D) Desgaste parcial da cartilagem (seta preta espessa), (E–F) Desgaste total da cartilagem (setas pretas finas), esclerose subcondral (cabeça de seta) e formação de osteófitos marginais (seta dupla). Imagem cortesia dos Drs. Hollis Potter e Catherine Hayter, Hospital for Special Surgery, Nova York, NY. Fonte: Loeser et al. (2012).	7
Figura 2 –	Uma rede neural convolucional simples, composta por apenas cinco camadas. Fonte: Saxena (2022).	11
Figura 3 –	Rede neural artificial com uma camada oculta. Fonte: James et al. (2000).	12
Figura 4 –	Arquitetura de uma rede perceptron multicamadas com duas entradas, uma camada oculta de quatro neurônios e três neurônios de saída (os neurônios de viés são mostrados, mas normalmente estão implícitos). Fonte: Géron (2019).	14
Figura 5 –	Propagação para frente. Fonte: Lillicrap et al. (2020).	16
Figura 6 –	Retropropagação. Fonte: Lillicrap et al. (2020).	17
Figura 7 –	Definição de visão computacional. Fonte: Team (2024).	18
Figura 8 –	Número de imagens em cada classe do conjunto de dados	24
Figura 9 –	Aprendizado residual	28
Figura 10 –	Um bloco de 5 camadas de uma DenseNet. Cada camada recebe como entrada a saída de todas as camadas anteriores.	29
Figura 11 –	Um módulo Inception.	30
Figura 12 –	Um módulo Inception com fatoração de convoluções.	31
Figura 13 –	Metodologia para os vision transformers	33
Figura 14 –	Matriz de confusão do modelo ResNet-50.	42

Lista de tabelas

Tabela 1 – Escala de Kellgren/Lawrence para classificação da severidade de osteoartrite.	9
Tabela 2 – Configuração dos modelos VGG-16 e VGG-19. Os parâmetros de cada camada convolucional são denotados por "conv<tamanho do campo receptivo>-<número de canais>". A função de ativação ReLU não é exibida por motivos de simplicidade.	27
Tabela 3 – Configuração das arquiteturas ResNet-34, ResNet-50 e ResNet-101. . .	29
Tabela 4 – Configuração das arquiteturas DenseNet-121 e DenseNet-169.	30
Tabela 5 – Configuração da arquitetura Inception-v3.	31
Tabela 6 – Desempenho dos modelos utilizando a abordagem Cross Entropy . . .	40
Tabela 7 – Desempenho dos modelos de RNCs e ViTs na classificação da OA de joelho usando a função da perda CORN.	43
Tabela 8 – Desempenho dos cinco melhores modelos (ordem por QWK) para as duas funções de perda.	44

Lista de abreviaturas e siglas

OA	Osteoartrite
KL	Kellgren/Lawrence
IA	Inteligência Artificial
RNC	Rede Neural Convolucional
ViT	Vision Transformer
WHO	World Health Organization
OAI	Osteoarthritis Initiative
NIH	National Institutes of Health
CAM	Class Activation Mapping
GAP	Global Average Pooling

Sumário

1	INTRODUÇÃO	1
	Introdução	1
1.1	Objetivos	2
1.1.1	Objetivo Geral	2
1.1.2	Objetivos Específicos	2
1.2	Organização do Trabalho	3
2	FUNDAMENTAÇÃO TEÓRICA	5
2.1	Osteoartrite de Joelho	5
2.1.1	Definição e Características Clínicas	5
2.1.2	Mudanças Patológicas da OA de Joelhos	6
2.1.3	Impacto da OA na Qualidade de Vida	6
2.1.4	Prevalência da OA	8
2.1.5	Diagnóstico e Métodos de Avaliação da OA	9
2.1.6	Classificação da OA de Joelhos	9
2.2	Rede Neural Convolucional (RNC)	10
2.2.0.1	Perceptron e Perceptron Multicamadas	11
2.2.0.1.1	Perceptron	11
2.2.0.1.2	Treinamento do Perceptron	12
2.2.0.1.3	Perceptron Multicamadas	13
2.2.0.2	Funções de Ativação	13
2.2.0.3	Algoritmo de Retropropagação	16
2.2.0.4	Otimização de Redes Neurais	17
2.3	Visão Computacional	18
2.4	Visão Computacional na Saúde	20
2.5	Aprendizado Profundo	21
3	METODOLOGIA	23
3.1	Coleta de dados	23
3.2	Pré-processamento das imagens	24
3.2.1	Normalização	24
3.2.2	Equalização de Histograma	24
3.2.3	Aumento de dados	25
3.3	Arquitetura do modelo de Rede Neural Convolucional	25
3.3.1	VGG (Visual Geometry Group Network)	26

3.3.2	ResNet (Residual Network)	27
3.3.3	DenseNet (Densely Connected Convolutional Networks)	28
3.3.4	Inception-v3	30
3.4	Arquitetura do modelo de Vision Transformer	32
3.4.1	ViT-B/16	32
3.4.2	DeiT (Data-efficient Image Transformer)	33
3.4.3	Swin Transformer (Shifted Window Transformer)	33
3.5	Métricas de avaliação	33
3.5.1	Acurácia	34
3.5.2	Precisão	34
3.5.3	Recall	34
3.5.4	F1-Score	34
3.5.5	Matriz de Confusão	35
3.5.6	AUC-ROC	35
3.6	Eficiência computacional	35
3.7	Método de visualização	36
4	RESULTADOS	39
4.0.1	Classificação em Cinco Classes	39
4.1	Resultados e Discussão	43
4.1.1	Visão Geral dos Resultados	43
4.1.2	Discussão	43
4.1.3	Principais Conclusões	44
	Referências	45

1 Introdução

A osteoartrite (OA), popularmente conhecida como artrose, é uma forma muito comum de doença reumática, caracterizada como uma condição multifatorial e degenerativa que afeta desde a cartilagem articular até os ossos adjacentes, resultando em sintomas de dor, deformidade e perda de função [Kraus et al. \(2015\)](#), [PACCA et al. \(2018\)](#). Esses impactos comprometem significativamente a qualidade de vida, especialmente em grupos mais afetados, como idosos, mulheres e indivíduos obesos [PACCA et al. \(2018\)](#). Além de sua alta prevalência, a OA é uma das principais causas de incapacidade no mundo, com maior incidência na articulação do joelho, seguido de quadril e da mão. Dados de 2020 apontam que a doença afeta cerca de 7,6% da população global, e projeções indicam um aumento de 60 a 100% até 2050 [Courties et al. \(2024\)](#).

Exercícios de propriocepção e fortalecimento muscular, assim como terapias farmacêuticas, têm sido aplicadas a pacientes diagnosticados com OA de joelho com o objetivo de controlar ou reduzir os sintomas de dor, uma vez que não existem medicamentos capazes de retardar o seu desenvolvimento [Sardim et al. \(2020\)](#), [Lin et al. \(2009\)](#). Essa abordagem é especialmente apropriada para pacientes em estágios iniciais da doença, quando a cartilagem ainda não foi completamente degradada [Kanamoto et al. \(2020\)](#). No entanto, o diagnóstico depende da experiência e cuidado médico na interpretação das radiografias, o que pode levar a inconsistências entre o grau previsto e o grau real, devido às mínimas diferenças entre os estágios adjacentes da doença [KELLGREN and LAWRENCE \(1957\)](#), [Mohammed et al. \(2023\)](#). Esses desafios têm impulsionado estudos sobre sistemas automáticos de detecção e classificação da OA de joelho.

A introdução de técnicas de inteligência artificial (IA) nos últimos anos tem permitido a automação de tarefas que antes eram realizadas manualmente, incluindo a interpretação de imagens médicas [Wang et al. \(2024\)](#). Alguns exemplos incluem a detecção de pneumonia [Tilve et al. \(2020\)](#), a identificação e classificação de câncer de pulmão em tomografias computadorizadas [Tekade and Rajeswari \(2018\)](#) e a detecção de retinopatia diabética em imagens de fundo de olho [Dai et al. \(2021\)](#). No campo da reumatologia, a visão computacional também tem sido aplicada para a detecção de OA de joelho a partir de radiografias, com o objetivo de automatizar o processo de diagnóstico e reduzir a subjetividade da interpretação humana, assim como na tarefa de classificação da severidade da doença através da escala de Kellgren/Lawrence [Mohammed et al. \(2023\)](#).

Esses estudos têm se concentrado em utilizar arquiteturas de aprendizado profundo, como Redes Neurais Convolucionais (RNCs), e compará-las entre si para identificar qual abordagem oferece melhor desempenho na classificação da severidade da OA. No entanto, a

operação de convolução limita o relacionamento entre pixels distantes numa imagem, o que pode prejudicar a capacidade de captar dependências de longo alcance em radiografias, por exemplo [Shamshad et al. \(2023\)](#). Como uma abordagem alternativa, ou até complementar, foram propostas arquiteturas baseadas em Transformers, capazes de performar muito bem em tarefas de classificação, como é o caso do Vision Transformer (ViT) [Dosovitskiy et al. \(2021\)](#). Essas arquiteturas têm sido aplicadas com sucesso em tarefas relacionadas à medicina, como o diagnóstico de COVID-19 a partir de radiografias, classificação de tumores e doenças de retina, tornando-se o estado da arte nesta área [Shamshad et al. \(2023\)](#).

Nesse sentido, este trabalho se propõe a fazer uma comparação entre o desempenho de RNCs e modelos de ViTs na tarefa de detecção e classificação da OA de joelho seguindo a escala de Kellgren/Lawrence a partir de radiografias. A comparação será feita com base em métricas de performance, como acurácia, precisão, recall e F1-score, além de analisar a eficiência computacional, incluindo tempo de treinamento e quantidade de computação usada. O objetivo é identificar qual abordagem é mais adequada para uso como uma ferramenta auxiliar em diagnósticos clínicos. Para isso, serão utilizadas técnicas de pré-processamento de imagens, seleção dos melhores hiperparâmetros e estratégias de treinamento, bem como a avaliação dos modelos de classificação propostos.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

O objetivo geral deste trabalho consiste em comparar o desempenho de modelos baseados em Redes Neurais Convolucionais (RNCs) e Vision Transformers (ViTs) para detectar e classificar a osteoartrite de joelho usando radiografias, facilitando o diagnóstico da doença por meio de uma ferramenta automatizada.

1.1.2 Objetivos Específicos

- Realizar uma revisão bibliográfica sobre a OA de joelho e as técnicas de visão computacional aplicadas à detecção de doenças reumáticas;
- Treinar os modelos propostos para classificar a severidade da OA de joelho;
- Comparar os modelos de RNCs e ViTs com base em métricas de performance e eficiência computacional;
- Otimizar os modelos mais promissores e avaliar o impacto das mudanças nos hiperparâmetros na performance dos modelos;

- Analisar os resultados obtidos e discutir as vantagens e desvantagens de cada abordagem.

A metodologia proposta para atingir os objetivos deste trabalho consiste nas seguintes etapas: coleta e pré-processamento de um conjunto de dados de radiografias de joelhos com diferentes graus de severidade da OA seguindo a escala de Kellgren/Lawrence; implementação da *pipeline* de treinamento dos modelos de RNCs e ViTs para classificar a severidade da OA de joelho mantendo a mesma arquitetura e hiperparâmetros; avaliação dos modelos com base em métricas de performance e eficiência computacional; otimização dos melhores modelos e avaliação do impacto das mudanças nos hiperparâmetros na performance dos mesmos; análise dos resultados obtidos e discussão das vantagens e desvantagens de cada abordagem.

1.2 ORGANIZAÇÃO DO TRABALHO

Este trabalho está organizado em seis capítulos, incluindo a introdução. No Capítulo 2, são apresentados os conceitos e definições necessárias para o entendimento deste trabalho, incluindo a osteoartrite de joelhos e suas características clínicas, a visão computacional na área da saúde e os conceitos fundamentais de arquiteturas de aprendizado profundo, incluindo as RNCs e os ViTs. No Capítulo 3, são abordados os trabalhos relacionados. No Capítulo 4, é apresentada a metodologia proposta para atingir os objetivos deste trabalho, assim como a avaliação dos modelos. No Capítulo 5, são apresentados os resultados obtidos e discussões sobre os mesmos. Por fim, no Capítulo 6, são apresentadas as conclusões finais deste trabalho, apontando as contribuições, limitações e sugestões para trabalhos futuros.

2 Fundamentação Teórica

Neste capítulo, são apresentados os conceitos e as definições necessárias para o entendimento deste trabalho. A Seção 2.1 apresenta a osteoartrite de joelhos e suas características clínicas. A seção 2.3 aborda a visão computacional na área da saúde. A Seção 2.5 mostra alguns conceitos fundamentais de arquiteturas de aprendizado profundo, incluindo as redes neurais convolucionais e os vision transformers.

2.1 OSTEOARTRITE DE JOELHO

2.1.1 Definição e Características Clínicas

A osteoartrite (OA) é definida como uma doença heterogênea e degenerativa, que afeta as articulações e estruturas ósseas de pacientes, causando sintomas de dor, deformidade e perda de função (Loeser et al., 2012). Considerando os fenótipos da doença, ou seja, as características clínicas e radiográficas observáveis, a OA é considerada altamente heterogênea, isso significa que pode ser causada por diversos fatores, incluindo:

- **Idade:** a OA é mais comum em idosos, devido ao desgaste natural e inevitável das articulações ao longo do tempo (Anderson and Loeser, 2010).
- **Sexo:** mulheres têm maior risco de desenvolver OA do que homens, especialmente após a menopausa, devido à diminuição dos níveis de estrogênio, que protege a cartilagem articular (Tschon et al., 2021).
- **Obesidade:** o excesso de peso também é uma condição de risco para a OA, pois aumenta a carga mecânica nas articulações, influenciando o início e a progressão da doença (PACCA et al., 2018).
- **Predisposição genética:** fatores genéticos também podem influenciar o desenvolvimento da OA, como a presença de mutações em genes relacionados à formação e manutenção da cartilagem articular (Spector and MacGregor, 2004).
- **Outros fatores:** lesões articulares, atividade física intensa, doenças metabólicas, entre outros.

A OA pode afetar diversas articulações, como joelhos, quadris, mãos, ombros, entre outras. No entanto, a junção do joelho é a área mais afetada devido ao suporte do peso corporal que está diretamente associados a movimentos essenciais, como caminhar, subir escadas e agachar (Kanamoto et al., 2020). Portanto, tais fatores fazem com que a

doença seja uma das principais causas de dor crônica e incapacidade funcional, levando a uma necessidade de identificar e classificar a OA de forma precisa e precoce, para que o tratamento seja iniciado o mais cedo possível a fim de retardar a progressão da doença e melhorar a qualidade de vida dos pacientes.

2.1.2 Mudanças Patológicas da OA de Joelhos

Entre as mudanças patológicas observadas na OA, estão:

- **Degradação da cartilagem articular:** a cartilagem articular é um tecido que reveste as extremidades ósseas, permitindo movimentos suaves e absorção de impactos. Na OA, ocorre uma perda progressiva da matriz cartilaginosa, onde as células da cartilagem, chamadas de condrócitos, se tornam "ativas" e aumentam a produção de enzimas que degradam a matriz ([Goldring and Marcu, 2009](#)).
- **Inflamação sinovial:** a membrana sinovial é um tecido que reveste as articulações e produz o líquido sinovial, que lubrifica e nutre a cartilagem. Na OA, ocorre a condição chamada sinovite, onde a membrana sinovial se torna inflamada, causando dano e destruição à cartilagem ([Pessler et al., 2008](#)).
- **Degeneração dos ligamentos:** os ligamentos são estruturas que conectam os ossos e estabilizam as articulações. Na OA, os ligamentos podem sofrer rupturas e degeneração, afetando a mecânica articular. Essa degeneração aumenta a predisposição para o desenvolvimento da doença ([Loeser et al., 2012](#)).
- **Degeneração do menisco:** o menisco, estrutura fibrocartilaginosa que na absorção de choques e na estabilidade articular, também é afetado na OA. Sua degeneração leva à perda da função de amortecimento e à piora da sobrecarga nas superfícies articulares ([Loeser et al., 2012](#)).
- **Alterações ósseas:** o osso subcondral, localizado abaixo da cartilagem, também é afetado na OA, como a formação de osteófitos, que são projeções ósseas anormais, e a esclerose subcondral, que é o aumento da densidade óssea. Essas alterações podem causar dor e limitação de movimentos ([van der Kraan and van den Berg, 2007](#)).

A Figura 1 ilustra as mudanças patológicas observadas na OA de joelhos a partir de imagens de ressonância magnética.

2.1.3 Impacto da OA na Qualidade de Vida

De acordo com o World Health Organization (WHO), "qualidade de vida" é definida como a percepção do indivíduo sobre sua posição de vida no contexto da cultura e sistema

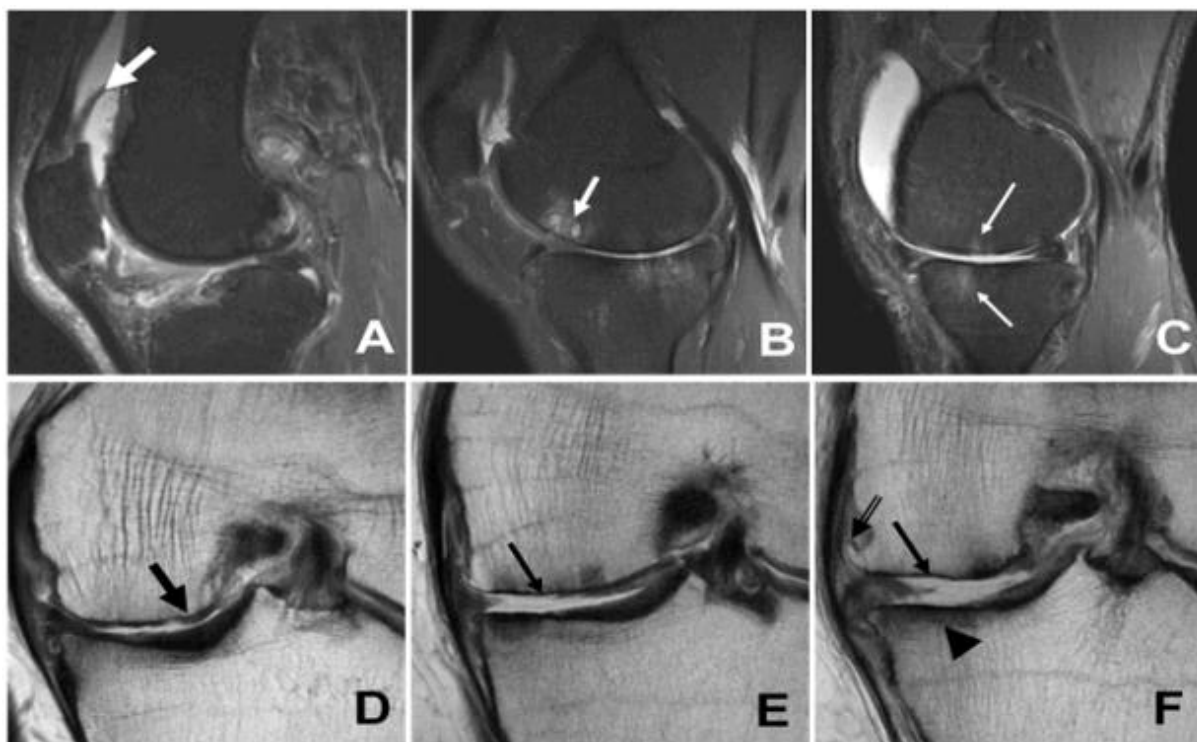


Figura 1 – Imagens de recuperação por inversão sagital (A–C) e eco de spin rápido coronal (D–F) ilustrando os achados da ressonância magnética na osteoartrite. (A) Sinovite reativa (seta branca espessa), (B) Formação de cistos subcondrais (seta branca), (C) Edema da medula óssea (setas brancas finas), (D) Desgaste parcial da cartilagem (seta preta espessa), (E–F) Desgaste total da cartilagem (setas pretas finas), esclerose subcondral (cabeça de seta) e formação de osteófitos marginais (seta dupla). Imagem cortesia dos Drs. Hollis Potter e Catherine Hayter, Hospital for Special Surgery, Nova York, NY. Fonte: [Loeser et al. \(2012\)](#).

de valores que ele vive e em relação aos seus objetivos, expectativas, padrões e preocupações ([Organization, 2012](#)).

Existe um grande esforço de pesquisadores e especialistas para avaliar o grau de incapacidade física causado pela doença, além de avaliar os efeitos de diferentes tratamentos em aspectos como dor, função física e mobilidade. No entanto, tais manifestações físicas afetam diretamente outras áreas na vida dos pacientes, como interações sociais, saúde mental e qualidade do sono ([Ferrel, 1992](#)). Além disso, comparado com outras doenças crônicas, pacientes com doenças musculoesqueléticas, como a OA, são os mais afetados em termos de qualidade de vida. A OA de joelho, especificamente, tende a declinar progressivamente a qualidade de vida conforme a progressão da doença ([Hoogeboom et al., 2013](#)).

[Desmeules et al. \(2009\)](#) realizaram um estudo com 197 pacientes com cirurgia agendada para substituição total do joelho (TKA) e avaliaram, através da escala de qualidade de vida SF-36 ([Ware and Sherbourne, 1992](#)), a relação entre a OA de joelho e

a qualidade de vida. Os resultados mostraram que a pontuação média da qualidade de vida dos pacientes era significativamente menor do que a população geral no Canadá ($p < 0,05$). Outros estudos também mostraram resultados similares em pacientes esperando por TKA (Snider et al., 2005; Kapetanakis, 2011). É razoável, portanto, que pacientes com OA de joelho severa tenham baixos níveis de qualidade de vida comparado com a população geral.

Sutbeyaz et al. (2007) fizeram um estudo com 28 pacientes obesos com OA de joelho e avaliaram a qualidade de vida através da escala de qualidade de vida SF-36. Os resultados mostraram que os pacientes obesos tiveram pontuações muito mais baixas em todos os domínios da escala SF-36, em comparação com o grupo de controle ($p < 0,001$). Além disso, a obesidade foi associada a uma pior qualidade de vida em pacientes com OA de joelho, o que sugere que a perda de peso pode ser benéfica para melhorar a qualidade de vida desses pacientes.

Complementarmente, Kawano et al. (2015) mostraram que existe uma relação do nível de escolaridade com a capacidade funcional e dor em pacientes com OA de joelho. O estudo foi conduzido com 93 pacientes tratados no Serviço de Ortopedia e Traumatologia do Hospital Santa Izabel e Santa Casa da Misericórdia da Bahia, em Salvador, Brasil. A avaliação da qualidade de vida foi feita através do questionário SF-36 e mostrou que pacientes com níveis mais baixos de escolaridade tiveram pontuações mais baixas nos domínios de capacidade funcional ($p < 0,001$), limitação funcional ($p = 0,009$) e dor ($p = 0,01$), em comparação com pacientes com níveis mais altos de escolaridade ($p < 0,05$). Além disso, a escolaridade foi associada a uma melhor qualidade de vida em pacientes com OA de joelho, o que sugere que a educação pode ser um fator importante para melhorar a qualidade de vida desses pacientes.

2.1.4 Prevalência da OA

Dados recentes do Global Burden of Disease (GBD) - o estudo epidemiológico observacional mais abrangente do mundo - revelaram que a prevalência da OA cresceu 132% entre 1990 e 2020, com projeções de crescimento de 60 a 100% até 2050, alcançando a marca de 1 bilhão de pessoas. Com uma prevalência de 7,6% da população global em 2020, o que equivale a aproximadamente 595 milhões de pessoas, a OA é mais comum em países desenvolvidos, devido à correlação com o status socioeconômico, e contribui significativamente para os chamados "anos vividos com incapacidade" (YLDs em inglês). Além disso, o estudo também aponta que a OA é mais comum em mulheres do que em homens, com prevalência de 8,0% e 5,8%, respectivamente, além de atingir principalmente idosos, especialmente aqueles acima de 70 anos, onde a OA assume a 7ª posição entre as principais causas de incapacidade, primeiramente afetando a articulação do joelho (Courties et al., 2024).

No Brasil, [Érika Rodrigues Senna et al. \(2004\)](#) realizaram um estudo com mais 3 mil pessoas e identificaram cerca de 7,2% com doenças reumáticas, sendo a OA a mais comum, com prevalência de 4,14%. Essa prevalência tende a aumentar visto que, além de existir uma correlação entre a OA e a obesidade, estima-se que o Brasil tenha uma taxa de sobrepeso e obesidade combinados de 68,1% em 2030 ([Brasília, 2024](#)).

2.1.5 Diagnóstico e Métodos de Avaliação da OA

O diagnóstico da OA normalmente é feito com base em exames clínicos, como a avaliação dos sintomas do paciente, exames de imagem, como radiografias e ressonâncias magnéticas, e exames laboratoriais, como a análise do líquido sinovial ([Kraus et al., 2015](#)). Exames de raio-x tem sido o método mais comum para diagnosticar a OA, pois é uma abordagem acessível e permite visualizar o espaço articular e alterações ósseas e cartilaginosas nas articulações, como a formação de osteófitos.

Essa avaliação é tipicamente feita por radiologistas a partir de radiografias do joelho estendido ou flexionado, dependendo da necessidade de visualização intra-articular ([Braun and Gold, 2012](#)). A partir dessas imagens, é possível fazer a classificação da severidade da OA e, em caso de diagnóstico, recomendar tratamentos farmacêuticos e não farmacêuticos, como exercícios de fortalecimento muscular e fisioterapia.

2.1.6 Classificação da OA de Joelhos

[KELLGREN and LAWRENCE \(1957\)](#) propuseram uma escala de classificação da OA baseada em radiografias e considerando fatores como a formação de osteófitos, estreitamento da cartilagem articular e esclerose subcondral. A escala de Kellgren/Lawrence (KL) classifica a OA em cinco estágios de progressão: 0 (nenhum), 1 (duvidoso), 2 (mínimo), 3 (moderado) e 4 (grave) ([Tabela 1](#)). Como a classificação é comumente feita por radiologistas, estes avaliam as radiografias e atribuem um grau de acordo com a experiência e cuidado médico na interpretação das imagens.

No entanto, a classificação manual pode ser subjetiva e suscetível a erros, assim como foi observado pelos autores, o que pode levar a diagnósticos tardios ou incorretos num cenário onde a detecção precoce é crucial para retardar a progressão da doença, uma vez que não existem medicamentos capazes de retardar o seu desenvolvimento .

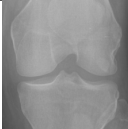


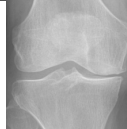
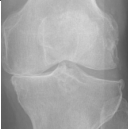
				
0 (saúdável)	1 (duvidoso)	2 (mínimo)	3 (moderado)	4 (severo)

Tabela 1 – Escala de Kellgren/Lawrence para classificação da severidade de osteoartrite.

2.2 REDE NEURAL CONVOLUCIONAL (RNC)

Uma rede neural artificial é um modelo computacional inspirado no cérebro humano (McCulloch and Pitts, 1943), onde neurônios artificiais recebem um conjunto de entradas ponderadas, realizam uma soma dessas entradas e aplicam uma função de ativação para produzir uma saída. Essa estrutura permite que as redes neurais aprendam padrões complexos a partir de dados, tornando-as adequadas para tarefas de processamento de linguagem natural, visão computacional, entre outras aplicações.

Em 2006, Hinton et al. (2006) propuseram o uso de redes neurais artificiais com múltiplas camadas com o objetivo de melhorar a capacidade dos modelos, o que levou a um renascimento do interesse nessas redes e ao desenvolvimento de novas arquiteturas, como é o caso da rede neural convolucional (CNN, do inglês *Convolutional Neural Network*).

As CNNs são modelos de aprendizado profundo projetados para processar dados com estrutura de grade, como imagens. Inspiradas na organização do córtex visual, CNNs são amplamente utilizadas em tarefas de visão computacional, como classificação de imagens, detecção de objetos e segmentação semântica.

A camada de convolução é o componente central das CNNs, responsável por extrair características locais dos dados de entrada. Essa camada utiliza filtros (ou *kernels*), que são pequenas matrizes de pesos (por exemplo, 3x3 ou 5x5) aplicadas em toda a imagem de entrada para gerar um mapa de características (ou *feature maps*), representando a presença dessas características em diferentes regiões da imagem.

Esses filtros são ajustados durante o treinamento da rede, permitindo que a CNN aprenda a detectar padrões relevantes, como bordas, texturas e formas. Conforme a rede avança pelas camadas, os filtros se tornam mais complexos e capazes de capturar características de alto nível, como objetos inteiros. Após as convoluções, é comum utilizar a função de ativação ReLU (*Rectified Linear Unit*), que substitui valores negativos por zero e introduz não linearidades no modelo, permitindo que ele aprenda representações complexas.

Após as camadas de convolução, as CNNs geralmente incluem camadas de *pooling* para reduzir a dimensionalidade dos *feature maps*, enquanto preservam as características mais relevantes. O *pooling* pode ser feito de várias maneiras, como *max pooling* (onde o valor máximo de uma região é mantido) ou *average pooling* (onde a média dos valores é calculada). Esse processo contribui para:

- reduzir a quantidade de parâmetros e o custo computacional da rede.
- tornar a rede mais robusta a pequenas variações nos dados de entrada.

Após diversas camadas de convolução e *pooling*, uma ou mais camadas totalmente

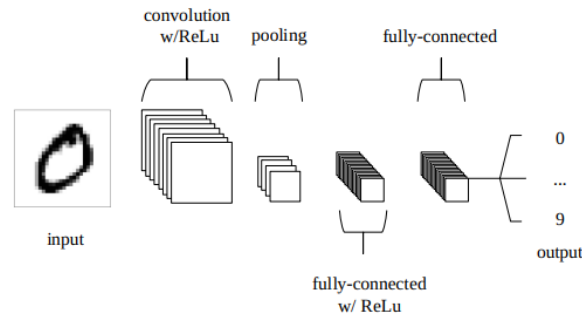


Figura 2 – Uma rede neural convolucional simples, composta por apenas cinco camadas. Fonte: [Saxena \(2022\)](#).

conectadas (*fully connected*) são adicionadas ao final da rede para combinar as características extraídas de camadas anteriores e realizar a tarefa de classificação. Cada neurônio dessas camadas está conectado a todos os valores da camada anterior, permitindo decisões baseadas em combinações globais das informações aprendidas. Em tarefas de classificação, a última camada totalmente conectada geralmente utiliza a função de ativação *softmax*, que transforma as saídas em probabilidades.

Durante o treinamento, a CNN ajusta os pesos dos filtros por meio do algoritmo de retropropagação (*backpropagation*), em que o erro de saída é retropropagado pela rede para atualizar os pesos e minimizar a função de perda. Esse processo é repetido por várias épocas, permitindo que a rede aprenda a reconhecer padrões complexos nos dados de entrada.

A [Figura 2](#) ilustra uma rede neural convolucional composta por cinco camadas. O número de camadas, a disposição dessas camadas, o número e tamanho dos filtros, a forma de conexão entre as camadas, entre outros fatores, podem variar dependendo da arquitetura escolhida. Em seguida, serão apresentadas algumas das arquiteturas populares de CNN que serão utilizadas neste trabalho.

2.2.0.1 Perceptron e Perceptron Multicamadas

2.2.0.1.1 Perceptron

O tipo mais simples de rede neural é chamado de perceptron, que consiste num modelo linear baseado em um neurônio artificial capaz de fazer previsões binárias. Ele é muito similar à regressão linear, mas com a adição de uma função de ativação que transforma a saída em uma previsão binária ([James et al., 2000](#)). O perceptron pode ser definido pela seguinte fórmula:

$$\hat{Y} = \begin{cases} 1 & \text{se } \sum_{i=1}^n w_i x_i > \theta \\ 0 & \text{caso contrário} \end{cases} \quad (2.1)$$

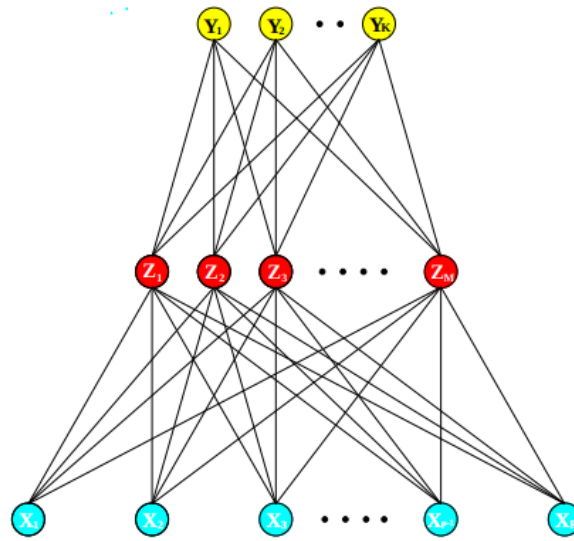


Figura 3 – Rede neural artificial com uma camada oculta. Fonte: [James et al. \(2000\)](#).

onde w_i são os pesos das variáveis preditoras x_i , θ é o limiar de ativação e \hat{Y} é a previsão do modelo. Este algoritmo, também conhecido como perceptron de camada única, serve como base para as redes neurais artificiais e é usado para resolver problemas de classificação binária.

A [Figura 3](#) ilustra uma rede neural típica com uma camada de entrada, uma camada oculta e uma camada de saída. Este tipo de rede é capaz de modelar tanto problemas de classificação quanto de regressão, dependendo da função de ativação usada e o número de neurônios na camada de saída. Para um problema de classificação com K classes, existem K neurônios na camada de saída, cada um representando uma classe possível ([James et al., 2000](#)).

2.2.0.1.2 Treinamento do Perceptron

O treinamento do perceptron é feito através do algoritmo de aprendizado supervisionado, onde o modelo é ajustado iterativamente para minimizar a função de perda. O algoritmo mais comum para treinar o perceptron é o algoritmo de Rosenblatt ([Rosenblatt, 1958](#)), que busca encontrar um hiperplano que separe duas classes através da minimização da distância de pontos mal classificados em relação ao hiperplano.

A primeira etapa consiste em inicializar os pesos, normalmente com valores aleatórios baixos ou zero. É comum utilizar w_i para representar o i -ésimo valor no vetor de pesos e $x_{j,i}$ para representar o valor da i -ésima variável preditora no j -ésimo valor no vetor de entrada de treinamento. Em seguida, é calculada a saída prevista y_j para cada valor de entrada x_j da seguinte forma:

$$\begin{aligned}
y_j(t) &= f[w(t) \cdot x_j] \\
&= f[w_0(t) \cdot x_{j,0} + w_1(t) \cdot x_{j,1} + \dots + w_n(t) \cdot x_{j,n}]
\end{aligned} \tag{2.2}$$

onde f é a função de ativação e $w(t)$ é o vetor de pesos no tempo t . A saída prevista $y_j(t)$ é então comparada com o valor real d_j para atualizar os pesos de acordo com a regra de aprendizado:

$$w_i(t+1) = w_i(t) + \eta \cdot (d_j - y_j(t)) \cdot x_{j,i} \tag{2.3}$$

onde η é a taxa de aprendizado, que controla o tamanho do passo de atualização dos pesos. O algoritmo continua iterando sobre os dados de treinamento até que a função de perda $\frac{1}{s} \sum_{j=1}^s |d_j - y_j(t)|$ seja minimizada, onde s é o número de observações no conjunto de treinamento, ou até que um critério de parada, como a convergência do erro, a acurácia do modelo ou o número máximo de iterações seja atingido.

2.2.0.1.3 Perceptron Multicamadas

Ao adicionar múltiplas camadas de neurônios entre a entrada e a saída, forma-se o que se denomina perceptron multicamadas (MLP – *Multilayer Perceptron*), geralmente composto por uma camada de entrada, uma ou mais camadas ocultas — responsáveis pelo processo de aprendizado — e uma camada de saída (Sarker, 2021). A Figura 4 apresenta uma arquitetura típica de um MLP.

Quando a rede neural é expandida com várias camadas ocultas, ela passa a ser chamada de rede neural profunda (*Deep Neural Network*), capaz de aprender representações hierárquicas, mais complexas e abstratas dos dados (Géron, 2019). Esse tipo de arquitetura permite a resolução de problemas altamente não lineares e desafiadores, frequentemente presentes em tarefas como visão computacional, processamento de linguagem natural e reconhecimento de voz.

2.2.0.2 Funções de Ativação

As funções de ativação desempenham um papel fundamental nas redes neurais artificiais, pois são responsáveis por introduzir não linearidade no modelo e melhorar a convergência da rede (Dubey et al., 2022). Sem elas, independentemente do número de camadas ocultas adicionadas, a rede se comportaria como um modelo linear, incapaz de aprender e representar padrões complexos e não lineares dos dados.

Matematicamente, uma função de ativação f transforma a saída z de um neurônio em uma nova saída $f(z)$, que será propagada para as camadas seguintes da rede. Algumas das funções de ativação mais comuns incluem:

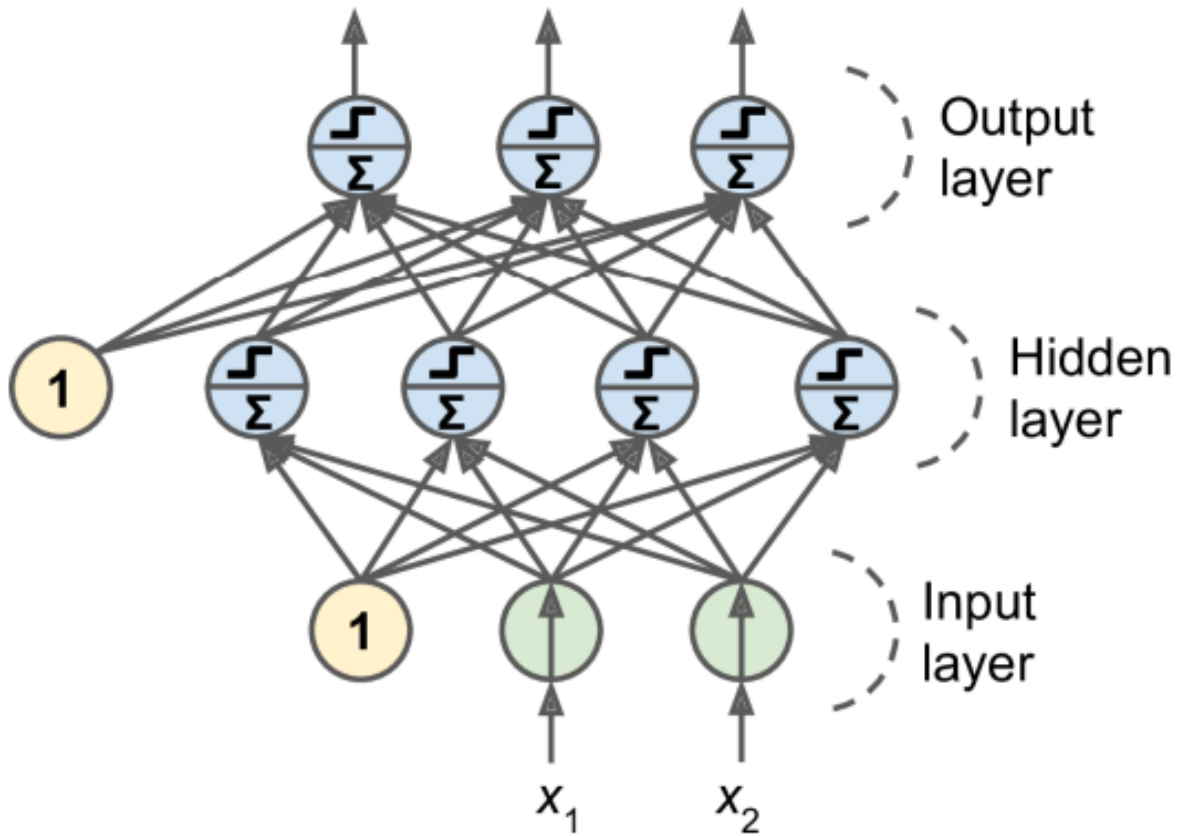


Figura 4 – Arquitetura de uma rede perceptron multicamadas com duas entradas, uma camada oculta de quatro neurônios e três neurônios de saída (os neurônios de viés são mostrados, mas normalmente estão implícitos). Fonte: Géron (2019).

- **Sigmoide:** é uma função limitada e diferenciável, que tem exatamente um ponto de inflexão. Devido a sua característica de limitar valores em um intervalo definido, ela é chamada de função de compressão (ou *squashing function* do inglês) (Lederer, 2021). A função sigmoide logística, também conhecida como função logística, é definida como:

$$f(z) = \sigma(z) = \frac{1}{1 + e^{-z}}$$

Essa função mapeia os valores de entrada para o intervalo $(0, 1)$, o que a torna útil para modelar probabilidades. Por isso, a função sigmoide é frequentemente utilizada em problemas de classificação binária, onde a saída é interpretada como uma probabilidade (Nwankpa et al., 2018).

Apesar de ser historicamente importante na teoria e prática de redes neurais, a função sigmoide sofre com o problema do *desvanecimento do gradiente* (*vanishing gradient*), onde os gradientes se tornam muito pequenos durante o treinamento, dificultando a atualização dos pesos e levando a um aprendizado lento ou estagnado, especialmente em redes profundas (Dubey et al., 2022).

- **Tangente Hiperbólica (tanh):** é uma função de ativação semelhante à sigmoide, mas com a vantagem de ser centrada em zero, o que pode contribuir para a convergência do modelo, além de ter uma saída mais ampla, variando entre -1 e 1 (Nwankpa et al., 2018). A função tangente hiperbólica é definida como:

$$f(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

A função tanh também sofre com o problema do desvanecimento do gradiente, mas em menor grau do que a sigmoide (Dubey et al., 2022). Outra condição importante a ser considerada é que a função tanh pode produzir “neurônios mortos” quando a entrada é muito grande ou muito pequena, resultando em gradientes nulos e impedindo a atualização dos pesos (Nwankpa et al., 2018).

- **ReLU (Rectified Linear Unit):** atualmente uma das funções mais utilizadas em aplicações de aprendizado profundo, a ReLU é uma função de ativação mais rápida e eficiente do que a sigmoide e a tangente hiperbólica (Nair and Hinton, 2010). De forma simples, a ReLU é uma função linear que retorna o valor de entrada se ele for maior que zero, e zero caso contrário, definida como:

$$f(z) = \max(0, z) = \begin{cases} z, & \text{se } z > 0 \\ 0, & \text{caso contrário} \end{cases}$$

Ela resolve parcialmente o problema do desvanecimento do gradiente, mas pode levar à inatividade de neurônios quando $z < 0$ (Dubey et al., 2022). Apesar disso, a ReLU é amplamente utilizada em redes neurais profundas devido à sua simplicidade e eficiência computacional.

- **Leaky ReLU:** é uma variação da ReLU que introduz uma pequena inclinação para valores negativos, evitando que os neurônios fiquem permanentemente inativos durante o processo de treinamento (Maas et al., 2013). O parâmetro α controla a inclinação da função para valores negativos, evitando que os gradientes sejam zero (Nwankpa et al., 2018). A Leaky ReLU é definida como:

$$f(z) = \alpha z + z = \begin{cases} z, & \text{se } z > 0 \\ \alpha z, & \text{caso contrário} \end{cases}$$

onde α é um pequeno valor positivo, tipicamente 0,01.

- **Softmax:** é uma função de ativação usada para computar a distribuição de probabilidade de um vetor de valores reais, onde a soma das saídas é igual a 1 (Nwankpa et al., 2018). A função softmax é definida como:

$$f(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}, \quad \text{para } i = 1, 2, \dots, K$$

onde K é o número total de classes.

A função softmax é frequentemente utilizada na camada de saída de redes neurais para problemas de classificação multiclasse, onde cada neurônio na camada de saída representa uma classe e a saída da função softmax é interpretada como a probabilidade de cada classe (Nwankpa et al., 2018).

2.2.0.3 Algoritmo de Retropropagação

Quando se trata de aprendizado supervisionado em redes neurais, o algoritmo de retropropagação (ou *backpropagation* em inglês) é um dos métodos mais utilizados para treinar modelos de redes neurais artificiais, especialmente em redes multicamadas (Sarker, 2021). Este algoritmo funciona ajustando os pesos dos neurônios de maneira a minimizar a função de perda através do cálculo do gradiente do erro da rede em relação aos pesos (Géron, 2019).

O algoritmo consiste em duas etapas principais: a fase de propagação para frente e a fase de retropropagação. Na fase de propagação para frente, os dados de entrada são transmitidos através da rede, camada por camada, até chegar à camada de saída. Durante essa fase, cada neurônio calcula sua saída com base nas entradas recebidas e nos pesos associados, aplicando a função de ativação correspondente (Sigmoid, ReLU, etc.). Por exemplo, para um neurônio k , sua saída h_k é dada por $h_k = f(a_k)$, onde $a_k = \sum_j h_j W_{jk}$, f é a função de ativação e W_{jk} é o peso associado à conexão entre o neurônio j e o neurônio k (Figura 5). A saída final da rede é então comparada com o valor real usando uma função de perda (MSE, entropia cruzada, etc.), que quantifica o erro da previsão. Por exemplo, para um neurônio l da camada de saída, o erro é calculado usando a saída prevista y_l e o valor real t_l (Lillicrap et al., 2020).

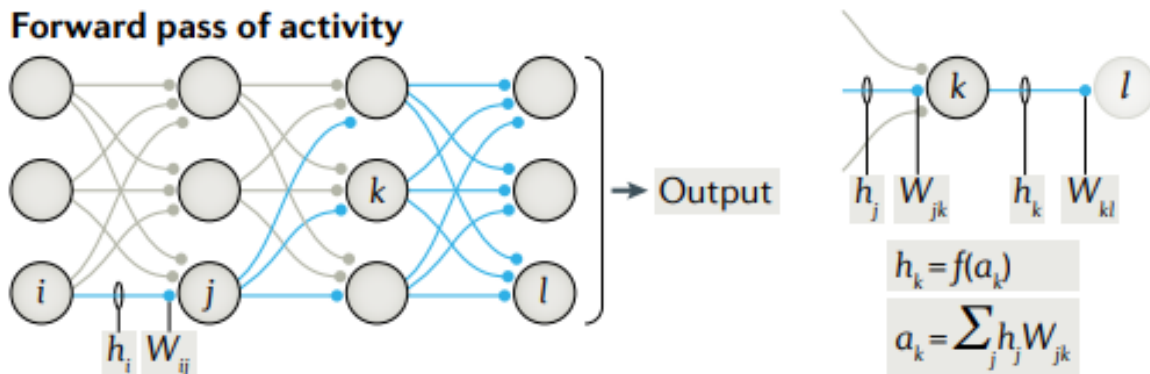


Figura 5 – Propagação para frente. Fonte: Lillicrap et al. (2020).

Na fase de retropropagação (), o erro calculado na camada de saída é propagado de volta através da rede, ajustando os pesos de cada neurônio com base na contribuição para o erro. Isso é feito usando a regra da cadeia, que permite calcular o gradiente do erro em relação aos pesos (Géron, 2019). A forma mais simples de usar o gradiente é mudar cada peso W_{ij} na direção do gradiente negativo, ou seja:

$$\Delta W_{ij} = -\eta \frac{\partial E}{\partial W_{ij}} = -\eta h_i \delta_j \quad (2.4)$$

onde η é a taxa de aprendizado, E é a função de perda, e $\delta_j = e_j f'(a_j) = (\sum_k \delta_k W_{jk}) f'(a_j)$ é o erro do neurônio j na camada oculta (Lillicrap et al., 2020). A retropropagação continua até que todos os pesos da rede sejam atualizados, minimizando assim a função de perda. Esse processo é repetido para várias iterações ou épocas, até que o modelo converja para um conjunto de pesos que minimizam o erro.

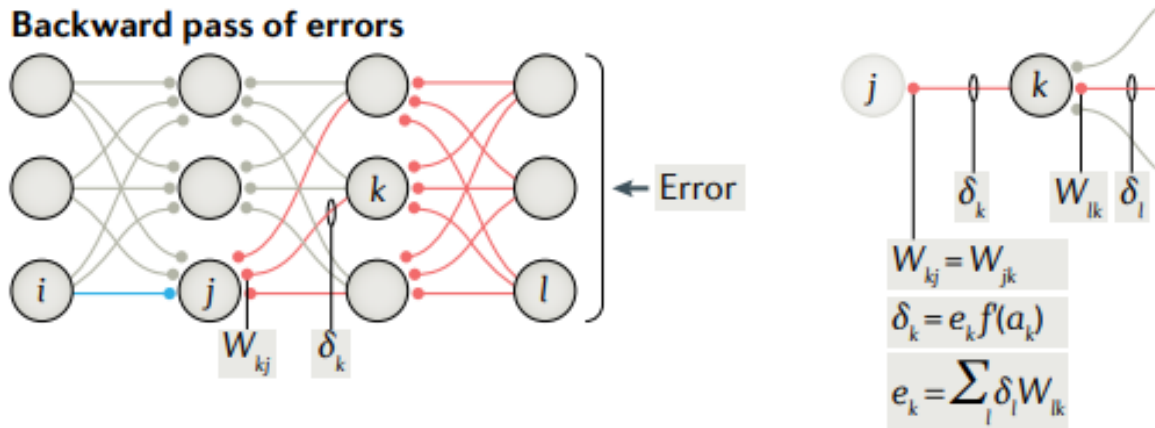


Figura 6 – Retropropagação. Fonte: Lillicrap et al. (2020).

2.2.0.4 Otimização de Redes Neurais

A otimização de redes neurais é um aspecto crucial no treinamento de modelos de aprendizado profundo, pois envolve a escolha de algoritmos e técnicas que permitem ajustar os pesos da rede para minimizar a função de perda.

Um dos algoritmos mais populares para otimização de redes neurais é o *Gradient Descent* (Ruder, 2017). No entanto, existem outros algoritmos, cada um com suas próprias características e vantagens, que podem ser adequados para diferentes tipos de problemas e arquiteturas de redes neurais, como por exemplo o *Stochastic Gradient Descent* (SGD), *Momentum*, *Nesterov Accelerated Gradient* (NAG), Adagrad, Adadelata, RMSprop e Adam.

Para tarefas de visão computacional, Kingma and Ba (2015) mostraram que o Adam é um dos algoritmos de otimização mais eficazes em redes neurais convolucionais. Testes no conjunto de dados MNIST mostraram que o Adam convergiu mais rapidamente e alcançou melhores resultados em comparação com outros algoritmos, como SGD e Adagrad.

2.3 VISÃO COMPUTACIONAL

A visão computacional é um campo da inteligência artificial (IA) e da ciência da computação (Figura 7) que estuda como as máquinas podem adquirir, processar, analisar e compreender imagens e vídeos do mundo real, com o objetivo de produzir representações visuais ou descrever o conteúdo visual de forma automática Team (2024).

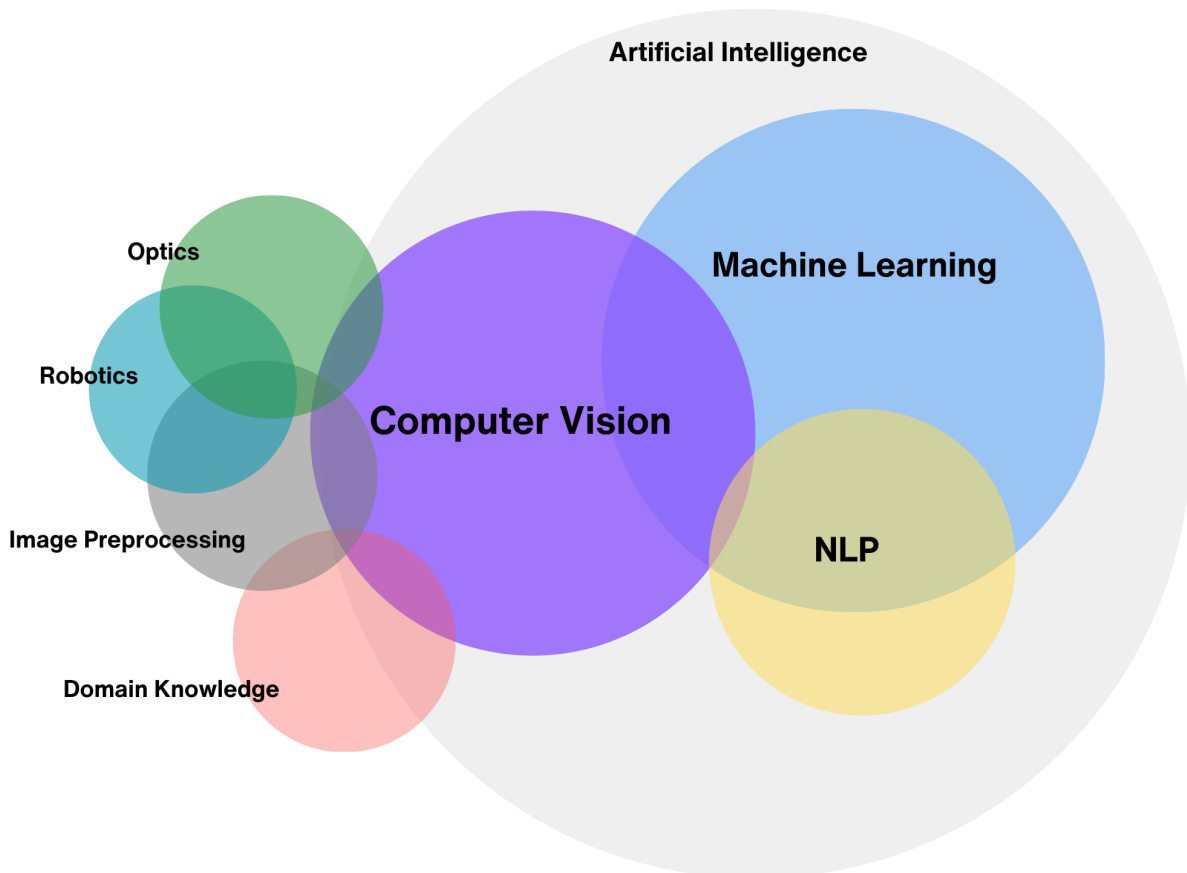


Figura 7 – Definição de visão computacional. Fonte: Team (2024).

Essa tecnologia tem se beneficiado de avanços significativos das últimas décadas, incluindo o desenvolvimento e aperfeiçoamento de algoritmos de aprendizado profundo, que permitem a extração de características complexas e abstratas de imagens e vídeos, o aumento da capacidade computacional com o uso de GPUs (Graphics Processing Units), e o desenvolvimento de grandes conjuntos de dados, como o ImageNet, que contém milhões de imagens rotuladas em centenas de categorias Esteva et al. (2021).

O aprendizado profundo revolucionou a forma como sistemas computacionais processam dados brutos. Tradicionalmente, a construção de modelos exigia conhecimento especializado para extrair manualmente características relevantes dos dados, como bordas, texturas e formas. Enquanto isso, o aprendizado profundo permite que redes neurais descu-

bram automaticamente essas representações em vários níveis de abstração. Esse avanço tem levado a conquistas notáveis em diversas áreas, como reconhecimento de fala, processamento de linguagem natural (PLN) e visão computacional [<https://www.cs.toronto.edu/hinton/absps/NatureDeepReview.pdf>].

Em 1963, Larry Roberts, um dos pioneiros da visão computacional, propôs métodos capazes de compreender objetos em 3D a partir de imagens 2D, representando o marco inicial da área [Machine Perception of Thre Dimensional Solids]. Na próxima década, nos anos 1970 e 1980, pesquisadores desenvolveram algoritmos para detectar bordas e cantos em imagens, modelagem poliédrica e não poliédrica, representação de objetos como interconexões de estruturas menores, fluxo óptico e estimativa de movimento [computer vision: algorithms and applications]. Além disso, a década de 1980 foi marcada pela publicação do artigo "Learning representations by back-propagating errors" de David Rumelhart, Geoffrey Hinton e Ronald Williams, que introduziu o algoritmo de retropropagação, que é amplamente utilizado em modelos de aprendizado profundo até hoje [NatureDeepReview, Learning representations by back-propagating errors].

No entanto, foi apenas na década de 2010 que o aprendizado profundo se tornou popular, com o desenvolvimento de arquiteturas de redes neurais profundas, como as redes neurais convolucionais (CNNs) e as redes neurais recorrentes (RNNs), amplamente adotadas pela comunidade de visão computacional [<https://www.cs.toronto.edu/hinton/absps/NatureDeepReview.pdf>]. Em 2012, a equipe de Geoffrey Hinton, chamada de Super Vision, venceu a competição ImageNet Large Scale Visual Recognition Challenge (ILSVRC) com a rede neural convolucional AlexNet, que alcançou resultados significativamente melhores do que os métodos tradicionais de visão computacional em tarefas de classificação e detecção de objetos em imagens [<https://arxiv.org/pdf/1409.0575>]. Esse avanço marcou a início da era do aprendizado profundo na visão computacional, permitindo a criação de modelos mais sofisticados e precisos, capazes de superar a capacidade humana em tarefas visuais.

Paralelamente, houve uma evolução significativa em hardware computacional, com o desenvolvimento de GPUs, que permitiu treinar modelos de aprendizado profundo em grandes conjuntos de dados de forma mais rápida e eficiente. As GPUs, inicialmente projetadas para renderização de gráficos em jogos e aplicações de design, foram adaptadas para acelerar cálculos matriciais e aplicações computacionalmente intensivas, incluindo o treinamento de redes neurais profundas, devido à sua capacidade de processar informação em paralelo [High performance convolutional neural networks for document processing, Forecasting GPU Performance for Deep Learning Training and Inference]. Em 2009, a equipe de Andrew Ng, da Universidade de Stanford, demonstrou que o uso de GPUs acelerou o treinamento de redes neurais profundas em 70 vezes em comparação com CPUs multi-core, o que permitiu a criação de modelos mais complexos em menos tempo [Large-

scale Deep Unsupervised Learning using Graphics Processors]. A partir de então, GPUs se tornaram uma ferramenta essencial para pesquisadores e instituições que trabalham com aprendizado profundo, trazendo mais agilidade e eficiência para o treinamento de modelos.

Apesar dos avanços em algoritmos de aprendizado profundo e hardware computacional, a visão computacional ainda enfrentava desafios quanto à falta de grandes conjuntos de dados rotulados para treinar os modelos de visão computacional. No entanto, em 2009 foi introduzido o ImageNet, um conjunto de dados com milhões de imagens em centenas de categorias, construído sobre a base de dados WordNet, que contém sinônimos e relações semânticas entre palavras. Com isso, o ImageNet se tornou o maior e mais diverso conjunto de dados de imagens disponível na época [ImageNet: A Large-Scale Hierarchical Image Database]. Mais tarde, em 2012, o ImageNet serviu como base para o ImageNet Large Scale Visual Recognition Challenge (ILSVRC), uma competição anual que avalia algoritmos de classificação e detecção de objetos em larga escala. O desafio foi marcado pela vitória da equipe de Geoffrey Hinton, onde propuseram a rede neural convolucional profunda chamada AlexNet capaz de alcançar altas taxas de acurácia e redução significativa na taxa de erro, marcando assim o início da era do aprendizado profundo e solidificando o papel do ImageNet como catalisador para inovações subsequentes na área [ImageNet Large Scale Visual Recognition Challenge].

[tecnicas da visao computacional]

[aplicacoes gerais]

2.4 VISÃO COMPUTACIONAL NA SAÚDE

No setor da saúde, a visão computacional tem desempenhado um papel crucial na automação de tarefas clínicas, diagnóstico de doenças, monitoramento de pacientes e tratamentos médicos. A análise de imagens médicas, como radiografias, tomografias, ressonâncias magnéticas e ultrassonografias, é uma das principais aplicações da visão computacional na saúde. A partir dessas imagens, é possível detectar e classificar patologias, monitorar o progresso de doenças, avaliar a eficácia de tratamentos e até mesmo recomendar tratamentos personalizados para pacientes. Além disso, a visão computacional também tem sido utilizada para automatizar tarefas clínicas, como a identificação de pacientes, a análise de exames laboratoriais e a triagem de pacientes em hospitais e clínicas [Javaid et al. \(2024\)](#).

Na área da saúde, a visão computacional tem sido muito utilizada para melhorar a acurácia de diagnósticos, automatização de tarefas clínicas e tratamentos médicos. Ao analisar imagens médicas, como radiografias, tomografias, ressonâncias magnéticas e ultrassonografias, a máquina pode detectar e classificar patologias com maior precisão e rapidez do que um médico humano. Além disso, a visão computacional pode ser utilizada

para monitorar o progresso de doenças, monitorar a eficácia de tratamentos e até mesmo recomendar tratamentos personalizados para pacientes [Javaid et al. \(2024\)](#).

2.5 APRENDIZADO PROFUNDO

O uso de modelos de aprendizado profundo baseados em redes neurais convolucionais (RNCs) tem ganhado espaço em tarefas de visão computacional. Aprendizado por transferência também é amplamente utilizado para reduzir o uso de recursos computacionais para tarefas que já são executadas por modelos existentes, como as redes residuais (ResNet), Visual Geometry Group (VGG) e as redes densamente conectadas (DenseNet) [Tariq et al. \(2023\)](#). Enquanto o uso de RNCs tem se mostrado útil em soluções de detecção em imagens médicas, a operação de convolução limita o relacionamento entre pixels distantes numa imagem. Para tanto, a habilidade de codificar dependências de longo alcance tem sido possível graças às arquiteturas de aprendizado profundas baseadas em atenção, como o Vision Transformer (ViT). Tais modelos de ViT têm sido empregados para várias tarefas, incluindo classificação e detecção de objetos [Shamshad et al. \(2023\)](#).

3 Metodologia

Esta seção descreve a metodologia proposta para a tarefa de classificação da OA de joelho a partir de radiografias. A principal abordagem desta pesquisa consiste no uso de *transfer learning* para aproveitar o conhecimento já obtido por modelos pré-treinados e melhorar a performance da predição final.

3.1 COLETA DE DADOS

A escolha e coleta dos dados é a primeira tarefa a ser realizada quando o objetivo é treinar um modelo de aprendizado profundo, incluindo redes neurais artificiais e vision transformers. Um conjunto de dados adequado é essencial para que o modelo tenha uma boa performance e seja útil para se tornar uma ferramenta de suporte no diagnóstico de OA de joelho. O conjunto de dados foi obtido a partir da plataforma Kaggle [Chen \(2018\)](#), uma fonte amplamente reconhecida por fornecer dados de alta qualidade e de domínio público para estudos acadêmicos e projetos de aprendizado de máquina. O conjunto de dados escolhido é baseado na Osteoarthritis Initiative (OAI), um estudo observacional multicêntrico de dez anos de homens e mulheres, patrocinado pelo National Institutes of Health (NIH), com o objetivo de permitir uma melhor compreensão da prevenção e tratamento da osteoartrite de joelho [of Health \(2024\)](#). Este conjunto contém radiografias de joelhos, juntamente com suas respectivas classificações de severidade da OA, conforme o sistema de Kellgren/Lawrence. Este dataset foi selecionado por sua relevância na plataforma, fornecendo uma base sólida para o treinamento dos modelos de RNCs e ViTs propostos nesta pesquisa. A [Figura 8](#) ilustra a distribuição do conjunto de dados entre treino, teste e validação.

O conjunto de dados contém quatro pastas nomeadas “auto_test”, “test”, “train” e “val”, cada uma contendo as subpastas com imagens 224x224 representando cada um dos graus de KL. O dataset foi dividido entre dados de treino, teste e validação, com uma proporção de 7:2:1. O conjunto de treino é usado para treinar os modelos e consiste na maior proporção de imagens. O conjunto de validação é usado para ajustar os hiperparâmetros do modelo e monitorar o seu desempenho, enquanto o conjunto de teste é usado após o treinamento completo do modelo, para medir o desempenho final e verificar sua capacidade de generalização em dados completamente novos.

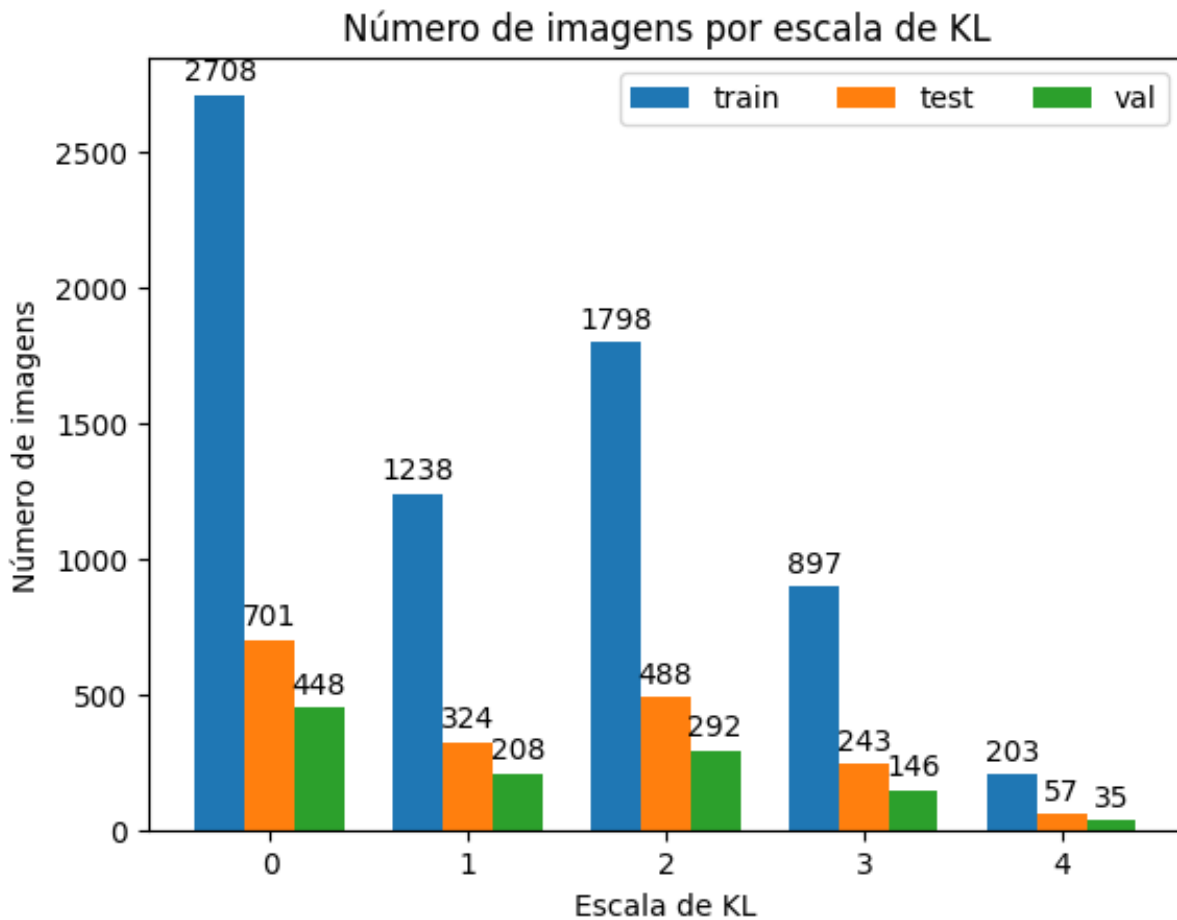


Figura 8 – Número de imagens em cada classe do conjunto de dados

3.2 PRÉ-PROCESSAMENTO DAS IMAGENS

O pré-processamento de imagens de raio-X é crucial para melhorar a qualidade e facilitar a análise automatizada pelos modelos. Para isso, algumas técnicas devem ser utilizadas, incluindo:

3.2.1 Normalização

A normalização de dados visa ajustar os valores para um intervalo padrão, melhorando a consistência dos dados e a eficiência dos modelos treinados. Para as radiografias, os pixels devem ter seus valores transformados para o intervalo entre 0 e 1.

3.2.2 Equalização de Histograma

A equalização de histograma é um método de processamento de imagem que busca melhorar o contraste e a visibilidade dos detalhes em uma imagem. Para isso, esta técnica redistribui os níveis de cinza da imagem, de forma que a distribuição dos valores de intensidade seja mais uniforme. Isso é feito calculando o histograma acumulado da

imagem original e utilizando-o para redistribuir os valores de cada pixel. Em particular, a equalização de histograma é útil para radiografias, onde a variação de intensidade pode ser sutil e a distinção entre diferentes graus de KL pode ser difícil.

Dada uma imagem $I(x, y)$ com intensidades de pixel $i \in \{0, 1, \dots, L - 1\}$ e número total de pixels N , onde x e y são as coordenadas do pixel, e L é o número de níveis de intensidade (normalmente $L = 256$ para imagens de 8 bits), a probabilidade de ocorrência de cada intensidade i é calculada como:

$$p(i) = \frac{n(i)}{N}$$

onde $n(i)$ é o número de pixels com intensidade i na imagem. O histograma acumulado H é então calculado como:

$$H(i) = \sum_{j=0}^i p(j)$$

O novo valor de intensidade j para um pixel com intensidade i é calculado como:

$$j = (L - 1) \times H(i)$$

onde $L - 1$ garante que o novo valor esteja no intervalo de intensidade da imagem. O resultado da equalização é normalmente arredondado para o valor inteiro mais próximo.

3.2.3 Aumento de dados

A ideia desta técnica é expandir artificialmente o tamanho e a variabilidade de um conjunto de dados, principalmente quando o volume de dados disponível é limitado. Isso torna os modelos mais robustos e genéricos, prevenindo *overfitting* e melhorando o desempenho em dados novos. As técnicas de aumento de dados que serão utilizadas nas radiografias são: rotação e reflexão (espelhamento) horizontal.

3.3 ARQUITETURA DO MODELO DE REDE NEURAL CONVOLUCIONAL

As redes neurais convolucionais possuem um papel muito relevante no contexto de inteligência artificial, especialmente em tarefas de visão computacional devido à sua capacidade de extrair características relevantes de imagens de forma automática, sem qualquer intervenção manual. Sua arquitetura é especialmente eficaz para reconhecer e classificar objetos em imagens complexas, inclusive em radiografias, com o intuito de

auxiliar no processo de diagnóstico médico. As RNCs conseguem identificar variações sutis que podem estar associadas a condições patológicas, como é o caso da osteoartrite de joelho, onde as variações entre os graus de KL reside no espaçamento articular da junção do joelho.

Fazer o treinamento de uma RNC sem nenhum conhecimento prévio do modelo é custoso em termos de quantidade de dados necessário, consumo de recursos computacionais e tempo. Para resolver este problema, o uso de *transfer learning* é essencial, pois permite aproveitar modelos já treinados em grandes conjuntos de dados genéricos, como o ImageNet, e adaptá-los para o conjunto de dados específico para o problema. Ao utilizar o *transfer learning*, as primeiras camadas do modelo, que capturam características gerais da imagem, são congeladas, enquanto as camadas finais são ajustadas para a tarefa específica, tal processo é chamado de *fine-tuning*. Isso economiza tempo e recursos computacionais e aumenta a eficácia do treinamento, resultando em modelos que podem fornecer diagnósticos precisos mesmo com volumes menores de dados disponíveis. Nos últimos anos, algumas arquiteturas performaram muito bem em algumas tarefas, como por exemplo a ResNet, VGG, Inception (GoogLeNet) e DenseNet. A arquitetura para os modelos de RNC pode ser vista na Figura ??.

3.3.1 VGG (Visual Geometry Group Network)

Os modelos VGG foram introduzidos pelo *Visual Geometry Group* da Universidade de Oxford por [Simonyan and Zisserman \(2015\)](#), que depois serviu como base para a competição do *ImageNet* em 2014, quando conquistaram o primeiro e segundo lugar na época. A arquitetura VGG é conhecida por sua simplicidade e profundidade, utilizando filtros convolucionais pequenos (3×3) empilhados em camadas profundas, variando de 11 a 19 camadas. O objetivo dos autores era explorar o impacto da profundidade na performance do modelo, e eles descobriram que redes neurais mais profundas superavam redes mais rasas, desde que treinadas adequadamente.

A arquitetura VGG processa imagens RGB de 224×224 pixels, utilizando uma série de camadas convolucionais seguidas por camadas de *pooling*, onde cada camada contém um número crescente de filtros 3×3 . O *stride* é fixo em 1 pixel, e o *padding* é utilizado para manter a dimensão da imagem. Após as camadas convolucionais, são aplicadas camadas de *max-pooling* com um tamanho de 2×2 e *stride* de 2, reduzindo a dimensão da imagem pela metade. Por fim, são adicionadas três camadas totalmente conectadas (ou *fully connected* do inglês), seguidas por uma camada de saída com ativação *softmax* para classificação. Além disso, as camadas escondidas são ativadas por funções ReLU, reponsáveis por introduzir a não-linearidade no modelo.

A tabela 2 apresenta a configuração das arquiteturas VGG-16 e VGG-19, com um total de 16 e 19 camadas, respectivamente. Ambas se destacaram na competição do

ImageNet e são amplamente utilizadas devido à sua performance em tarefas de classificação de imagens médicas (Saini et al., 2023; Sitaula and Hossain, 2021). Por esse motivo, estas arquiteturas serão utilizadas nesta pesquisa como comparação com os demais modelos.

VGG-16	VGG-19
16 camadas	19 camadas
imagem RGB de entrada (224 x 224)	
conv3-64	conv3-64
conv3-64	conv3-64
maxpool	
conv3-128	conv3-128
conv3-128	conv3-128
maxpool	
conv3-256	conv3-256
conv3-256	conv3-256
conv3-256	conv3-256
	conv3-256
maxpool	
conv3-512	conv3-512
conv3-512	conv3-512
conv3-512	conv3-512
	conv3-512
maxpool	
conv3-512	conv3-512
conv3-512	conv3-512
conv3-512	conv3-512
	conv3-512
maxpool	
FC-4096	
FC-4096	
FC-1000	
softmax	

Tabela 2 – Configuração dos modelos VGG-16 e VGG-19. Os parâmetros de cada camada convolucional são denotados por "conv<tamanho do campo receptivo>-<número de canais>". A função de ativação ReLU não é exibida por motivos de simplicidade.

3.3.2 ResNet (Residual Network)

He et al. (2016) venceram a competição ILSVRC 2015 com a arquitetura *Residual Network* (*ResNet*), que introduziu a ideia de blocos residuais e alcançou uma taxa de erro de 3,57% no conjunto de validação do *ImageNet* com um *ensemble* de seus modelos. Os autores abordaram o problema da degradação de desempenho: conforme a profundidade da rede aumentava, a acurácia saturava e começava a diminuir. Para resolver, eles introduziram a ideia de conexões de atalho (*skip connections*) entre as camadas, onde o sinal de entrada de uma camada é somado ao sinal de saída de uma camada subsequente (Figura 9).

Formalmente, considerando que o objetivo de uma rede neural é aprender uma função $H(x)$, onde x é a entrada, a ResNet propõe que a rede aprenda uma função residual $F(x) = H(x) - x$, onde a entrada x é adicionada à saída $H(x)$, reformulando a função de

aprendizado como $H(x) = F(x) + x$. Essa abordagem permite que a rede aprenda funções de identidade mais facilmente, facilitando o treinamento de redes mais profundas sem adicionar complexidade.

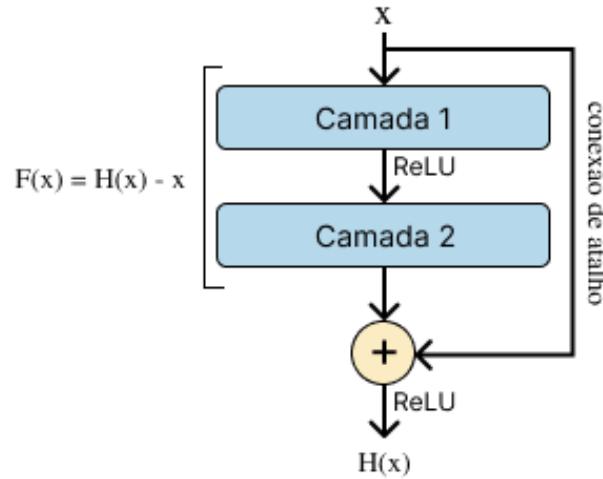


Figura 9 – Aprendizado residual

A arquitetura ResNet é composta por pilhas de blocos residuais que consistem em duas camadas convolucionais, com um *Batch Normalization* e uma função de ativação ReLU entre elas. As camadas convolucionais utilizam filtros de tamanho 3×3 , com um *stride* de 1 e *padding* de 1, para manter a dimensão da imagem. A saída do bloco residual é então somada à entrada original, permitindo que o modelo aprenda a função residual. A rede termina com uma camada de *average pooling* global e uma camada totalmente conectada (ou *fully connected* do inglês) com ativação *softmax* para classificação.

A tabela 3 apresenta a configuração das arquiteturas ResNet-34, ResNet-50 e ResNet-101, que são variantes da ResNet com diferentes profundidades. Essas arquiteturas foram escolhidas devido à sua popularidade e eficácia em tarefas de classificação de imagens, especialmente em radiografias (Leung et al., 2020).

3.3.3 DenseNet (Densely Connected Convolutional Networks)

A arquitetura DenseNet introduziu uma nova abordagem para lidar com redes profundas e aliviar o problema de *vanishing gradients*, melhorando a propagação e reuso da informação, além de diminuir o número de parâmetros. A ideia principal foi conectar cada camada a todas as camadas anteriores, formando conexões densas entre elas. Isso significa que cada camada recebe como entrada não apenas a saída da camada anterior, mas também as saídas de todas as camadas anteriores (Figura 10). Essa abordagem permite que o modelo aprenda representações mais ricas e complexas, facilitando a extração de características relevantes para a tarefa de classificação (Huang et al., 2017).

Camada	Tamanho da saída	34 camadas	50 camadas	101 camadas
conv1	112×112	7×7, 64, stride 2		
		3×3 max pool, stride 2		
conv2_x	56×56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax		
FLOPs		3.6×10^9	3.8×10^9	7.6×10^9

Tabela 3 – Configuração das arquiteturas ResNet-34, ResNet-50 e ResNet-101.

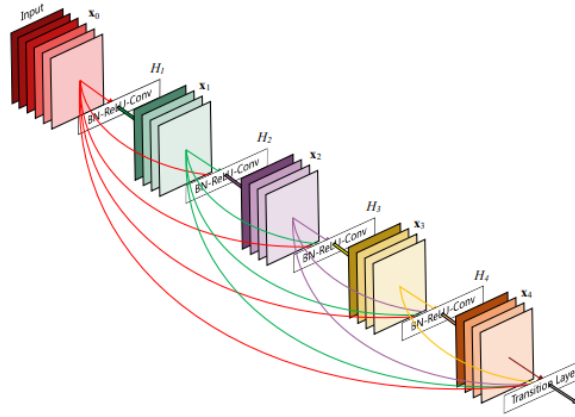


Figura 10 – Um bloco de 5 camadas de uma DenseNet. Cada camada recebe como entrada a saída de todas as camadas anteriores.

O componente fundamental da DenseNet é o bloco denso (ou *dense block* em inglês), que consiste em várias camadas convolucionais conectadas densamente. Cada camada dentro do bloco denso aplica três operações consecutivas: *batch normalization* (BN), seguido de uma função de ativação ReLU e, por fim, uma convolução 3×3 . Após a aplicação do bloco denso, uma transição (ou *transition* em inglês) é realizada para reduzir a dimensão dos *feature maps* usando uma camada de convolução 1×1 , seguida por uma camada de *average pooling* 2×2 .

Portanto, a arquitetura DenseNet é composta por quatro blocos densos, cada um seguido por camadas de transição. A saída final (classificador) é obtida através de uma camada de *global average pooling* e uma camada totalmente conectada com ativação *softmax* para classificação. A tabela 4 apresenta a configuração das arquiteturas DenseNet-121 e DenseNet-169, que são variantes da DenseNet com diferentes profundidades. Essas arquiteturas foram escolhidas devido à sua popularidade e eficácia em tarefas de classificação

de imagens, especialmente em radiografias (Leung et al., 2020).

Camadas	Tamanho da saída	DenseNet-121	DenseNet-169
Convolução	112×112	7×7 conv, stride 2	
Pooling	56×56	3×3 max pool, stride 2	
Dense Block (1)	56×56	$\begin{bmatrix} 1 \times 3 \\ 3 \times 3 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 3 \\ 3 \times 3 \end{bmatrix} \times 6$
Transition Layer (1)	56×56	1×1 conv	
	28×28	2×2 average pool, stride 2	
Dense Block (2)	28×28	$\begin{bmatrix} 1 \times 3 \\ 3 \times 3 \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 3 \\ 3 \times 3 \end{bmatrix} \times 12$
Transition Layer (2)	28×28	1×1 conv	
	14×14	2×2 average pool, stride 2	
Dense Block (3)	14×14	$\begin{bmatrix} 1 \times 3 \\ 3 \times 3 \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 3 \\ 3 \times 3 \end{bmatrix} \times 32$
Transition Layer (3)	14×14	1×1 conv	
	7×7	2×2 average pool, stride 2	
Dense Block (4)	7×7	$\begin{bmatrix} 1 \times 3 \\ 3 \times 3 \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 3 \\ 3 \times 3 \end{bmatrix} \times 32$
Classification Layer	1×1	7×7 global average pool	
		1000D fully-connected, softmax	

Tabela 4 – Configuração das arquiteturas DenseNet-121 e DenseNet-169.

3.3.4 Inception-v3

A arquitetura Inception, introduzida por Szegedy et al. (2015) no contexto do desafio ILSVRC 2014, representou um avanço significativo na evolução das redes neurais convolucionais. Seu principal diferencial está na proposta de uma estrutura modular - o módulo Inception - que combina convoluções de diferentes tamanhos (1×1 , 3×3 , 5×5) e operações de *pooling* em paralelo, promovendo o processamento de informações em múltiplas escalas (Figura 11).

O modelo GoogLeNet, uma instância da arquitetura Inception com 22 camadas profundas, obteve o primeiro lugar no ILSVRC 2014 (Russakovsky et al., 2015), alcançando um notável desempenho em tarefas de classificação e detecção, mesmo utilizando significativamente menos parâmetros que modelos anteriores, como o VGG.

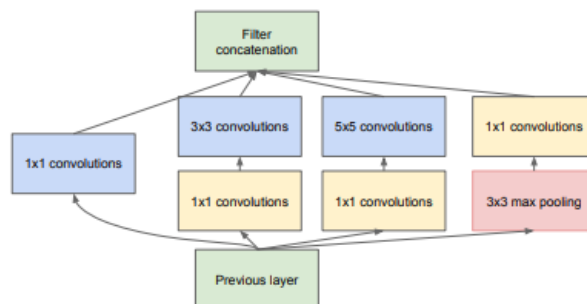


Figura 11 – Um módulo Inception.

A arquitetura Inception-v3 (Szegedy et al., 2016) representa uma evolução significativa em relação ao modelo original Inception (GoogLeNet), incorporando diversas inovações voltadas à melhoria da eficiência computacional e da acurácia. Entre as principais contribuições estão a fatoração de convoluções em operações menores e assimétricas (Figura 12), o uso mais sistemático da normalização em lote (*batch normalization*) e a adoção da técnica de *label smoothing* como forma de regularização. Tais aprimoramentos resultaram em um modelo mais profundo e preciso, mantendo um custo computacional viável para aplicações práticas.

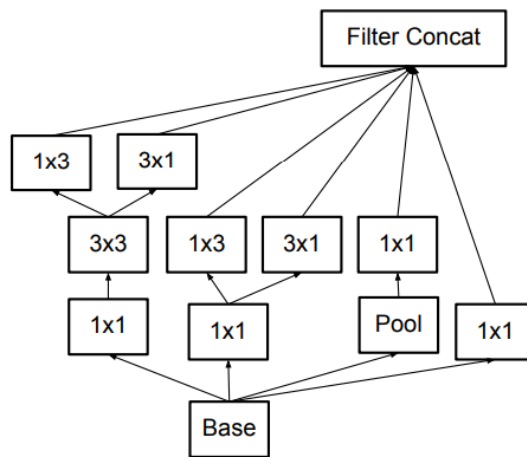


Figura 12 – Um módulo Inception com fatoração de convoluções.

A Tabela 5 apresenta a configuração da arquitetura Inception-v3, com um total de 42 camadas, que inclui a fatoração de convoluções tradicionais 7×7 em convoluções 3×3 . A arquitetura substitui o otimizador padrão do SGD por um otimizador mais avançado, o RMSProp, favorecendo a convergência do modelo durante o treinamento, além de utilizar classificadores auxiliares com normalização em lote nas camadas intermediárias, melhorando a propagação do sinal do gradiente e, por consequência, a eficiência do treinamento.

type	patch size/stride	input size
conv	$3 \times 3/2$	$299 \times 299 \times 3$
conv	$3 \times 3/1$	$149 \times 149 \times 32$
conv padded	$3 \times 3/1$	$147 \times 147 \times 32$
pool	$3 \times 3/2$	$147 \times 147 \times 64$
conv	$3 \times 3/1$	$73 \times 73 \times 64$
conv	$3 \times 3/2$	$71 \times 71 \times 80$
conv	$3 \times 3/1$	$35 \times 35 \times 192$
3×Inception		$35 \times 35 \times 288$
5×Inception		$17 \times 17 \times 768$
2×Inception		$8 \times 8 \times 1280$
pool	8×8	$8 \times 8 \times 2048$
linear	logits	$1 \times 1 \times 2048$
softmax	classifier	$1 \times 1 \times 1000$

Tabela 5 – Configuração da arquitetura Inception-v3.

Além de seu excelente desempenho na tarefa de classificação de imagens do ILSVRC 2012 (Russakovsky et al., 2015), a arquitetura Inception-v3 tem sido utilizada em outras aplicações, incluindo o diagnóstico médico. Por exemplo, Mujahid et al. (2022) adotaram a arquitetura Inception-v3 para a tarefa de classificação de pneumonia em radiografias e obtiveram resultados promissores, alcançando uma acurácia de 99,29%, com um ensemble, e superando outros modelos, como VGG-16 e ResNet-50.

3.4 ARQUITETURA DO MODELO DE VISION TRANSFORMER

A arquitetura Vision Transformer tem se destacado como uma abordagem poderosa para tarefas de visão computacional devido à sua capacidade de capturar relações globais em imagens através do mecanismo de atenção (Dosovitskiy et al. (2021)). Essa abordagem permite que os modelos de ViTs alcancem ótimos resultados e superem as limitações das RNCs, que focam mais em características locais da imagem. Tal capacidade é particularmente relevante para o diagnóstico de patologias em imagens médicas, incluindo radiografias, onde o modelo é capaz de processar toda a imagem simultaneamente, associando partes distantes e próximas com igual relevância. Além disso, os ViTs também se beneficiam do *transfer learning*, permitindo que os modelos sejam treinados de forma eficiente em conjuntos de dados limitados. Para esta pesquisa será feito o *fine-tuning* de alguns modelos de ViT para a tarefa de classificação da OA de joelho, como o ViT-B/16, DeiT (Data-efficient Image Transformer), Swin Transformer (Shifted Window Transformer) e ResNet50-ViT-B/16. A arquitetura para os modelos de ViT pode ser vista na Figura 13.

3.4.1 ViT-B/16

O ViT-B/16 (Dosovitskiy et al. (2021)) é uma das primeiras variantes da arquitetura Vision Transformer, onde "B" representa o modelo base (base model) e "16" refere-se ao tamanho do *patch* em que a imagem é dividida (16x16 pixels). O ViT-B/16 recebe uma imagem e a divide em *patches*, tratando cada *patch* como um *token*, semelhante ao processamento de palavras em texto nos *transformers* tradicionais. O modelo usa um mecanismo de atenção para processar os *tokens* de maneira global, capturando interdependências entre diferentes regiões da radiografia. Essa abordagem permite que o ViT-B/16 compreenda melhor a estrutura geral da imagem, identificando padrões que podem se estender por grandes áreas da mesma. Este modelo pode ser especialmente eficaz para a tarefa de classificação da OA de joelho, visto que existe o padrão notável do espaçamento articular que se estende horizontalmente na radiografia.

3.4.2 DeiT (Data-efficient Image Transformer)

O DeiT [Touvron et al. \(2021\)](#) é uma versão otimizada dos ViTs, projetada para melhorar a eficiência no uso de dados. Enquanto os ViTs originais, como o ViT-B/16, geralmente precisam de grandes quantidades de dados para atingir um bom desempenho, o DeiT foi projetado para ser treinado em conjunto de dados reduzidos. Isso acontece devido à técnica do *distillation token*, que permite ao modelo aprender a partir de um "professor" (modelo mais simples), aumentando a eficiência do treinamento. Este modelo pode ser particularmente útil na tarefa de classificação da OA de joelho, podendo ser um importante fator ao comparar com outros modelos de ViTs e RNCs.

3.4.3 Swin Transformer (Shifted Window Transformer)

O Swin Transformer [Liu et al. \(2021\)](#) é uma arquitetura de ViT que introduz uma abordagem nova que utiliza *hierarchical feature maps* e *sliding windows* para aplicar a atenção e melhorar a eficiência e performance do modelo. Em vez de processar toda a imagem como uma sequência de *patches* globalmente, o Swin Transformer aplica a atenção dentro de pequenas janelas locais, de forma hierárquica, permitindo que o modelo mantenha a eficiência computacional e ainda capture detalhes locais e globais. Conforme o modelo avança pelas camadas, as janelas se expandem e se deslocam, permitindo que o modelo agregue contexto global ao longo do processamento. Essa estrutura hierárquica é particularmente eficaz para imagens de alta resolução, como as radiografias, onde há muitos detalhes importantes em diferentes escalas. Além disso, o Swin Transformer pode ser facilmente escalado para diferentes tamanhos de imagens e é altamente eficiente em termos de uso de memória e poder computacional, sendo uma escolha apropriada para a tarefa de classificação da OA de joelho.

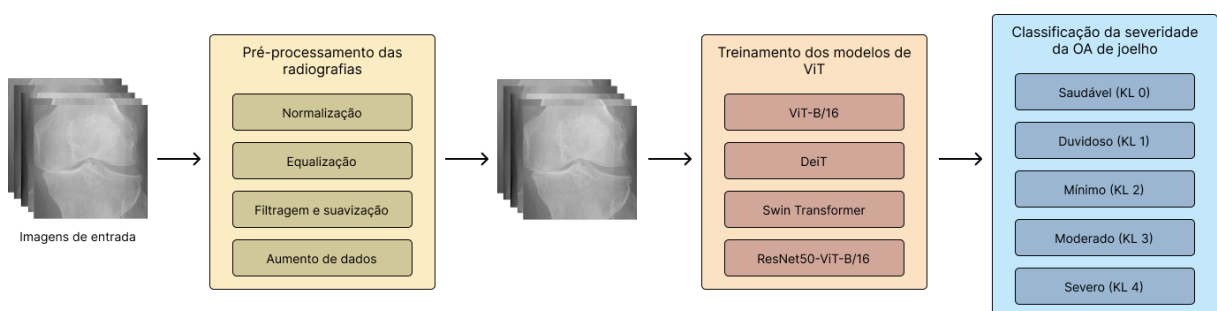


Figura 13 – Metodologia para os vision transformers

3.5 MÉTRICAS DE AVALIAÇÃO

Para comparar a performance dos modelos treinados na tarefa de classificação da severidade da OA de joelho, serão utilizadas as seguintes métricas de avaliação: acurácia,

precisão, revocação, F1-score e matriz de confusão. Essas métricas são amplamente utilizadas em problemas de classificação para medir a qualidade das previsões e o equilíbrio entre os diferentes tipos de erros. Para o cálculo das métricas, os seguintes acrônimos serão utilizados nas fórmulas:

- TP é o número de verdadeiros positivos,
- TN é o número de verdadeiros negativos,
- FP é o número de falsos positivos,
- FN é o número de falsos negativos.

3.5.1 Acurácia

A acurácia mede a proporção de previsões corretas em relação ao total de exemplos. Ela pode ser calculada pela fórmula:

$$\text{Acurácia} = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.1)$$

3.5.2 Precisão

A precisão indica a proporção de exemplos classificados como positivos que realmente são positivos. Ela é calculada pela fórmula:

$$\text{Precisão} = \frac{TP}{TP + FP} \quad (3.2)$$

3.5.3 Recall

O recall mede a capacidade do modelo de identificar corretamente todos os exemplos positivos. É definido como:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3.3)$$

3.5.4 F1-Score

O F1-score é a média harmônica entre precisão e recall, e é uma métrica útil quando busca-se um equilíbrio entre os dois. A fórmula do F1-score é:

$$F1 = 2 \cdot \frac{\text{Precisão} \cdot \text{Recall}}{\text{Precisão} + \text{Recall}} \quad (3.4)$$

3.5.5 Matriz de Confusão

A matriz de confusão é uma ferramenta para visualizar o desempenho do modelo de classificação, detalhando as previsões corretas e incorretas em cada classe. Ela apresenta os valores de TP , TN , FP e FN de forma estruturada, permitindo avaliar o desempenho em classes específicas.

	Previsto Positivo	Previsto Negativo
Verdadeiro Positivo	TP	FN
Verdadeiro Negativo	FP	TN

3.5.6 AUC-ROC

Para tarefas de classificação binária, será utilizada também a métrica AUC-ROC (Área Sob a Curva da Característica de Operação do Receptor), que mede a capacidade do modelo de separar as classes positivas e negativas. A curva ROC é um gráfico que exibe a taxa de verdadeiros positivos (sensibilidade) em função da taxa de falsos positivos.

$$\text{AUC-ROC} = \int_0^1 \text{TPR}(FPR) dFPR \quad (3.5)$$

onde TPR é a taxa de verdadeiros positivos e FPR é a taxa de falsos positivos.

3.6 EFICIÊNCIA COMPUTACIONAL

Além da performance dos modelos, a eficiência computacional é um fator importante a ser considerado devido ao custo associado ao treinamento e inferência dos modelos. Para avaliar essa métrica, serão considerados o tempo de treinamento e a quantidade de computação utilizada por cada modelo. O tempo de treinamento, medido em minutos, é calculado da seguinte forma:

$$\text{Tempo de Treinamento} = \text{Tempo Final} - \text{Tempo Inicial} \quad (3.6)$$

onde o tempo final é o momento em que o treinamento é concluído e o tempo inicial é o momento em que ele é iniciado. A quantidade de computação é medida em FLOPs (operações de ponto flutuante por segundo), que representa uma métrica largamente utilizada para medir a quantidade de cálculos necessários para treinar um modelo de aprendizado profundo e está diretamente relacionada à eficiência computacional. A quantidade de FLOPs pode ser calculada de uma forma aproximada a partir do número de operações nas etapas de *forward* e *backward* do modelo, o número de exemplos no conjunto de dados e o número de épocas de treinamento [Lohn and Musser \(2022\)](#). Para

esta pesquisa, será utilizada a biblioteca *FLOPs Counter PyTorch* [Sovrasov \(2018-2024\)](#) para calcular a quantidade de FLOPs de cada modelo.

3.7 MÉTODO DE VISUALIZAÇÃO

A visualização é uma técnica importante para avaliar quais foram as regiões da imagens que ajudaram o modelo a fazer determinada previsão. O método de visualização Grad-CAM (Gradient-weighted Class Activation Mapping) é uma técnica usada para interpretar e visualizar as decisões feitas por redes neurais convolucionais (RNCs). Em tarefas de classificação, como a avaliação da severidade da OA de joelho a partir de radiografias, entender quais regiões da imagem contribuíram para a decisão do modelo é crucial para a validação e a confiança nos resultados do modelo.

O Grad-CAM fornece mapas de ativação que mostram quais partes da imagem foram mais influentes para a predição de uma classe específica [Selvaraju et al. \(2016\)](#). Para isso, essa técnica utiliza os gradientes da saída da camada final da rede em relação às ativações das camadas intermediárias para gerar uma visualização da importância das regiões da imagem.

Primeiro, é gerado um mapa de localização a partir da RNC utilizada para classificar a imagem usando a técnica do Class Activation Mapping (CAM). O CAM utiliza mapas de características convolucionais, que são globalmente agrupados usando a técnica de *Global Average Pooling* (GAP) e transformados linearmente para produzir uma pontuação y_c para cada classe c . Especificamente, se a penúltima camada da RNC produz K mapas de características $A_k \in \mathbb{R}^{u \times v}$, esses mapas são agrupados espacialmente e combinados linearmente para gerar a pontuação:

$$y_c = \sum_k w_{ck} \frac{1}{Z} \sum_i \sum_j A_{k_{ij}}$$

Para produzir o mapa de localização L_c^{CAM} para a classe c , CAM calcula a combinação linear dos mapas de características finais usando os pesos aprendidos da camada final:

$$L_c^{CAM} = \sum_k w_{ck} A_k$$

Este mapa é então normalizado para o intervalo entre 0 e 1 para fins de visualização.

Em seguida, os gradientes são então globalmente averiguados (*pooling*) para obter pesos que indicam a importância de cada canal de ativação. Esses pesos são usados para ponderar as ativações da camada convolucional final. A seguinte fórmula representa este cálculo dos pesos:

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k}$$

O peso α_k^c representa a linearização parcial da rede e captura a importância de k para a classe c . Por fim, o mapa de ativação é obtido ao multiplicar as ativações ponderadas pelos pesos dos gradientes. Esse mapa é então normalizado e sobreposto na imagem original para mostrar as áreas mais influentes na decisão do modelo.

A fórmula para o Grad-CAM pode ser expressa como:

$$\text{Grad-CAM} = \text{ReLU} \left(\sum_k \alpha_k^c A^k \right)$$

Para esta pesquisa, a utilização do Grad-CAM permitirá a visualização das regiões das radiografias que o modelo considera mais relevantes para suas decisões de classificação. Isso não só facilita a interpretação dos resultados do modelo, mas também ajuda na validação de sua eficácia ao garantir que o modelo está focando nas áreas corretas da imagem, como o espaço articular do joelho.

4 Resultados

Esta pesquisa explora a transferência de aprendizado utilizando modelos pré-treinados no dataset ImageNet, aplicando ajuste fino para classificar o nível de severidade da osteoartrite de joelho com base na escala de Kellgren/Lawrence. O treinamento dos modelos foi realizado com a linguagem de programação Python, através de notebooks disponibilizados pela plataforma Google Colab, aproveitando os recursos computacionais de uma GPU T4 para acelerar o treino e a experimentação.

A classificação da osteoartrite de joelho foi organizada em quatro cenários: (i) classificação com 5 classes, (ii) classificação com 4 classes, (iii) classificação com 3 classes e (iv) classificação binária. O primeiro cenário foi o mais complexo, pois envolveu a classificação em cinco classes distintas (KL0, KL1, KL2, KL3 e KL4), enquanto os outros cenários foram simplificações do primeiro, reduzindo o número de classes a serem classificadas. A escolha do cenário de cinco classes foi motivada pela necessidade de uma classificação mais detalhada e precisa da osteoartrite de joelho, permitindo uma melhor avaliação da progressão da doença.

Para garantir a consistência entre os modelos, o treinamento foi realizado utilizando os mesmos hiperparâmetros. O conjunto de dados foi dividido em 70% para treinamento, 10% para teste e 20% para validação. Para mitigar possíveis vieses, a base de dados foi balanceada por meio de técnicas de *undersampling* e *oversampling*, limitando cada classe a um máximo de 1700 imagens, complementadas por estratégias de *data augmentation*. O treinamento foi configurado com *batches* de 28 imagens, executado ao longo de 30 épocas, com um *early stopping* com 5 épocas de paciência para evitar *overfitting*.

Neste estudo, foi utilizada duas abordagens de função de perda: *crossentropy* e *CORN* (Conditional Ordinal Regression for Neural Networks). A função de perda *crossentropy* é amplamente utilizada em tarefas de classificação, enquanto a função de perda CORN é projetada especificamente para problemas de classificação ordinal, onde a ordem das classes é relevante. Essa abordagem é particularmente útil em cenários onde as classes não são mutuamente exclusivas, como na classificação da osteoartrite de joelho, onde os estágios da doença são sequenciais e possuem uma relação ordinal. Em ambos os casos, o otimizador adotado foi o Adam, configurado com uma taxa de aprendizado inicial de 0.0001, ajustada dinamicamente a cada 3 épocas.

4.0.1 Classificação em Cinco Classes

A ?? apresenta os resultados dos modelos de RNCs e ViTs treinados para a classificação da OA de joelho usando a função de perda *crossentropy*. A acurácia geral dos

Tabela 6 – Desempenho dos modelos utilizando a abordagem Cross Entropy

Modelo	Accuracy	Kappa	QWK	MAE
resnet34	0.7027	0.5899	0.8422	0.3469
resnet50	0.7104	0.6014	0.8478	0.3354
resnet101	0.6961	0.5800	0.8406	0.3541
vgg16	0.6922	0.5691	0.8290	0.3696
vgg19	0.7060	0.5888	0.8404	0.3497
densenet121	0.7204	0.6110	0.8514	0.3271
densenet169	0.7220	0.6158	0.8519	0.3249
inception_v3	0.7022	0.5931	0.8500	0.3376
google_vit	0.7005	0.5878	0.8380	0.3497
facebook_deit	0.6939	0.5775	0.8342	0.3585
davit	0.7358	0.6362	0.8647	0.3023
maxvit_t	0.7220	0.6165	0.8616	0.3155
gcvit	0.7363	0.6372	0.8725	0.2940
swin_b	0.6977	0.5805	0.8425	0.3508

modelos variou de 0.6922 a 0.7363, com o modelo GCViT apresentando a maior acurácia geral de 0.7363. Além disso, o modelo GCViT também obteve o melhor desempenho em termos de Kappa, QWK e MAE, com valores de 0.6372, 0.8725 e 0.2940, respectivamente. Esses resultados indicam que o modelo GCViT foi capaz de aprender padrões relevantes para a classificação da OA de joelho, superando os demais modelos avaliados.

O modelo DaViT também apresentou um desempenho notável, com uma acurácia geral de 0.7358, Kappa de 0.6362, QWK de 0.8647 e MAE de 0.3023. Esses resultados sugerem que o modelo DaViT é eficaz na extração de características relevantes em imagens médicas, embora tenha apresentado um desempenho ligeiramente inferior ao do modelo GCViT.

A [Tabela 8](#) mostra a acurácia dos modelos de RNCs e ViTs treinados para a classificação da OA de joelho usando a função de perda *crossentropy*. Em relação ao tempo de treinamento, é possível notar que o modelo mais rápido foi o ResNet-50, com um tempo de 11.29 segundos. Por outro lado, o modelo mais lento foi o DeiT, com um tempo de 79.5 segundos. Tais valores não necessariamente indicam que o modelo mais rápido é o pior, ou o contrário, mas é importante considerar o tempo de treinamento como um fator relevante ao escolher um modelo, especialmente se houver restrições de recursos computacionais. O tempo de treinamento mostrado varia, principalmente, com o número de épocas, já que modelos que levaram mais tempo são aqueles que tiveram a parada antecipada mais tarde, ou executaram as 30 épocas completas. ’

Quanto à acurácia geral (*overall*), todos os modelos apresentaram resultados razoavelmente bons, com valores variando de 0.6723 a 0.7319. Isso indica que todos os modelos foram capazes de aprender padrões relevantes para a classificação da OA de joelho.

No entanto, é importante notar que o modelo DenseNet-169 obteve a maior acurácia geral, com um valor de 0.7319. Isso sugere que arquiteturas de RNCs densamente conectadas podem ser muito eficazes na extração de características relevantes em imagens médicas como radiografias de joelho. Além disso, os modelos de conexões residuais (ResNet) também apresentaram resultados competitivos, com acurácias gerais variando de 0.7044 a 0.7248, onde o ResNet-50 obteve a maior acurácia dentre eles e com o menor tempo de treinamento, oferecendo um bom equilíbrio entre generalização do modelo e custo computacional.

Por outro lado, os modelos da família VGG (VGG-16 e VGG-19) apresentaram acurácias gerais mais baixas, variando de 0.6723 a 0.6851, o que sugere que essas arquiteturas mais simples podem não ser tão eficazes na extração de características complexas em radiografias de joelho. Embora fosse esperado que esses modelos tivessem desempenho inferior em relação aos modelos ResNet, devido à sua profundidade, os resultados indicam que esses modelos são capazes de aprender padrões relevantes e ter uma menor probabilidade de *overfitting*, como observado no tempo de treinamento do VGG-16, que foi maior que a maioria dos modelos justamente por não ter parada antecipada em virtude da queda do erro no conjunto de validação.

O GoogLeNet, com sua arquitetura Inception (versão 3), permitiu que o modelo tivesse uma acurácia geral de 0.7215, indicando que o modelo pode ser eficaz na extração de características relevantes e superar a maioria dos modelos de RNCs. Esse comportamento pode ser justificado pelo uso de uma técnica chamada de "bottleneck" ou "redução de dimensionalidade", que reduz a quantidade de parâmetros e a complexidade computacional do modelo, sem comprometer significativamente o desempenho.

Os modelos de transformers, por sua vez, apresentaram acurácias gerais variando de 0.6862 a 0.7215, indicando que essas arquiteturas podem ser eficazes, mas talvez não sejam tão eficientes quanto os modelos de RNCs. O modelo Swin Transformer obteve a maior acurácia geral entre os modelos de transformers, com um valor de 0.6977, sugerindo que a abordagem hierárquica de atenção pode ser eficaz na extração de características relevantes em radiografias de joelho.

Entretanto, é importante notar que a acurácia para a classe KL 1 foi baixa para todos os modelos, variando de 0.2562 a 0.4475. Isso indica que a classificação da OA de joelho no estágio 1 (duvidoso) pode ser mais desafiadora, possivelmente devido à semelhança visual com as classes adjacentes KL 0 e KL 2. Esse resultado pode ser observado na [Figura 14](#), que mostra a matriz de confusão do modelo ResNet-50. A classe KL 1 tem a menor acurácia dentre todas as classes, o que reflete o desafio na classificação dessa classe devido ao nível de detalhe ou até mesmo incoerência na rotulação das imagens do dataset.

Em resumo, os modelos ResNet-50 e DenseNet-169 se destacaram em termos de tempo de treinamento e acurácia geral, respectivamente. No entanto, é importante considerar as características de cada classe ao escolher um modelo, pois diferentes modelos

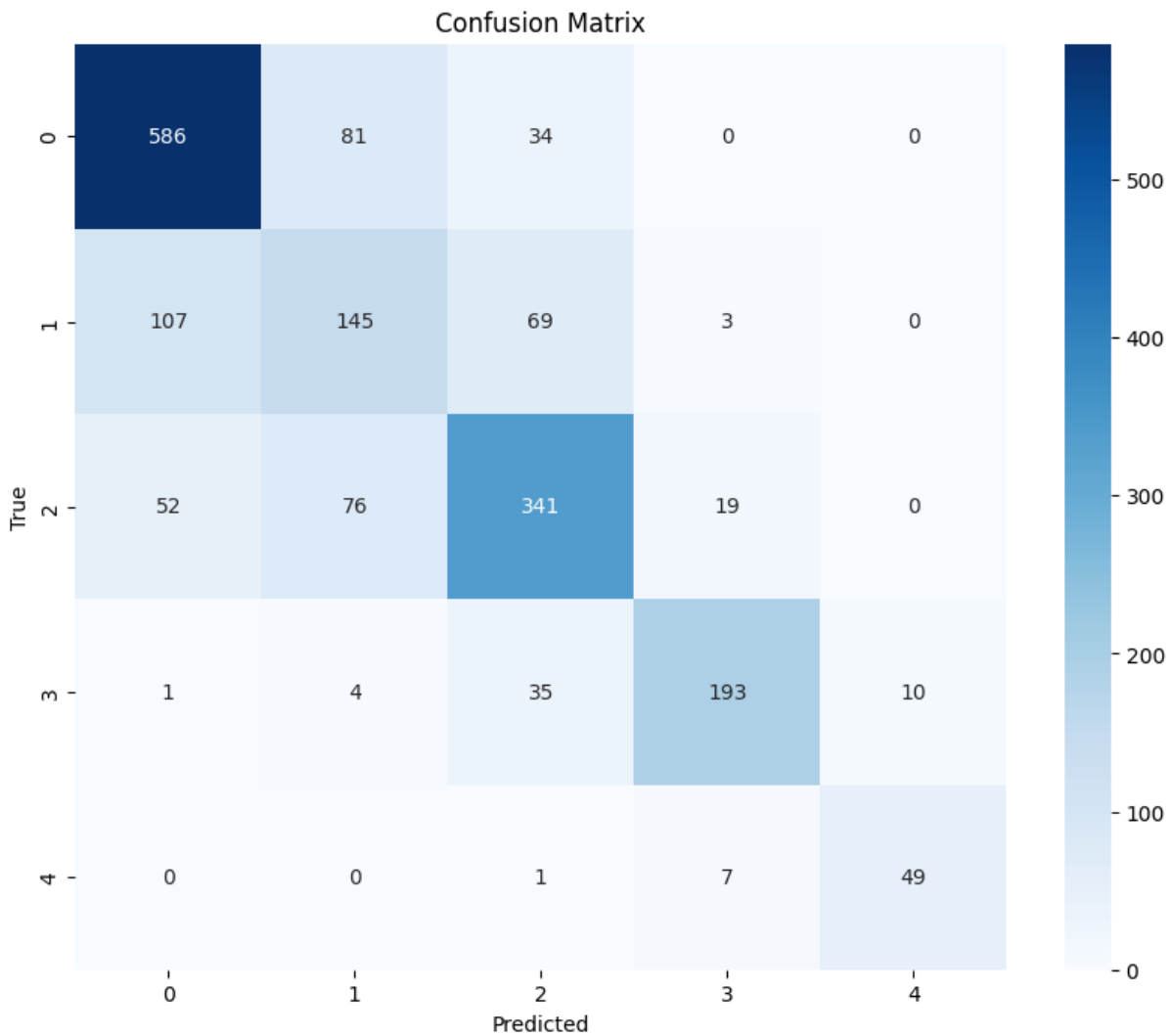


Figura 14 – Matriz de confusão do modelo ResNet-50.

podem ter desempenhos diferentes para cada classe.

A [Tabela 7](#) apresenta os resultados dos modelos de RNCs e ViTs treinados para a classificação da OA de joelho usando a função de perda CORN. Em relação ao tempo de treinamento, não houve uma mudança significativa comparado com a função de perda *crossentropy*. O modelo mais rápido foi, novamente, o ResNet-50, com um tempo de 10.32 segundos, enquanto o modelo mais lento foi o Swin Transformer, com um tempo de 35.58 segundos. Em relação à acurácia geral, os resultados variaram de 0.6546 a 0.7181, indicando que a função de perda CORN pode ser eficaz na classificação da OA de joelho, mas não necessariamente supera a função de perda *crossentropy*. Isso é justificado pelo fato de que a função de perda CORN é mais adequada quando o modelo faz previsões mais afastadas do rótulo real, o que não foi evidenciado ao observar as matrizes de confusão dos modelos.

No entanto, é importante notar que o modelo ResNet-50 obteve a maior acurácia geral, com um valor de 0.7181, superando os demais modelos, inclusive o modelo DenseNet-169, que obteve a maior acurácia geral com a função de perda *crossentropy*. Isso sugere

Modelo	Tempo	Overall	Classe KL				
			0	1	2	3	4
ResNet-34	14.93	0.6895	0.7518	0.5586	0.6107	0.8107	0.8246
ResNet-50	10.32	0.7181	0.796	0.5031	0.6824	0.823	0.8421
ResNet-101	16.17	0.6994	0.7418	0.4506	0.707	0.8519	0.8772
VGG-16	19.29	0.6762	0.7646	0.358	0.6824	0.7984	0.8246
VGG-19	24.05	0.6669	0.7974	0.3549	0.6066	0.7901	0.8246
DenseNet-121	10.62	0.6911	0.729	0.4444	0.7172	0.8272	0.8246
DenseNet-169	13.75	0.717	0.7874	0.5833	0.6393	0.8148	0.8596
Inception-v3	17.09	0.701	0.7932	0.5093	0.6639	0.7325	0.8421
ViT-B	36.97	0.6817	0.7447	0.4815	0.6393	0.8066	0.8772
DeiT	34.73	0.6602	0.7047	0.4877	0.6209	0.7984	0.8421
Swin	35.58	0.6546	0.7803	0.4658	0.6722	0.8395	0.7894

Tabela 7 – Desempenho dos modelos de RNCs e ViTs na classificação da OA de joelho usando a função da perda CORN.

que a função de perda CORN pode ser eficaz em arquiteturas de RNCs, especialmente aquelas com conexões residuais. Entretanto, o modelo DenseNet-169 foi quem obteve a maior acurácia para a classe KL 1, com um valor de 0.5833, que é a classe mais desafiadora de ser classificada, como observado anteriormente.

4.1 RESULTADOS E DISCUSSÃO

4.1.1 Visão Geral dos Resultados

Os experimentos envolveram avaliação de 18 arquiteturas (CNN e ViT), cada uma treinada com duas funções de perda: *cross entropy* e *Corn* (cumulative ordinal regression). As métricas principais consideradas foram acurácia, kappa de Cohen (coeficiente de concordância), weighted Quadratic Weighted Kappa (QWK) e Mean Absolute Error (MAE). A Tabela 8 resume os cinco modelos com melhor desempenho em termos de QWK, para ambas as funções de perda.

4.1.2 Discussão

Observa-se que o modelo **GCViT** apresentou o melhor desempenho global, atingindo QWK de 0.8725 com *cross entropy* e 0.8752 com *Corn*, além de MAE mínimo de 0.29399 e 0.29123, respectivamente;contentReference[oaicite:1]index=1. Logo em seguida, os modelos **DaViT** e **MaxViT__T** também se destacaram, com QWK acima de 0.86 e MAE abaixo de 0.32, evidenciando a eficácia de transformers especializados para tarefas de classificação ordinal de osteoartrite.

Entre as CNNs, as versões densas (*DenseNet169* e *DenseNet121*) obtiveram desempenhos notáveis, com QWK acima de 0.85 e acurácias próximas a 72% sob *cross entropy*,

mas ficaram abaixo dos ViTs em QWK e MAE;

Ainda, a função de perda *Corn* mostrou ligeira melhora em QWK para a maioria dos modelos de ViT, embora com tempo de treinamento tipicamente maior do que com *cross entropy* (por exemplo, GCViT: 3454 s vs. 2423 s);

Tabela 8 – Desempenho dos cinco melhores modelos (ordem por QWK) para as duas funções de perda.

Modelo	Cross Entropy					Corn				
	Acc.	Kappa	QWK	MAE	Tempo (s)	Acc.	Kappa	QWK	MAE	Tempo (s)
GCViT	0.7363	0.6372	0.8725	0.29399	3454.0	0.7347	0.6369	0.8752	0.29123	2423.0
DaViT	0.7358	0.6362	0.8647	0.30226	3045.0	0.7286	0.6288	0.8713	0.29950	2345.0
MaxViT_T	0.7220	0.6165	0.8616	0.31550	1489.6	0.7038	0.5934	0.8636	0.32267	1045.0
DenseNet169	0.7220	0.6158	0.8519	0.32488	888.2	0.7187	0.6143	0.8660	0.31219	1345.0
DenseNet121	0.7204	0.6110	0.8514	0.32708	1148.4	0.7071	0.6001	0.8592	0.32377	645.0

4.1.3 Principais Conclusões

- **GCViT** foi o modelo de melhor desempenho, com QWK máximo de 0.8752 (Corn) e MAE mínimo de 0.2912, indicando superior capacidade de modelar a natureza ordinal da escala KL.
- Transformers (GCViT, DaViT, MaxViT_T) superaram consistentemente as CNNs clássicas em QWK e MAE, embora demandem maior custo computacional.
- A função de perda *Corn* proporcionou ganhos moderados em QWK para ViTs, justificando seu uso em tarefas de classificação ordinal.
- Dentre as CNNs, *DenseNet169* e *DenseNet121* foram as mais competitivas, alcançando QWK acima de 0.85.

Referências

- A. Shane Anderson and Richard F. Loeser. Why is osteoarthritis an age-related disease?, 2010. ISSN 15216942. 5
- Fiocruz Brasília. Metade dos adultos brasileiros com obesidade em 20 anos. <https://www.fiocruzbrasil.br/quase-metade-dos-adultos-brasileiros-viverao-com-obesidade-em-20-anos/>, 2024. Acessado em: 10 de março de 2025. 9
- Hillary J. Braun and Garry E. Gold. Diagnosis of osteoarthritis: Imaging. *Bone*, 51, 2012. ISSN 87563282. doi: 10.1016/j.bone.2011.11.019. 9
- Pingjun Chen. Knee osteoarthritis dataset with severity grading. <https://www.kaggle.com/datasets/shashwatwork/knee-osteoarthritis-dataset-with-severity>, 2018. Acessado em: 29 de setembro de 2024. 23
- Alice Courties, Inès Kouki, Nadine Soliman, Sylvain Mathieu, and Jérémie Sellam. Osteoarthritis year in review 2024: Epidemiology and therapy. *Osteoarthritis and Cartilage*, 32 (11):1397–1404, 2024. ISSN 1063-4584. doi: <https://doi.org/10.1016/j.joca.2024.07.014>. URL <https://www.sciencedirect.com/science/article/pii/S1063458424013207>. 1, 8
- Ling Dai, Liang Wu, Huating Li, Chun Cai, Qiang Wu, Hongyu Kong, Ruhan Liu, Xiangning Wang, Xuhong Hou, Yuexing Liu, Xiaoxue Long, Yang Wen, Lina Lu, Yaxin Shen, Yan Chen, Dinggang Shen, Xiaokang Yang, Haidong Zou, Bin Sheng, and Weiping Jia. A deep learning system for detecting diabetic retinopathy across the disease spectrum. *Nature Communications*, 12, 2021. ISSN 20411723. doi: 10.1038/s41467-021-23458-5. 1
- François Desmeules, Clermont E. Dionne, Étienne Belzile, Renée Bourbonnais, and Pierre Frémont. Waiting for total knee replacement surgery: Factors associated with pain, stiffness, function and quality of life. *BMC Musculoskeletal Disorders*, 10, 2009. ISSN 14712474. doi: 10.1186/1471-2474-10-52. 7
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR 2021 - 9th International Conference on Learning Representations*, 2021. 2, 32

- Shiv Ram Dubey, Satish Kumar Singh, and Bidyut Baran Chaudhuri. Activation functions in deep learning: A comprehensive survey and benchmark, 2022. ISSN 18728286. 13, 14, 15
- Andre Esteva, Katherine Chou, Serena Yeung, Nikhil Naik, Ali Madani, Ali Mottaghi, Yun Liu, Eric Topol, Jeff Dean, and Richard Socher. Deep learning-enabled medical computer vision, 2021. ISSN 23986352. 18
- B. A. Ferrel. Pain management in elderly people, 1992. ISSN 10172572. 7
- Mary B. Goldring and Kenneth B. Marcu. Cartilage homeostasis in health and rheumatic diseases, 2009. ISSN 14786354. 6
- Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras and TensorFlow: concepts, tools, and techniques to build intelligent systems*. 2019. 5, 13, 14, 16, 17
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2016-December, 2016. doi: 10.1109/CVPR.2016.90. 27
- Geoffrey E. Hinton, Simon Osindero, and Yee Whye Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18, 2006. ISSN 08997667. doi: 10.1162/neco.2006.18.7.1527. 10
- Thomas J. Hoogeboom, Alfons A. den Broeder, Rob A. de Bie, and Cornelia H.M. Van Den Ende. Longitudinal impact of joint pain comorbidity on quality of life and activity levels in knee osteoarthritis: Data from the osteoarthritis initiative. *Rheumatology (United Kingdom)*, 52, 2013. ISSN 14620324. doi: 10.1093/rheumatology/kes314. 7
- Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, volume 2017-January, 2017. doi: 10.1109/CVPR.2017.243. 28
- Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to Statistical Learning*, volume 7. 2000. doi: 10.1007/978-1-4614-7138-7. 5, 11, 12
- Mohd Javaid, Abid Haleem, Ravi Pratap Singh, and Mumtaz Ahmed. Computer vision to enhance healthcare domain: An overview of features, implementation, and opportunities. *Intelligent Pharmacy*, 2(6):792–803, 2024. ISSN 2949-866X. doi: <https://doi.org/10.1016/j.ipha.2024.05.007>. URL <https://www.sciencedirect.com/science/article/pii/S2949866X24000662>. 20, 21

- Takashi Kanamoto, Tatsuo Mae, Teruki Yokoyama, Hiroyuki Tanaka, Kosuke Ebina, and Ken Nakata. Significance and definition of early knee osteoarthritis, 2020. ISSN 24156809. [1](#), [5](#)
- S. Kapetanakis. Evaluation of improvement in quality of life and physical activity after total knee arthroplasty in greek elderly women. *The Open Orthopaedics Journal*, 5, 2011. ISSN 18743250. doi: 10.2174/1874325001105010343. [8](#)
- Marcio Massao Kawano, Ivan Luis Andrade Araújo, Martha Cavalcante Castro, and Marcos Almeida Matos. Assessment of quality of life in patients with knee osteoarthritis. *Acta Ortopedica Brasileira*, 23, 2015. ISSN 14137852. doi: 10.1590/1413-785220152306150596. [8](#)
- J. H. KELLGREN and J. S. LAWRENCE. Radiological assessment of osteo-arthritis. *Annals of the rheumatic diseases*, 16, 1957. ISSN 00034967. doi: 10.1136/ard.16.4.494. [1](#), [9](#)
- Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015. [17](#)
- V. B. Kraus, F. J. Blanco, M. Englund, M. A. Karsdal, and L. S. Lohmander. Call for standardized definitions of osteoarthritis and risk stratification for clinical trials and clinical use, 2015. ISSN 15229653. [1](#), [9](#)
- Johannes Lederer. Activation functions in artificial neural networks: A systematic overview. *arXiv preprint arXiv:2101.09957*, 2021. [14](#)
- Kevin Leung, Bofei Zhang, Jimin Tan, Yiqiu Shen, Krzysztof J. Geras, James S. Babb, Kyunghyun Cho, Gregory Chang, and Cem M. Deniz. Prediction of total knee replacement and diagnosis of osteoarthritis by using deep learning on knee radiographs: Data from the osteoarthritis initiative. *Radiology*, 296, 2020. ISSN 15271315. doi: 10.1148/radiol.2020192091. [28](#), [30](#)
- Timothy P. Lillicrap, Adam Santoro, Luke Marris, Colin J. Akerman, and Geoffrey Hinton. Backpropagation and the brain. *Nature Reviews Neuroscience*, 21, 2020. ISSN 14710048. doi: 10.1038/s41583-020-0277-3. [5](#), [16](#), [17](#)
- Da Hon Lin, Chien Ho Janice Lin, Yeong Fwu Lin, and Mei Hwa Jan. Efficacy of 2 non-weight-bearing interventions, proprioception training versus strength training, for patients with knee osteoarthritis: A randomized clinical trial. *Journal of Orthopaedic and Sports Physical Therapy*, 39, 2009. ISSN 01906011. doi: 10.2519/jospt.2009.2923. [1](#)

- Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE International Conference on Computer Vision*, 2021. doi: 10.1109/ICCV48922.2021.00986. 33
- Richard F. Loeser, Steven R. Goldring, Carla R. Scanzello, and Mary B. Goldring. Osteoarthritis: A disease of the joint as an organ, 2012. ISSN 00043591. 5, 6, 7
- Andrew Lohn and Micah Musser. Ai and compute. *Blog Open AI*, 2022. 35
- Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *in ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013. 15
- Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5, 1943. ISSN 00074985. doi: 10.1007/BF02478259. 10
- Abdul Sami Mohammed, Ahmed Abul Hasanaath, Ghazanfar Latif, and Abul Bashar. Knee osteoarthritis detection and severity classification using residual neural networks on preprocessed x-ray images. *Diagnostics*, 13, 2023. ISSN 20754418. doi: 10.3390/diagnostics13081380. 1
- Muhammad Mujahid, Furqan Rustam, Roberto Álvarez, Juan Luis Vidal Mazón, Isabel de la Torre Díez, and Imran Ashraf. Pneumonia classification from x-ray images with inception-v3 and convolutional neural network. *Diagnostics*, 12, 2022. ISSN 20754418. doi: 10.3390/diagnostics12051280. 32
- Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML 2010 - Proceedings, 27th International Conference on Machine Learning*, 2010. 15
- Chigozie Nwankpa, Winifred Ijomah, Anthony Gachagan, and Stephen Marshall. Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*, 2018. 14, 15, 16
- National Institutes of Health. Osteoarthritis initiative. <https://www.niams.nih.gov/grants-funding/funded-research/osteoarthritis-initiative>, 2024. Acessado em: 17 de julho de 2024. 23
- World Health Organization. Whoqol: Measuring quality of life. <https://www.who.int/tools/whoqol>, 2012. Acessado em: 08 de março de 2025. 7
- Daniel Moreira PACCA, Gustavo Constantino DE-CAMPOS, Alessandro Rozin ZORZI, Elinton Adami CHAIM, and João Batista DE-MIRANDA. Prevalência de dor articular

- e osteoartrite na população obesa brasileira. *ABCD. Arquivos Brasileiros de Cirurgia Digestiva (São Paulo)*, 31, 2018. ISSN 2317-6326. doi: 10.1590/0102-672020180001e1344. 1, 5
- F. Pessler, L. Dai, C. Diaz-Torne, C. Gomez-Vaquero, M. E. Paessler, D. H. Zheng, E. Einhorn, U. Range, C. Scanzello, and H. R. Schumacher. The synovitis of "non-inflammatory" orthopaedic arthropathies: A quantitative histological and immunohistochemical analysis. *Annals of the Rheumatic Diseases*, 67, 2008. ISSN 00034967. doi: 10.1136/ard.2008.087775. 6
- F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65, 1958. ISSN 0033295X. doi: 10.1037/h0042519. 12
- Sebastian Ruder. An overview optimization gradients. *arXiv preprint arXiv:1609.04747*, 2017. ISSN 0006341X. 17
- Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115, 2015. ISSN 15731405. doi: 10.1007/s11263-015-0816-y. 30, 32
- Deepak Saini, Ashima Khosla, Trilok Chand, Devendra K. Chouhan, and Mahesh Prakash. Automated knee osteoarthritis severity classification using three-stage preprocessing method and vgg16 architecture. *International Journal of Imaging Systems and Technology*, 33, 2023. ISSN 10981098. doi: 10.1002/ima.22845. 27
- André Cabral Sardim, Rodrigo Paschoal Prado, and Carlos Eduardo Pinfieldi. Efeito da fotobiomodulação associada a exercícios na dor e na funcionalidade de pacientes com osteoartrite de joelho: estudo-piloto. *Fisioterapia e Pesquisa*, 27, 2020. ISSN 1809-2950. doi: 10.1590/1809-2950/18020027022020. 1
- Iqbal H. Sarker. Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions, 2021. ISSN 26618907. 13, 16
- Aarush Saxena. An introduction to convolutional neural networks. *International Journal for Research in Applied Science and Engineering Technology*, 10, 2022. doi: 10.22214/ijraset.2022.47789. 5, 11
- Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization*, 17, 2016. ISSN 00418781. 36

- Fahad Shamshad, Salman Khan, Syed Waqas Zamir, Muhammad Haris Khan, Munawar Hayat, Fahad Shahbaz Khan, and Huazhu Fu. Transformers in medical imaging: A survey, 2023. ISSN 13618423. 2, 21
- Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015. 26
- Chiranjibi Sitaula and Mohammad Belayet Hossain. Attention-based vgg-16 model for covid-19 chest x-ray image classification. *Applied Intelligence*, 51, 2021. ISSN 15737497. doi: 10.1007/s10489-020-02055-x. 27
- Matthew G. Snider, Steven J. MacDonald, and Ralph Pototschnik. Waiting times and patient perspectives for total hip and knee arthroplasty in rural and urban ontario, 2005. ISSN 0008428X. 8
- Vladislav Sovrasov. ptflops: a flops counting tool for neural networks in pytorch framework, 2018-2024. URL <https://github.com/sovrasov/flops-counter.pytorch>. 36
- Tim D. Spector and Alex J. MacGregor. Risk factors for osteoarthritis: Genetics. *Osteoarthritis and Cartilage*, 12, 2004. ISSN 10634584. doi: 10.1016/j.joca.2003.09.005. 5
- Serap Tomruk Sutbeyaz, Nebahat Sezer, Belma F. Koseoglu, Faruk Ibrahimoglu, and Demet Tekin. Influence of knee osteoarthritis on exercise capacity and quality of life in obese adults. *Obesity*, 15, 2007. ISSN 19307381. doi: 10.1038/oby.2007.246. 8
- Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 07-12-June-2015, 2015. doi: 10.1109/CVPR.2015.7298594. 30
- Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2016-December, 2016. doi: 10.1109/CVPR.2016.308. 31
- Tayyaba Tariq, Zobia Suhail, and Zubair Nawaz. Knee osteoarthritis detection and classification using x-rays. *IEEE Access*, 11, 2023. ISSN 21693536. doi: 10.1109/ACCESS.2023.3276810. 21
- Hugging Face Team. What is computer vision?, 2024. URL <https://huggingface.co/learn/computer-vision-course/unit1/chapter1/definition>. Accessed: 2024-12-15. 5, 18

- Ruchita Tekade and K. Rajeswari. Lung cancer detection and classification using deep learning. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–5, 2018. doi: 10.1109/ICCUBEA.2018.8697352. 1
- Ashitosh Tilve, Shrameet Nayak, Saurabh Vernekar, Dhanashri Turi, Pratiksha R. Shetgaonkar, and Shailendra Aswale. Pneumonia detection using deep learning approaches. In *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, pages 1–8, 2020. doi: 10.1109/ic-ETITE47903.2020.152. 1
- Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers and distillation through attention. In *Proceedings of Machine Learning Research*, volume 139, 2021. 33
- Matilde Tschon, Deyanira Contartese, Stefania Pagani, Veronica Borsari, and Milena Fini. Gender and sex are key determinants in osteoarthritis not only confounding variables. a systematic review of clinical data, 2021. ISSN 20770383. 5
- Peter M. van der Kraan and Wim B. van den Berg. Osteophytes: relevance and biology, 2007. ISSN 10634584. 6
- Haoran Wang, Qiuye Jin, Shiman Li, Siyu Liu, Manning Wang, and Zhijian Song. A comprehensive survey on deep active learning in medical image analysis. *Medical Image Analysis*, 95:103201, 2024. ISSN 1361-8415. doi: <https://doi.org/10.1016/j.media.2024.103201>. URL <https://www.sciencedirect.com/science/article/pii/S1361841524001269>. 1
- John E. Ware and Cathy Donald Sherbourne. The mos 36-item short-form health survey (sf-36): I. conceptual framework and item selection. *Medical Care*, 30, 1992. ISSN 15371948. doi: 10.1097/00005650-199206000-00002. 7
- Érika Rodrigues Senna, Ana Letícia P. De Barros, Edvânia O. Silva, Isabella F. Costa, Leonardo Victor B. Pereira, Rozana Mesquita Ciconelli, and Marcos Bosi Ferraz. Prevalence of rheumatic diseases in brazil: A study using the copcord approach. *Journal of Rheumatology*, 31, 2004. ISSN 0315162X. 9