

Universidade Federal do Piauí
Centro de Ciências da Natureza
Programa de Pós-Graduação em Ciência da Computação

WellNet: Uma Rede Híbrida ViT-CNN para a Classificação de Rachaduras e Fissuras em Imagens de Estruturas de Concreto

Wellington Viana da Silva

Número de Ordem PPGCC: M001

Teresina-PI, Julho de 2025

Wellington Viana da Silva

WellNet: Uma Rede Híbrida ViT-CNN para a Classificação de Rachaduras e Fissuras em Imagens de Estruturas de Concreto

Defesa de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFPI (área de concentração: Visão Computacional), como parte dos requisitos necessários para a obtenção do Título de Mestre em Ciência da Computação.

Universidade Federal do Piauí – UFPI

Centro de Ciências da Natureza

Programa de Pós-Graduação em Ciência da Computação

Orientador: Prof. Dr. Kelson Rômulo Teixeira Reis

Coorientador: Prof. Dr. André Castelo Branco Soares

Teresina-PI

Julho de 2025

FICHA CATALOGRÁFICA
Universidade Federal do Piauí
Sistema de Bibliotecas UFPI - SIBi/UFPI
Biblioteca Setorial do CCN

S586w Silva, Wellington Viana da.
WellNet: uma rede híbrida ViT-CNN para a classificação de rachaduras e fissuras em imagens de estrututas de concreto / Wellington Viana da Silva . -- 2025.
108 f.

Dissertação (Mestrado) - Universidade Federal do Piauí. Centro de Ciências da Natureza. Programa de Pós-Graduação em Ciências da Computação, Teresina, 2025.
"Orientador: Prof. Dr. Kelson Rômulo Teixeira Reis. Coorientador: Prof. Dr. André Castelo Branco Soares".

1. Redes Neurais Convolucionais. 2. Vision Transformers (ViT). 3. Construção Civil. 4. Rachaduras. I. Reis, Kelson Rômulo Teixeira. II. Soares, André Castelo Branco. II. Titulo.

CDD 006.32

Bibliotecária: Caryne Maria da Silva Gomes - CRB3/1461

Wellington Viana da Silva

WellNet: Uma Rede Híbrida ViT-CNN para a Classificação de Rachaduras e Fissuras em Imagens de Estruturas de Concreto

Defesa de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFPI (área de concentração: Visão Computacional), como parte dos requisitos necessários para a obtenção do Título de Mestre em Ciência da Computação.

gov.br Documento assinado digitalmente
KELSON ROMULO TEIXEIRA AIRES
Data: 15/07/2025 21:51:41-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Kelson Rômulo Teixeira Reis
Orientador

gov.br Documento assinado digitalmente
ANDRE CASTELO BRANCO SOARES
Data: 25/07/2025 12:10:15-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. André Castelo Branco Soares
Co-Orientador

gov.br Documento assinado digitalmente
IVAN SARAIVA SILVA
Data: 16/07/2025 14:54:06-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Ivan Saraiva Silva
Examinador Interno

gov.br Documento assinado digitalmente
VINICIUS PONTE MACHADO
Data: 23/07/2025 22:46:46-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Vinícius Ponte Machado
Examinador Interno

gov.br Documento assinado digitalmente
ANSELMO CARDOSO DE PAIVA
Data: 24/07/2025 07:44:04-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Anselmo Cardoso de Paiva
Examinador Externo

Teresina-PI
Julho de 2025

Aos meus pais Antonio Luis e Maria da Cruz, e minha esposa Ana Paz
por sempre estarem comigo em todos os momentos.

Agradecimentos

Agradeço a Deus por ser minha força e inspiração em todos os momentos dessa jornada.

Aos meus pais, Antonio Luis e Maria da Cruz, sou profundamente grato por acreditarem em mim e me apoiarem incondicionalmente em todas as etapas da minha vida. A paciência e o amor de vocês foram essenciais para que eu chegasse até aqui.

À minha esposa, Ana Paz, agradeço pela compreensão, carinho e incentivo constantes. Obrigado por estar ao meu lado nos momentos mais difíceis e por celebrar comigo cada pequena vitória.

Agradeço aos meus irmãos, especialmente à Karina e ao meu cunhado Victor, por suas valiosas sugestões para a revisão da escrita desta pesquisa e pelo apoio constante.

Aos meus amigos, em especial ao Martony e ao José de Arimatéia, expresso minha mais profunda gratidão por dedicarem seu tempo e esforço na revisão deste trabalho, oferecendo sugestões valiosas que contribuíram significativamente para a melhoria e qualidade do conteúdo apresentado.

Por fim, agradeço ao meu orientador, Kelson Rômulo, e ao co-orientador André Soares, por todos os conselhos, paciência e ajuda ao longo desse período. A orientação de vocês foi fundamental para a realização deste trabalho.

"A ciência é um processo de transformação contínua; o que hoje é desconhecido, amanhã pode ser desvendado pela perseverança e pelo desejo de descobrir." (Autor Desconhecido)

Resumo

A classificação de rachaduras e fissuras em superfícies de concreto é uma tarefa essencial para a manutenção preventiva de estruturas civis, pois quando não detectadas a tempo, podem comprometer a integridade estrutural e resultar em custos elevados de reparo. Neste contexto, esta proposta de pesquisa apresenta a WellNet, uma rede neural híbrida que se propõe como uma alternativa entre as abordagens para classificação de rachaduras e fissuras em imagens de estruturas de concretos. A WellNet combina as capacidades das Redes Neurais Convolucionais (CNN), conhecidas por sua eficiência na extração de características locais, com a abordagem dos *Vision Transformers* (ViT), que captura informações contextuais globais. O modelo foi validado utilizando o conjunto de (Özgenel, 2019), com diferentes configurações de hiperparâmetros e técnicas de aumento de dados. Foram realizados experimentos com e sem aumento de dados, evidenciando que a WellNet apresenta resultados comparáveis a arquiteturas CNN e híbridas em métricas como acurácia de 99,70%, precisão de 99,50%, *recall* de 99,90% e F1-*score* de 99,70%. Além disso, o modelo se destaca pelo seu espaço em memória de (26.4 MB), tornando-o viável para uso em sistemas embarcados e dispositivos móveis, facilitando inspeções automatizadas em fachadas de edificações e locais de difícil acesso.

Palavras-chave: classificação de rachaduras e fissuras ; redes neurais convolucionais; transformadores de visão; aprendizado por transferência; CNN; ViT.

Abstract

The classification of cracks and fissures in concrete surfaces is an essential task for preventive maintenance of civil structures, since, when not detected in time, they can compromise structural integrity and result in high repair costs. In this context, this research proposal presents WellNet, a hybrid neural network that proposes itself as an alternative among approaches for crack and fissure classification in images of concrete structures. WellNet combines the capabilities of Convolutional Neural Networks (CNN), known for their efficiency in local feature extraction, with the approach of *Vision Transformers* (ViT), which captures global contextual information. The model was validated using the dataset from (Özgenel, 2019), with different hyperparameter configurations and data augmentation techniques. Experiments were conducted with and without data augmentation, showing that WellNet presents results comparable to CNN and hybrid architectures in metrics such as accuracy of 99.70%, precision of 99.50%, *recall* of 99.90%, and F1-*score* of 99.70%. Furthermore, the model stands out for its memory space of 26.4 MB, making it viable for use in embedded systems and mobile devices, facilitating automated inspections in building facades and hard-to-reach locations.

Keywords: crack and fissure classification; convolutional neural networks; vision transformers; transfer learning; CNN; ViT.

Lista de ilustrações

Figura 1 – Relação entre Inteligência Artificial, Visão Computacional e Processamento de Imagem. 11
Figura 2 – Relação entre Inteligência Artificial, Aprendizado de Máquina e Aprendizado Profundo. 13
Figura 3 – Fluxo do Processo de Transfer Learning. 17
Figura 4 – Exemplo: Imagens de rachaduras. 35
Figura 5 – Exemplo: Imagens de Não rachaduras. 35
Figura 6 – Fluxograma do Pré-Processamento. 36
Figura 7 – Processo da Validação Cruzada. 37
Figura 8 – Diagrama de Fluxo de Treinamento. 39
Figura 9 – Arquitetura da WellNet. 46
Figura 10 – Análise dos patches do *Vision Transformer*: (a) Imagem original mostrando uma rachadura vertical em superfície de concreto; (b) Norma L2 dos patches, onde valores mais altos (amarelo e verde) indicam maior magnitude de ativação, claramente delineando a trajetória da rachadura; (c) Mapa de ativação mostrando a intensidade de resposta do ViT, com forte ativação ao longo da rachadura; (d-f) Visualização dos canais 0, 48 e 96 dos *embeddings* dos patches, demonstrando diferentes padrões de codificação espacial. 55
Figura 11 – Transição entre *Vision Transformer* e camadas convolucionais: (Superior) Distribuição de *features* para *Tokens* selecionados (0, 49, 98, 147) mostrando as primeiras 64 dimensões de cada token. O *Token* 49 apresenta pico pronunciado na *feature* 42, indicando especialização. (Inferior) Mapas de características da camada convolucional intermediária para os canais 0, 16, 32 e 48, demonstrando como os *embeddings* dos *Tokens* são reorganizados em representações espaciais 2D. 56
Figura 12 – Visualização completa das camadas convolucionais mostrando 16 canais uniformemente distribuídos de cada camada. Conv1 (superior): 64 canais totais, resolução 14×14. Conv2 (meio): 128 canais totais, resolução 7×7. Conv3 (inferior): 256 canais totais, resolução 3×3. A progressão mostra claramente o aumento na abstração e redução na resolução espacial. . . 59
Figura 13 – Resumo das camadas convolucionais mostrando os 6 canais mais ativos de cada camada. Conv1: Canais 38, 39, 48, 9, 49, 24 apresentam as maiores ativações médias. Conv2: Canais 68, 126, 26, 74, 107, 99 demonstram forte resposta. Conv3: Canais 185, 77, 215, 81, 99, 146 codificam as representações finais mais discriminativas. 61

Figura 14 – Análise estatística detalhada: (Esquerda) Histogramas de distribuição de ativações mostrando assimetria negativa em todas as camadas. (Centro) Mapas de ativação média revelando padrões espaciais persistentes. (Direita) Gráficos de barras dos 10 canais mais ativos, demonstrando a hierarquia de importância dos filtros. 62
Figura 15 – Aplicação Web da WellNet. 72
Figura 16 – Resultados de Classificação da Aplicação Web da WellNet 74
Figura 17 – Exemplos Adicionais de Resultados de Classificação da Aplicação Web da WellNet 75
Figura 18 – Matriz de confusão do Experimento 1. 84
Figura 19 – Visão geral das métricas de desempenho do Experimento 1. 85
Figura 20 – Curva ROC do Experimento 1. 86
Figura 21 – Matriz de confusão do Experimento 2. 87
Figura 22 – Visão geral das métricas de desempenho do Experimento 2. 88
Figura 23 – Curva ROC do Experimento 2. 89
Figura 24 – Matriz de confusão do Experimento 3. 89
Figura 25 – Visão geral das métricas de desempenho do Experimento 3. 90
Figura 26 – Curva ROC do Experimento 3. 91
Figura 27 – Matriz de confusão do Experimento 4. 92
Figura 28 – Visão geral das métricas de desempenho do Experimento 4. 93
Figura 29 – Curva ROC do Experimento 4. 94

Lista de tabelas

Tabela 1 – Tabela Comparativa entre a WellNet e Outras Redes para Classificação, Detecção e Segmentação de Rachaduras 27
Tabela 2 – Comparação entre abordagens arquiteturais híbridas para detecção de rachaduras 30
Tabela 3 – Comparação entre abordagens arquiteturais diretas e decompostas . . . 49
Tabela 4 – Resumo das Camadas e Parâmetros da Arquitetura WellNet 54
Tabela 5 – Análise quantitativa detalhada das camadas convolucionais 58
Tabela 6 – Comparação de eficiência computacional: resultados experimentais . . . 67
Tabela 7 – Distribuição percentual de parâmetros na WellNet 69
Tabela 8 – Resumo dos Experimentos 81
Tabela 9 – Resultados Consolidados dos Experimentos de Treinamento 82
Tabela 10 – Resultados com Dados Externos 83
Tabela 11 – Comparação dos Resultados da WellNet com Outros Modelos 96
Tabela 12 – Diferenças Absolutas entre WellNet e Outros Modelos (em pontos percentuais) 96
Tabela 13 – Posicionamento dos Modelos por Métrica Individual 96

Lista de abreviaturas e siglas

CNN - Convolutional Neural Networks

ViT - Vision Transformers

IA - Inteligência Artificial

AM - Aprendizado de Máquina

AP - Aprendizado Profundo

Sumário

1 INTRODUÇÃO 1
1.1 Motivação 4
1.2 Objetivos 4
1.3 Contribuições 4
1.4 Considerações Finais 5

I PREPARAÇÃO DA PESQUISA 7

2 REFERENCIAL TEÓRICO 9
2.1 Definições Normativas: Fissuras e Rachaduras 9
2.2 Visão Computacional 10
2.3 Deep Learning - (Aprendizado Profundo) 12
2.3.1 Relação entre Inteligência Artificial, Aprendizado de Máquina e Aprendizado Profundo 12
2.3.2 Avanços das Arquiteturas de *Deep Learning* 14
2.4 Transfer Learning - (Aprendizado por Transferência) 15
2.4.1 Aplicação do Transfer Learning 16
2.5 Vision Transformers (ViT) 18

3 ESTADO DA ARTE 23
3.1 Trabalhos Relacionados à Classificação de Rachaduras 23
3.1.1 Fundamentos das Abordagens Convolucionais 23
3.1.2 Evolução para Arquiteturas Transformer 23
3.1.3 Integração de Arquiteturas Híbridas 24
3.1.4 Otimização Computacional e Aplicações Práticas 24
3.1.5 Inovações Arquiteturais Recentes 25
3.1.6 Integração de Conhecimentos na WellNet 25
3.1.7 Direções Futuras e Limitações a Serem Exploradas 26
3.2 Comparação de Características com Trabalhos Relacionados 26
3.2.1 Diferenciais da WellNet 27
3.2.2 Características de Outras Abordagens 28
3.2.3 Contribuições e Potenciais de Aplicação 28
3.2.4 Abordagem ViT→CNN: Fundamentação Teórica e Comparação Arquitetural 29
3.2.4.1 Fundamentação Teórica da Arquitetura ViT→CNN 29
3.2.4.2 Comparação com Abordagens Arquiteturais Alternativas 30

3.2.4.3 Análise das Abordagens CNN→ViT: . . . 30
3.2.4.4 Vantagens Conceituais da Abordagem ViT→CNN: . . . 30
3.2.4.5 Limitações da Integração Híbrida: . . . 31

4 MATERIAIS E MÉTODOS . . . 33
4.1 Ferramentas, Bibliotecas e Métodos Utilizados . . . 33
4.1.1 Ferramentas e Bibliotecas Utilizadas . . . 33
4.2 Base de Dados . . . 34
4.2.1 Dataset (Özgenel, 2019) . . . 34
4.3 Aquisição e Pré-processamento dos Dados . . . 35
4.4 Divisão dos Dados e Validação Cruzada . . . 36
4.5 Treinamento e Configurações do Modelo . . . 37
4.6 Hiperparâmetros Utilizados . . . 39
4.6.1 Metodologia de Seleção . . . 39
4.6.2 Hiperparâmetros Selecionados . . . 40
4.6.3 Limitações da Metodologia Empírica . . . 40
4.7 *hardware* e Ferramentas Auxiliares . . . 41
4.8 Impactos da Arquitetura Híbrida . . . 41
4.9 Considerações Finais . . . 42

II PROPOSTA 43

5 SISTEMA PROPOSTO . . . 45
5.1 Primeira Parte do Sistema Proposto: A arquitetura . . . 45
5.2 Arquitetura WellNet . . . 46
5.2.1 Camada de Entrada . . . 47
5.2.2 Bloco *Vision Transformers*: *Backbone* (Extrator de Características) . . . 47
5.2.3 Camada de Convolução Intermediária - Conversão . . . 48
5.2.4 Justificativa da necessidade Arquitetural da Camada Intermediária . . . 49
5.2.4.0.1 Análise Comparativa de Abordagens Arquiteturais . . . 49
5.2.4.0.2 Validação Experimental . . . 49
5.2.4.0.3 Análise de Eficiência Paramétrica . . . 49
5.2.4.0.4 Fundamentação Teórica da Separação de Responsabilidades . . . 50
5.2.4.0.5 Justificativa da Implementação via Conv2d . . . 50
5.2.5 Bloco de Camadas Convolucionais . . . 51
5.2.6 Bloco de Camadas Densas *(Fully Connected)* . . . 52
5.2.7 Parâmetros da Arquitetura WellNet . . . 54
5.3 Análise dos Mapas de Características da Arquitetura WellNet . . . 54
5.3.1 Extração de Características pelo Vision Transformer . . . 54

5.3.2 Transição ViT para Camadas Convolucionais . . . 56
5.3.3 Análise Hierárquica das Camadas Convolucionais . . . 58
5.3.4 Análise dos Canais Mais Discriminativos . . . 61
5.3.5 Análise Estatística Detalhada das Distribuições . . . 62
5.3.6 Eficiência Computacional da Arquitetura WellNet . . . 64
5.3.7 Implicações para Detecção de Rachaduras . . . 71
5.4 Segunda Parte do Sistema: O protótipo web . . . 72
5.5 Aplicação Web . . . 72
5.5.1 Validação e Desempenho da Inferência do Modelo . . . 75
5.5.2 Limitações da Aplicação Web da WellNet . . . 76
5.6 Considerações Finais . . . 77

III PARTE FINAL 79

6 RESULTADOS E DISCUSSÃO . . . 81
6.1 Análise dos Resultados dos Experimentos: Dados de Treinamento . 81
6.1.1 Resumo dos Experimentos . . . 81
6.2 Resultado do Modelo Treinado e Testado: Dados Externos . . . 83
6.2.1 Métricas dos Experimentos - Dados Externos . . . 84
6.3 Comparativo de Resultados com Trabalhos Relacionados . . . 94
6.3.1 Metodologia de Avaliação . . . 94
6.3.2 Métricas de Avaliação . . . 94
6.3.3 Resultados Comparativos . . . 95
6.3.4 Análise Quantitativa dos Resultados . . . 96
6.3.5 Análise de Ranking por Métrica . . . 96
6.3.6 Análise das Diferenças Observadas . . . 97
6.3.6.1 Comparação com CNN (DINIZ et al., 2023) . . . 97
6.3.6.2 Comparação com DeepCrack (ZOU et al., 2019) . . . 97
6.3.6.3 Comparação com AlexNet Pré-treinada (KARMOKAR et al., 2020) . . . 97
6.3.6.4 Comparação com ViT+Detector Textura (SHAHIN et al., 2024) . . . 97
6.3.7 Limitações da Análise Comparativa . . . 97
6.3.8 Análise de Perfis de Desempenho . . . 98
6.3.9 Posicionamento da WellNet no Contexto Atual . . . 98
6.3.10 Considerações sobre Aplicabilidade . . . 99
6.4 Considerações Finais . . . 99

7 CONCLUSÕES E TRABALHOS FUTUROS . . . 101
7.1 Conclusões . . . 101
7.2 Trabalhos Futuros . . . 102

REFERÊNCIAS 105

1 Introdução

A classificação de rachaduras e fissuras em superfícies de concreto é uma tarefa essencial para a manutenção preventiva de estruturas de concreto, garantindo a segurança e a longevidade dessas construções (SILVA et al., 2024). Rachaduras e fissuras em estruturas civis, como edifícios, pontes e estradas, representam um grande risco, pois se não detectadas a tempo, podem comprometer a integridade estrutural e levar a custos elevados de manutenção e reparo. Dessa forma, a inspeção eficiente dessas rachaduras e fissuras é crucial para prevenir danos maiores e dispendiosos.

A análise automatizada de rachaduras e fissuras em estruturas civis, incluindo edificações de concreto, alvenaria, estruturas históricas e pavimentação, pode ser categorizada em três abordagens fundamentais distintas: classificação, detecção e segmentação. A **classificação de rachaduras e fissuras** consiste no processo de determinar se uma imagem contém ou não rachaduras e fissuras, tratando-se de uma tarefa de classificação binária que categoriza as imagens em classes como "com rachadura" e "sem rachadura", aplicável tanto em estruturas de concreto quanto em muros de alvenaria (YE et al., 2024; AL-SMADI et al., 2022). A **detecção de rachaduras e fissuras** vai além da simples classificação, localizando espacialmente as rachaduras e fissuras na imagem através de caixas delimitadoras (*bounding boxes*), permitindo identificar não apenas a presença, mas também a localização aproximada das rachaduras e fissuras em diferentes tipos de estruturas civis, desde pontes de concreto até paredes de tijolos históricos (CHEN; LIU; CHEN, 2022; LOVERDOS; SARHOSIS, 2022).

Por sua vez, a **segmentação de rachaduras e fissuras** representa a abordagem mais precisa, realizando classificação pixel-a-pixel para delinear exatamente os contornos das rachaduras e fissuras, fornecendo informações detalhadas sobre a geometria, densidade e caminho das rachaduras e fissuras tanto em superfícies de concreto quanto em estruturas de alvenaria complexas (DAIS et al., 2021; CHEN; LIU; CHEN, 2022). Esta última abordagem é particularmente valiosa para quantificação precisa de danos estruturais em diversas tipologias de superfícies, uma vez que permite a medição exata da largura, comprimento e área das rachaduras e fissuras para avaliação da integridade estrutural de edifícios histórico, estruturas modernas de concreto e de alvenaria.

A identificação precoce de rachaduras e fissuras em estruturas civis é fundamental para garantir a segurança pública e minimizar custos financeiros. rachaduras e fissuras não detectadas podem evoluir para falhas estruturais graves, comprometendo a estabilidade de pontes, edifícios e outras infraestruturas, o que representa um risco direto à vida humana e à segurança de comunidades inteiras. Além disso, o custo de manutenção preventiva

e reparos localizados é consideravelmente menor do que o de reparos emergenciais ou reconstruções após falhas. Portanto, a identificação e o tratamento precoces de rachaduras e fissuras são essenciais para a preservação das infraestruturas e para a economia de recursos públicos e privados.

O concreto é um dos materiais mais utilizados na construção civil, sendo preferido principalmente devido ao seu custo acessível e alta versatilidade. Com o aumento das demandas neste segmento no Brasil e no mundo, observou-se uma intensificação na execução de obras, o que trouxe desafios adicionais na garantia da qualidade das estruturas. Esses desafios podem resultar no surgimento de patologias em edificações, como trincas e rachaduras e fissuras, que se não tratadas adequadamente, podem comprometer a integridade das construções. A identificação e correção dessas manifestações anomalias estruturais são fundamentais para garantir a durabilidade e segurança das edificações (SILVA; LAURSEN, 2022).

A ocorrência de rachaduras e fissuras e outras manifestações danos estruturais em edificações é um fenômeno comum, especialmente quando não se realiza manutenção preventiva de maneira adequada. No estudo conduzido por (SILVA; PINHEIRO, 2022), foram identificadas diversas patologias em residências, como rachaduras e fissuras, destacamento de revestimentos cerâmicos, manchas de umidade e crescimento de fungos. Essas falhas estão associadas a fatores como falta de manutenção periódica e questões relacionadas ao planejamento e qualidade dos materiais utilizados.

Historicamente, a inspeção de rachaduras e fissuras tem sido feita manualmente, o que exige muito tempo, mão de obra especializada e está sujeita a erros humanos. Além do mais, esse processo envolve riscos significativos à segurança dos trabalhadores, especialmente em locais de difícil acesso, como fachadas de prédios e pontes. O uso de drones equipados com câmeras surge como uma alternativa promissora, permitindo o monitoramento visual das estruturas sem a necessidade de expor os trabalhadores a condições perigosas (KARMOKAR et al., 2020).

Apesar das vantagens trazidas pelo uso de tecnologias de captura de imagem, a classificação e detecção automática de rachaduras e fissuras em condições reais de campo é desafiadora devido às variações nas condições de iluminação, presença de sombras, texturas irregulares e outros fatores ambientais. Segundo (PAL et al., 2021), as técnicas de aprendizado profundo frequentemente enfrentam dificuldades quando aplicadas a imagens de concreto capturadas em condições adversas, como sombras e variações de iluminação. Muitos modelos treinados em ambientes ideais não apresentam o mesmo desempenho em situações reais, onde as condições são muito mais variáveis. Essas limitações evidenciam a necessidade de desenvolver modelos mais robustos, capazes de lidar com a complexidade das condições de campo.

As Redes Neurais Convolucionais (CNNs, do inglês Convolutional Neural Networks)

são um tipo específico de rede neural projetada para lidar com dados com formato de grade, como imagens. Inspiradas pelos processos de reconhecimento visual presentes em sistemas biológicos, as CNNs foram inicialmente desenvolvidas por Yann LeCun e colaboradores nos anos 1990, como descrito na LeNet (LECUN et al., 1998). As CNNs utilizam convoluções para extrair características locais da imagem, como bordas, texturas e formas, em diferentes níveis de abstração, tornando-as amplamente aplicáveis em tarefas de visão computacional, como classificação de objetos e detecção de anomalias.

As CNNs têm sido amplamente utilizadas para identificar padrões em imagens, como rachaduras e fissuras em superfícies de concreto. Entretanto, essas redes têm dificuldades para identificar relações entre partes distantes de uma mesma imagem, uma vez que a operação de convolução, caracterizada por seu campo receptivo fixo, é eficaz na identificação de padrões locais, mas é limitada na captura de características contextuais globais ou dependências de longo alcance (AL-MAQTARI et al., 2024). Nesse cenário, os *Vision Transforms* (ViT) surgem como uma abordagem complementar e promissora. Diferentemente das CNNs, os ViTs utilizam mecanismos de atenção que permitem aprender interdependências espaciais globais, essenciais em tarefas de detecção de anomalias e padrões complexos (DOSOVITSKIY et al., 2021).

Diante dos desafios na classificação e detecção de patologias em estruturas civis e dos avanços tecnológicos, a **WellNet** foi concebida. O nome WellNet é uma junção das iniciais do autor **Well** (de Wellington) com **Networks**, refletindo o objetivo de criar uma rede neural eficiente para a classificação de rachaduras e fissuras. A WellNet combina as vantagens das camadas convolucionais, eficazes na extração de padrões locais, e dos *Vision Transformers*, que capturam informações globais da imagem. Essa integração visa criar uma solução eficaz para a classificação de rachaduras e fissuras, sendo uma alternativa promissora e também aplicável a outras tarefas de visão computacional. Esta pesquisa apresenta uma comparação da WellNet com outras arquiteturas da literatura, avaliando seu desempenho em termos de acurácia, precisão, *recall* e *F1-score*.

Portanto, o diferencial da WellNet, reside na sua arquitetura otimizada e não profunda, que integra mecanismos de atenção dos *Vision Transformers* com camadas convolucionais, através de módulos específicos para extração de características para classificação de rachaduras e fissuras. A arquitetura enxuta proposta minimiza a complexidade computacional, resultando em tempos de inferência reduzidos e viabilizando sua implementação em dispositivos de recursos limitados, sem comprometer a eficácia na captura de padrões locais e globais essenciais para a classificação precisa de rachaduras e fissuras.

1.1 Motivação

Apesar dos avanços em técnicas automatizadas de visão computacional, persistem lacunas importantes na literatura atual. As CNNs tradicionais apresentam limitações para capturar dependências de longo alcance devido ao campo receptivo fixo (AL-MAQTARI et al., 2024), enquanto os *Vision Transformers* carecem de processamento hierárquico de características locais e demandam recursos computacionais elevados.

Existe uma carência de abordagens híbridas ViT→ CNN que integrem sistematicamente as vantagens complementares de ambas as arquiteturas, especificamente otimizadas para classificação de rachaduras e fissuras. As propostas existentes não abordam adequadamente o equilíbrio entre precisão e eficiência computacional necessário para aplicações práticas.

Neste contexto, a WellNet surge como uma solução alternativa que visa combinar o poder das redes neurais convolucionais com a eficiência dos *Vision Transformers*, proporcionando um modelo preditivo capaz de realizar a classificação de maneira precisa, rápida e eficiente.

1.2 Objetivos

O objetivo geral do estudo é propor e avaliar a abordagem ViT→CNN de uma rede neural, denominada WellNet, que integra a arquitetura *Vision Transformers* (ViT) com *Convolutional Neural Networks* (CNNs) para aprimorar a classificação automatizada de imagens de rachaduras e fissuras em superfícies de concreto.

Os objetivos específicos deste estudo são:

- Analisar o desempenho da rede proposta em termos de acurácia, precisão, *recall* e *F1-score*, comparando os resultados com outras soluções existentes, na literatura, para validação da eficácia do modelo.

- Avaliar a eficiência da arquitetura WellNet, considerando sua viabilidade para implementação em ambientes de recursos computacionais limitados, como dispositivos móveis ou sistemas embarcados.

1.3 Contribuições

A WellNet oferece as seguintes contribuições para classificação automatizada de rachaduras e fissuras:

- **Arquitetura híbrida otimizada**: Integração sistemática entre *Vision Transformers* e camadas convolucionais combinando eficiência computacional com capacidade de captura de padrões globais.
- **Viabilidade para implantação**: Arquitetura compacta adequada para sistemas embarcados e dispositivos móveis, possibilitando inspeções automatizadas em locais de difícil acesso.
- **Sistema web integrado**: Aplicação web, prática para uso por profissionais de engenharia, facilitando a análise estrutural preliminar em ambientes reais.

1.4 Considerações Finais

Este trabalho apresenta a WellNet, uma arquitetura híbrida que integra *Vision Transformers* com camadas convolucionais das CNNs para classificação automatizada de rachaduras e fissuras e fissuras. A proposta surge da necessidade de soluções eficientes que mitiguem riscos e custos associados a falhas estruturais.

A automação auxilia inspeções que tradicionalmente dependentes de mão de obra especializada e sujeitas a riscos ocupacionais. A WellNet ajuda com essa limitação através da combinação otimizada entre análise contextual global inicial das ViT e refinamento de características locais das camadas convolucionais das CNNs.

Os objetivos envolvem criar um modelo competitivo em métricas de desempenho, mantendo viabilidade para dispositivos de recursos limitados. A arquitetura compacta desenvolvida, apresenta tempo de inferência reduzido, característica essencial para aplicações práticas. O sistema web integrado facilita o uso por profissionais do setor.

A WellNet representa uma alternativa promissora para inspeção estrutural, oferecendo precisão, eficiência temporal e aplicabilidade prática. Este trabalho estabelece bases para futuras pesquisas em detecção e segmentação de rachaduras e fissuras na engenharia civil.

Parte I

Preparação da pesquisa

2 Referencial Teórico

Neste capítulo, exploraremos o referencial teórico para explicar os conceitos que embasam a pesquisa no campo da Visão Computacional, *Deep Learning*, *Transfer Learning* e *Vision Transformers*. Iniciaremos com as definições normativas de fissuras, trincas e rachaduras e uma fundamentação em Visão Computacional, abordando especificamente o papel das Redes Neurais Convolucionais (CNNs). Em seguida, aprofundaremos o conceito de *Transfer Learning*, uma técnica que permite reaproveitar o conhecimento adquirido por modelos pré-treinados em grandes conjuntos de dados. Por fim, introduziremos os ViTs, uma inovação na área que, baseada na arquitetura *Transformer* originalmente desenvolvida para processamento de linguagem natural, captura relações globais e dependências contextuais de longo alcance nas imagens através de mecanismos de atenção, possibilitando desempenho superior em cenários complexos, especialmente quando há disponibilidade de grandes volumes de dados para treinamento.

2.1 Definições Normativas: Fissuras e Rachaduras

Segundo as normas técnicas brasileiras vigentes, fissuras e rachaduras são definidas como separações completas ou incompletas do concreto ou argamassa, em duas ou mais partes, causadas pela quebra ou fratura do material (ABNT, 2010; ABNT, 2021). A distinção entre essas manifestações estruturais, baseia-se principalmente na amplitude de abertura apresentada pela descontinuidade no material.

As duas principais normas que estabelecem esses critérios são a NBR 9575:2010 (Impermeabilização) e a NBR 15575:2021 (Desempenho de Edificações Habitacionais). A NBR 9575:2010 classifica as aberturas em: fissuras (até 0,5 mm, incluindo microfissuras até 0,05 mm) e rachaduras (superiores a 1,0 mm) (ABNT, 2010). A NBR 15575:2021 adota critério diferente, definindo fissuras como aberturas até 0,6 mm e rachaduras como aberturas superiores a 0,6 mm (ABNT, 2021).

A NBR 6118:2023 reconhece que “as fissuras são inevitáveis em elementos estruturais” e estabelece métodos para controle de fissuração em estruturas de concreto (ABNT, 2023). A ordem hierárquica crescente de severidade é: Fissuras → Rachaduras, onde cada categoria representa aumento progressivo na amplitude da abertura e gravidade do problema estrutural. Esta classificação é fundamental para avaliação de integridade estrutural e definição de estratégias de manutenção preventiva, considerando que diferentes normas estabelecem limites ligeiramente distintos.

As diferenças entre os critérios normativos refletem a complexidade da engenharia

diagnóstica e a necessidade de interpretação contextual por profissionais habilitados. Na prática, a identificação precoce de qualquer manifestação estrutural é mais crítica que sua classificação dimensional precisa, especialmente em inspeções preventivas onde o objetivo principal é detectar anomalias antes que evoluam para problemas estruturais graves. Esta realidade fundamenta a importância de sistemas automatizados de classificação que priorizam a sensibilidade na identificação de patologias.

A variabilidade entre os critérios normativos também evidencia a natureza evolutiva do conhecimento técnico na área de durabilidade estrutural. As normas são periodicamente revisadas para incorporar novos conhecimentos científicos e experiências práticas acumuladas pela comunidade técnica. Além disso, fatores como condições ambientais, tipo de estrutura e vida útil projetada influenciam significativamente a interpretação dos limites estabelecidos, exigindo análise caso a caso por parte dos profissionais responsáveis pela avaliação estrutural. Essa complexidade interpretativa reforça a importância de ferramentas auxiliares de diagnóstico que possam complementar a expertise humana na identificação sistemática de anomalias estruturais.

Para fins desta pesquisa, todas as manifestações estruturais de descontinuidade serão tratadas genericamente sob o termo "rachaduras". Esta escolha metodológica justifica-se pelo objetivo de classificação binária em visão computacional, onde o modelo preditivo da WellNet visa identificar a presença de anomalias estruturais independentemente de sua amplitude específica. A uniformização terminológica contribui para maior generalização na avaliação automatizada de riscos em diferentes contextos construtivos. Além disso, facilita o treinamento e validação do modelo, garantindo consistência entre as bases de dados utilizadas.

2.2 Visão Computacional

Visão Computacional constitui uma subárea da Inteligência Artificial dedicada à interpretação automática de dados visuais, abrangendo tarefas como classificação, detecção e segmentação de objetos (SZELISKI, 2010). Esta disciplina integra Processamento Digital de Imagens, Aprendizado de Máquina e *Deep Learning* para capacitar sistemas computacionais na análise e tomada de decisões baseadas em informações visuais.

A Figura 1 ilustra a hierarquia entre Inteligência Artificial, Visão Computacional e Processamento de Imagem. O processamento de imagem fornece as bases técnicas para extração de características visuais, enquanto a visão computacional aplica estas técnicas em contextos de interpretação automática.

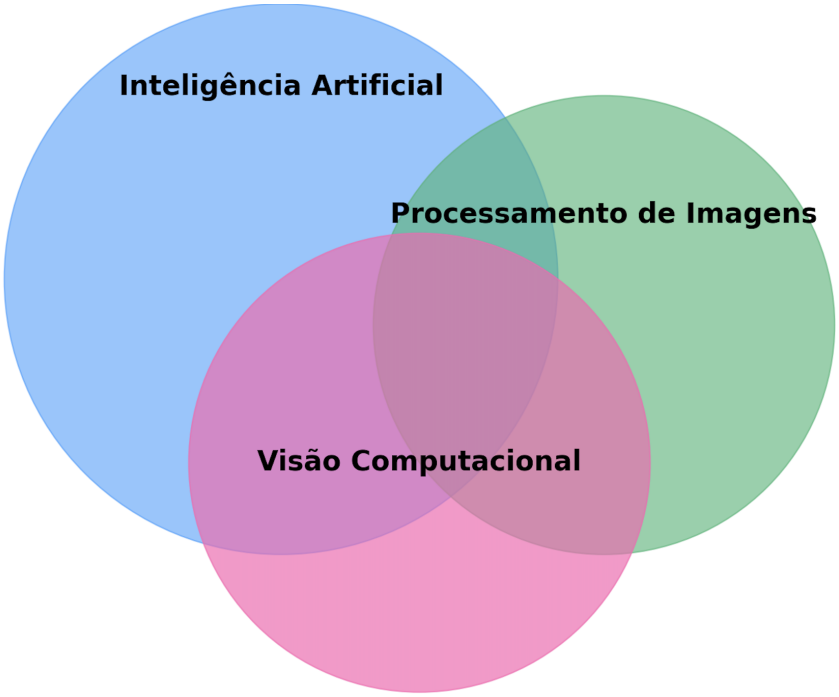


Figura 1 – Relação entre Inteligência Artificial, Visão Computacional e Processamento de Imagem.

As técnicas clássicas incluem operadores de gradiente (Sobel, Canny), transformações matemáticas (Fourier, Hough) e algoritmos de segmentação (GONZALEZ; WOODS, 2010; SZELISKI, 2010). A transformação de Hough detecta padrões lineares, sendo aplicável a rachaduras retilíneas, mas apresenta limitações em estruturas irregulares ou curvilíneas. Os operadores de gradiente destacam mudanças bruscas de intensidade, porém são sensíveis a ruído e variações de iluminação, fatores críticos em inspeções de campo. Estas abordagens oferecem interpretabilidade e baixo custo computacional, mas enfrentam limitações metodológicas: dependência de ajuste manual de parâmetros, sensibilidade a condições ambientais e dificuldade em generalizar para diferentes tipos de superfícies e padrões de rachadura (SZELISKI, 2010).

O trabalho de (CAO, 2021) aplicou visão computacional em tabuleiros de pontes, combinando melhoria de imagem e extração de características. Contudo, a validação se limitou a ambiente controlado, sem considerar variações sazonais, condições climáticas adversas ou diferentes tipos de concreto, limitações que comprometem a aplicabilidade prática. Posteriormente, (BAE; AN, 2023) propuseram quantificação estatística através de segmentação semântica profunda (técnica que classifica cada pixel da imagem em categorias específicas) seguida de cálculo dimensional via transformação da distância Euclidiana (método que determina dimensões baseado em distâncias geométricas). Embora apresentem resultados promissores, o estudo não reporta métricas de precisão, *recall* ou *F1-score*, utilizando apenas comparações dimensionais com medições reais - abordagem insuficiente para avaliar desempenho em cenários de desequilíbrio de classes, comum na

detecção de anomalias estruturais.

As técnicas baseadas em *Deep Learning*, embora demonstrem melhor adaptação a variações ambientais (LECUN; BENGIO; HINTON, 2015), enfrentam desafios significativos: necessidade de grandes conjuntos de dados rotulados, falta de interpretabilidade das decisões e custo computacional elevado. Estudos recentes carecem de validação em condições reais de campo, limitando-se frequentemente a *dataset*s controlados que não refletem a complexidade de inspeções estruturais práticas. A ausência de padronização em métricas de avaliação dificulta comparações entre métodos. Enquanto trabalhos clássicos focam em acurácia global, abordagens modernas requerem análise de precisão, *recall* e *F1-score* para avaliar adequadamente o desempenho em detecção de rachaduras, tarefa caracterizada por classes desbalanceadas onde a classe "rachadura" é minoritária.

A integração entre métodos clássicos e modernos permanece pouco explorada, representando uma lacuna para desenvolvimento de sistemas híbridos que combinem interpretabilidade, eficiência computacional e robustez em condições adversas. Esta necessidade justifica investigações em arquiteturas que aproveitem simultaneamente a transparência dos métodos tradicionais e a capacidade de generalização das técnicas de aprendizado profundo.

2.3 Deep Learning - (Aprendizado Profundo)

O *Deep Learning*(Aprendizado Profundo) representa um subconjunto do *Machine Learning*(Aprendizado de Máquina), que utiliza redes neurais artificiais com múltiplas camadas para modelar dados complexos (LECUN; BENGIO; HINTON, 2015). A denominação "profundo" se refere à arquitetura em camadas, onde cada camada processa e transforma informações da camada anterior, criando representações hierárquicas progressivamente mais abstratas. Esta abordagem permite aprendizado automático de características relevantes, eliminando a engenharia manual de *features* (características extraídas manualmente dos dados de entrada).

2.3.1 Relação entre Inteligência Artificial, Aprendizado de Máquina e Aprendizado Profundo

A Figura 2 ilustra a hierarquia entre Inteligência Artificial (IA), Aprendizado de Máquina (AM) e Aprendizado Profundo (AP). A IA constitui o campo mais amplo, englobando sistemas que realizam tarefas cognitivas como raciocínio e tomada de decisões, incluindo algoritmos de aprendizado, regras simbólicas e técnicas baseadas em lógica.

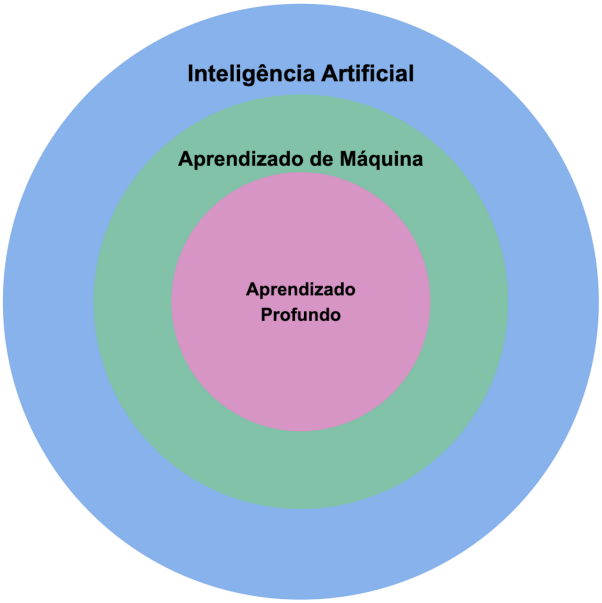


Figura 2 – Relação entre Inteligência Artificial, Aprendizado de Máquina e Aprendizado Profundo.

O AM, subárea da IA, concentra-se no desenvolvimento de algoritmos que aprendem padrões a partir de dados sem programação explícita. O AP, por sua vez, especializa-se em redes neurais profundas para modelagem de padrões complexos, sendo aplicado em reconhecimento de voz, visão computacional e classificação de anomalias estruturais.

Apesar dos avanços, o AP enfrenta limitações metodológicas consideráveis. A necessidade de grandes conjuntos de dados rotulados representa um obstáculo prático, especialmente em domínios especializados como detecção de rachaduras estruturais, onde a rotulação requer expertise técnica custosa (LECUN; BENGIO; HINTON, 2015). A "caixa preta" característica destes modelos dificulta a interpretabilidade das decisões, limitando sua aplicação em contextos que demandam explicabilidade, como inspeção de segurança estrutural.

O viés de *dataset* constitui outra limitação crítica. Modelos treinados em *datasets* controlados frequentemente apresentam desempenho degradado em condições reais devido a diferenças de distribuição entre dados de treinamento e operação, fenômeno conhecido como mudança de domínio (WANG et al., 2024). Esta limitação é particularmente relevante em aplicações de Visão Computacional para inspeção estrutural, onde variações de iluminação, ângulos de captura e tipos de superfície afetam significativamente o desempenho.

A avaliação de modelos de *Deep Learning* tradicionalmente enfatiza acurácia global, métrica inadequada para problemas com classes desbalanceadas. Em detecção de anomalias estruturais, onde rachaduras constituem uma classe minoritária, métricas como precisão (proporção de detecções corretas entre todas as detecções), *recall* (proporção de rachaduras efetivamente detectadas) e *F1-score* (média harmônica entre precisão e *recall*) fornecem

avaliação mais adequada do desempenho prático (LECUN; BENGIO; HINTON, 2015).

A validação em condições controladas, comum na literatura, não reflete desafios de implementação real. Estudos carecem de avaliação em ambientes externos, com variações climáticas, diferentes tipos de materiais e condições de manutenção diversas, fatores que impactam diretamente a aplicabilidade prática dos modelos desenvolvidos.

2.3.2 Avanços das Arquiteturas de *Deep Learning*

O uso de *Deep Learning* em Visão Computacional tem sido explorado amplamente, resultando em avanços no reconhecimento de padrões complexos e extração de características visuais. Diversos trabalhos contribuíram para a evolução das arquiteturas de Redes Neurais Profundas.

O desenvolvimento inicial foi marcado pelo trabalho de Yann LeCun, que introduziu as primeiras CNNs na década de 1990. (LECUN et al., 1998) propuseram a LeNet, arquitetura aplicada na classificação de dígitos manuscritos que se tornou ponto de partida para aplicação de redes convolucionais em problemas visuais. A LeNet demonstrou a eficácia das CNNs na extração de características visuais, preparando o terreno para avanços posteriores.

Em 2012, a AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) representou avanço considerável no *Deep Learning*. Foi a primeira arquitetura de rede neural profunda a vencer a competição *ImageNet* com margem considerável, demonstrando o poder das redes profundas para classificação de imagens em larga escala. Utilizava várias camadas convolucionais, seguidas de *pooling* e camadas totalmente conectadas, estabelecendo a base para arquiteturas modernas. Inspirou o desenvolvimento de redes mais profundas e complexas posteriormente utilizadas em tarefas como classificação, detecção e segmentação de rachaduras.

No trabalho de (HE et al., 2016), foram introduzidas inovações na arquitetura ResNet, propondo melhorias na capacidade de aprendizagem profunda, especialmente em redes residuais. Esta pesquisa melhorou a eficiência e precisão dos modelos, superando limitações que redes mais profundas enfrentavam em termos de degradação do gradiente.

A MobileNet (HOWARD et al., 2017) e a EfficientNet (TAN; LE, 2020) desempenharam papel importante, especialmente em aplicações que exigem relação balanceada entre precisão e eficiência computacional. O MobileNet é conhecido por sua leveza e eficiência em dispositivos móveis, enquanto o EfficientNet introduziu abordagem de escalonamento de modelos, permitindo obter altos níveis de precisão ao otimizar o balanceamento entre profundidade, largura e resolução do modelo.

Esses avanços, desde a LeNet (LECUN et al., 1998), passando pela AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) e chegando às ResNets (HE et al., 2016),

moldaram a base das arquiteturas modernas aplicadas à Visão Computacional. Cada arquitetura trouxe contribuição única: a LeNet abriu caminho para utilização de redes convolucionais em problemas visuais; a AlexNet provou a eficácia das redes profundas para classificação de imagens em larga escala; e a ResNet resolveu problemas de degradação do gradiente, permitindo redes mais profundas e eficazes. Arquiteturas como MobileNet (HOWARD et al., 2017) e EfficientNet (TAN; LE, 2020) ampliaram o uso de redes neurais para dispositivos móveis e sistemas embarcados, trazendo nova perspectiva de eficiência computacional.

2.4 Transfer Learning - (Aprendizado por Transferência)

O *Transfer Learning* é uma técnica que permite reutilizar o conhecimento adquirido por um modelo neural previamente treinado em uma tarefa para acelerar e melhorar o aprendizado em uma nova tarefa relacionada (PAN; YANG, 2009). Esta abordagem é valiosa quando se dispõe de dados limitados para a nova tarefa, pois aproveita representações já aprendidas de grandes conjuntos de dados, reduzindo o tempo de treinamento e a necessidade de recursos computacionais.

A pesquisa inicial de (PAN; YANG, 2010) estabeleceu o conceito e as técnicas de aprendizado por transferência, demonstrando como o conhecimento adquirido por um modelo em uma tarefa pode ser reutilizado em outra, minimizando o custo de treinamento e acelerando o desenvolvimento de novos modelos. Desde 2019, avanços permitiram adaptar modelos a domínios específicos de maneira mais eficiente, possibilitando a aplicação dessa técnica em áreas complexas como a detecção de rachaduras em estruturas de concreto.

Uma das vantagens do *Transfer Learning* é a redução no tempo de treinamento. Em problemas de Visão Computacional, frequentemente não se tem grande quantidade de dados rotulados disponível, e o treinamento do zero pode ser inviável. Através do uso de modelos pré-treinados, que já adquiriram conhecimento de características gerais a partir de grandes conjuntos de dados, é possível adaptar essas características a problemas específicos, tornando o processo de aprendizado mais rápido e eficaz.

No campo do *Deep Learning*, a transferência de aprendizagem tem se mostrado útil para lidar com desafios que envolvem modelos complexos, como o *Vision Transformers* (ViT) (DOSOVITSKIY et al., 2021). Originalmente utilizado em processamento de linguagem natural, o ViT foi adaptado para Visão Computacional e demonstrou resultados ao ser treinado em grandes conjuntos de dados, como a *ImageNet*. O ViT se destacou pela sua capacidade de capturar relações globais na imagem, indo além das características locais que as CNNs são capazes de detectar. Essa característica foi útil na construção da WellNet, que se beneficiou da integração do ViT para aprimorar a capacidade de classificação de rachaduras, conseguindo um desempenho eficiente com menor custo computacional.

A aplicação do *Transfer Learning* na WellNet permitiu reduzir a necessidade de hardware robusto durante o treinamento. Utilizando modelos pré-treinados, a WellNet conseguiu alavancar o aprendizado anterior de características visuais, sendo adaptada a partir de um *dataset* menor e específico, contendo exemplos de rachaduras em superfícies de concreto. Assim, mesmo com limitações de hardware, a WellNet foi capaz de atingir resultados, superando as limitações que existiriam caso fosse necessário treinar todo o modelo a partir do zero.

A pesquisa de (SHARMA; VARDHAN, 2023) destacou que a transferência de aprendizado é uma técnica eficaz para melhorar a precisão de modelos híbridos, especialmente em tarefas que exigem a combinação de diferentes tipos de informações visuais. Ao combinar ViT com CNN, a WellNet foi capaz de capturar tanto características globais quanto locais, resultando em um modelo que é eficiente para lidar com a complexidade das superfícies de concreto e os diferentes padrões de imagens de rachaduras presentes nas estruturas. Essa combinação de informações permite que o modelo seja mais resiliente a variações como iluminação, ângulos de captura de imagem, e tipos de superfície, problemas recorrentes na prática da inspeção visual automatizada.

A técnica de *Transfer Learning* apresenta versatilidade. Com essa abordagem, a WellNet pode ser ajustada para outras tarefas relacionadas, como a classificação de outras anomalias em superfícies ou em diferentes materiais estruturais. Isso confere ao sistema flexibilidade, permitindo que ele seja aplicado em um espectro maior de problemas dentro do setor de construção e engenharia.

Os avanços no campo do *Transfer Learning*, desde o trabalho de (PAN; YANG, 2010), até as inovações apresentadas por (DOSOVITSKIY et al., 2021) e (SHARMA; VARDHAN, 2023), foram essenciais para a concepção e desenvolvimento de modelos híbridos como a WellNet. Ao aproveitar o conhecimento prévio de modelos treinados em grandes bases de dados, como o ViT, a WellNet atingiu um nível de precisão na classificação de imagens de rachaduras em estruturas de concreto. A pesquisa mostrou que a combinação do poder de generalização dos *Transformers* com a capacidade das redes convolucionais permitiu criar um modelo com precisão e adaptabilidade. Esses avanços são primordiais, para que o modelo se torne uma solução prática no setor de construção civil, ampliando a aplicabilidade do *Deep Learning* em tarefas que exigem precisão e resiliência em ambientes reais.

2.4.1 Aplicação do Transfer Learning

A Figura 3 ilustra um fluxo genérico do processo de *Transfer Learning*, amplamente utilizado para adaptar modelos pré-treinados a novos problemas, como a classificação de imagens de rachaduras em estruturas de concreto. Este processo representa uma abordagem sistemática que permite aproveitar o conhecimento previamente adquirido por modelos

treinados em grandes bases de dados.

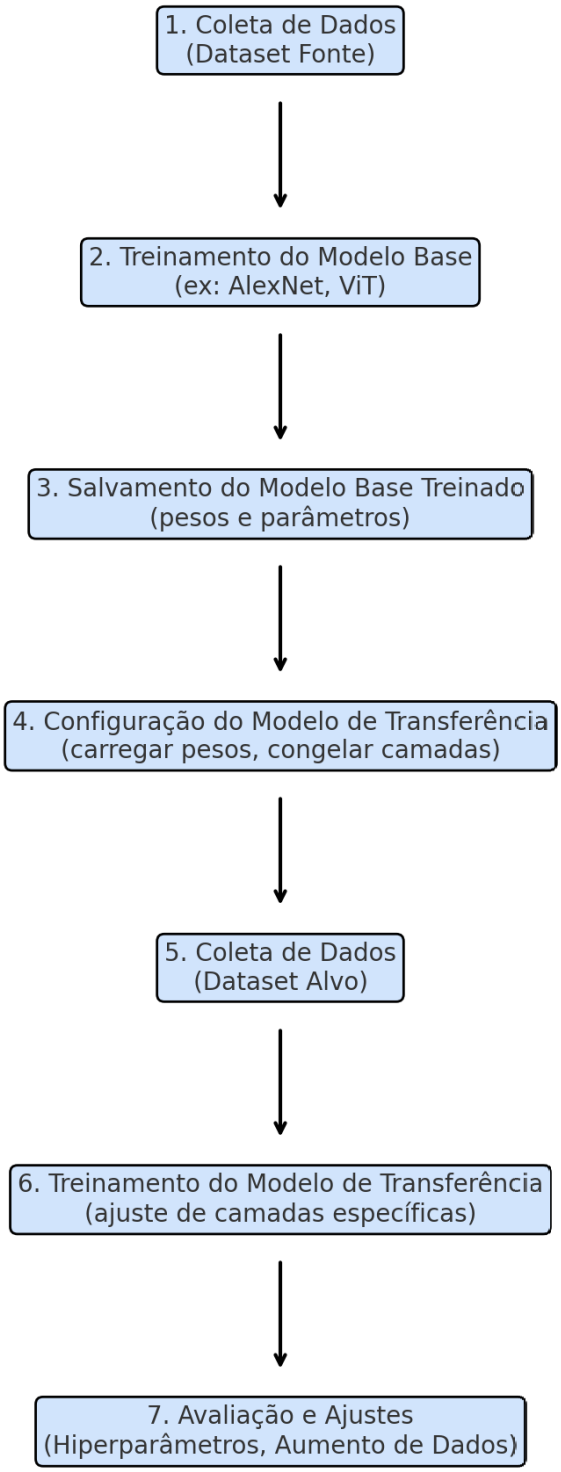


Figura 3 – Fluxo do Processo de Transfer Learning.

1. **Coleta de Dados (*dataset* Fonte)**: A primeira etapa envolve a coleta de um conjunto de dados que será utilizado como *dataset* fonte. Esses dados são necessários para o treinamento inicial do modelo base, como a *ImageNet*, que oferece variedade de exemplos visuais e permite ao modelo aprender características básicas das imagens.

2. **Treinamento do Modelo Base (ex: AlexNet, ViT)**: Nesta fase, o modelo base é treinado utilizando o *dataset* fonte. Modelos como AlexNet e ViT são exemplos de modelos base. O objetivo é que o modelo aprenda características visuais gerais, que posteriormente serão adaptadas para uma tarefa específica.

3. **Salvamento do Modelo Base Treinado (Pesos e Parâmetros)**: Após o treinamento inicial, o modelo base é salvo, incluindo seus pesos e parâmetros aprendidos. Esse modelo treinado será usado como ponto de partida para o processo de transferência de aprendizagem, economizando tempo e recursos computacionais.

4. **Configuração do Modelo de Transferência (Carregar Pesos, Congelar Camadas)**: Em seguida, o modelo pré-treinado é carregado e preparado para a transferência de aprendizagem. Algumas camadas do modelo são congeladas, geralmente as iniciais, pois já capturam características básicas das imagens (como bordas e texturas), enquanto camadas finais são deixadas livres para ajuste durante o novo treinamento.

5. **Coleta de Dados (*dataset* Alvo)**: A coleta de dados do *dataset* alvo envolve um conjunto específico de imagens relacionadas à nova tarefa. Esses dados são utilizados para adaptar o modelo base à nova tarefa, ajustando-o para uma classificação mais específica.

6. **Treinamento do Modelo de Transferência (Ajuste de Camadas Específicas)**: Com o *dataset* alvo, o modelo é treinado novamente, ajustando as camadas não congeladas. Essa etapa permite que o modelo aprenda características específicas relacionadas ao problema em questão, utilizando os dados do conjunto alvo.

7. **Avaliação e Ajustes (Hiperparâmetros, Aumento de Dados)**: Durante o processo de treinamento, são realizados ajustes nos hiperparâmetros e técnicas de aumento de dados são aplicadas para melhorar a capacidade de generalização do modelo. Isso inclui a escolha de épocas de treinamento, tamanho de lote e aplicação de *Data Augmentation*, visando garantir que o modelo não sofra de sobreajuste e possa performar bem em dados desconhecidos.

O processo de *Transfer Learning* proporciona redução no custo computacional e no tempo de treinamento, ao mesmo tempo em que aproveita características prévias aprendidas em grandes bases de dados, adaptando-as para tarefas específicas.

2.5 Vision Transformers (ViT)

Os ViTs adaptam a arquitetura *Transformers* para processamento de imagens (DOSOVITSKIY et al., 2021). A inovação consiste em tratar imagens como sequências

de *patches* visuais, aplicando o mesmo princípio usado no processamento de sequências textuais (VASWANI et al., 2017). Cada *Patch* da imagem é convertido em um *Embedding* e processado pelo mecanismo de autoatenção, permitindo que o modelo capture relações espaciais globais na imagem sem as limitações de campo receptivo das convoluções tradicionais (AL-MAQTARI et al., 2024).

A arquitetura *Transformers* constitui um modelo de rede neural que processa sequências utilizando exclusivamente mecanismos de atenção, prescindindo de convoluções ou recorrência (VASWANI et al., 2017). Originalmente desenvolvida para Processamento de Linguagem Natural, esta arquitetura permitiu o processamento paralelo de sequências inteiras, superando limitações de modelos sequenciais tradicionais como Redes Neurais Recorrentes (RNNs) e redes de Memória de Longo e Curto Prazo (LSTMs) (HOUDT; MOSQUERA; NÁPOLES, 2020). O *Transformer*, baseia-se em uma arquitetura encoder-decoder, onde o encoder processa a sequência de entrada e o decoder gera a sequência de saída (VASWANI et al., 2017). Cada bloco contém camadas de autoatenção multi-cabeça seguidas por redes neurais feedforward, com conexões residuais e normalização de camadas (BA; KIROS; HINTON, 2016).

O conceito de *Embedding*, refere-se ao processo de conversão de dados categóricos ou de alta dimensionalidade em representações vetoriais densas de dimensionalidade fixa (BENGIO; COURVILLE; VINCENT, 2013). Em termos práticos, é como criar um "código numérico" que preserva as características dos dados originais em um espaço vetorial contínuo. No contexto de Processamento de Linguagem Natural, palavras são convertidas em vetores numéricos que capturam suas relações semânticas (MIKOLOV et al., 2013). Para *Vision Transformers*, cada *patch* da imagem é convertido em um vetor de números reais através de transformações lineares aprendíveis (DOSOVITSKIY et al., 2021). Este vetor de *Embedding* captura as características visuais do *patch*, como cores, texturas e padrões, em uma forma que os mecanismos de atenção podem processar matematicamente.

Por exemplo, se uma imagem de 224×224 pixels é dividida em *patches* de 16×16 pixels, resulta-se em 196 *patches*, cada um convertido em um vetor de 768 dimensões (DOSOVITSKIY et al., 2021). Este processo preserva a informação visual relevante enquanto padroniza a representação para processamento pelo *Transformer*. Adicionalmente, são adicionados *Embeddings* posicionais para preservar informações sobre a localização espacial de cada *patch* na imagem original (GEHRING et al., 2017).

O mecanismo de *Self-Attention* (autoatenção) representa o núcleo dos *Transformers* (VASWANI et al., 2017). Para compreender seu funcionamento, considere uma analogia com a leitura humana: ao ler uma frase, frequentemente relacionamos palavras distantes para compreender o contexto completo. A autoatenção opera de forma similar, permitindo que cada elemento da sequência "observe" todos os outros elementos simultaneamente. Tecnicamente, para cada elemento de uma sequência, o mecanismo calcula três representações

vetoriais através de transformações lineares aprendíveis (VASWANI et al., 2017): **Query** (Q) representa "o que estou procurando", **Key** (K) representa "o que posso oferecer" e **Value** (V) contém "a informação útil".

O processo de atenção calcula a similaridade entre a *Query* de um elemento e todas as *Keys* da sequência através do produto escalar, determinando quanta "atenção" cada elemento deve receber (BAHDANAU; CHO; BENGIO, 2014). Esta pontuação de atenção é normalizada pela função *Softmax* e utilizada para ponderar os *Values*, gerando uma representação final que incorpora informações contextuais de toda a sequência. Matematicamente, isso é expresso como (VASWANI et al., 2017):

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

onde d_k é a dimensionalidade das chaves, utilizada para normalização e estabilização do gradiente durante o treinamento. O mecanismo de multi-cabeça (*Multi-Head Attention*) aplica múltiplas funções de atenção em paralelo, permitindo ao modelo focar em diferentes tipos de relações simultaneamente (VASWANI et al., 2017).

Os ViTs, propostos por (DOSOVITSKIY et al., 2021), introduziram uma abordagem para o uso de autoatenção em imagens. O trabalho mostrou que os *Transformers* poderiam ser aplicados em tarefas visuais, alcançando resultados comparáveis ou superiores às CNNs quando treinados em grandes conjuntos de dados. Ao dividir as imagens em pequenos *patches* e tratá-los como *tokens*, os ViTs conseguem capturar relações globais em uma imagem, superando as limitações das CNNs, que se concentram em características locais.

A introdução dos ViTs representou uma mudança de paradigma em relação ao modo como as características visuais são extraídas e processadas, inspirando o desenvolvimento de soluções híbridas como a WellNet. A WellNet combina as vantagens dos Vision Transformers, que capturam relações globais nas imagens, e das CNNs, que extraem características locais, resultando em uma arquitetura eficiente para a classificação de rachaduras.

Em 2021, (TOUVRON et al., 2021) apresentaram o DeiT (*Data-efficient Image Transformer*), um ViT que melhorou a eficiência dos *Transformers* em termos de dados e custo computacional no treinamento. A contribuição do DeiT foi a possibilidade de treinar modelos baseados em *Transformers* de maneira mais eficiente, utilizando menos dados e reduzindo o custo computacional, tornando os ViTs acessíveis para um maior número de aplicações. O DeiT se mostrou útil em contextos onde a quantidade de dados disponíveis é limitada, desafio comum na classificação de anomalias estruturais.

Os ViTs, ao aplicarem autoatenção, conseguem identificar relações contextuais em diferentes regiões da imagem, tratando cada *patch* como um *token*, similar ao processamento de sequências em NLP. Isso difere das CNNs, que utilizam convoluções locais e filtros

para detectar padrões como bordas, texturas e formas. Enquanto as CNNs são eficazes em capturar detalhes locais, os ViTs se destacam por captar a "visão geral" da imagem, criando uma abordagem complementar.

Recentemente, (LIU et al., 2021) propuseram os *Swin Transformers*, uma variante do ViT que introduz janelas deslizantes para particionar imagens em regiões menores, permitindo que a autoatenção seja aplicada localmente. Essa abordagem híbrida de atenção local e global foi adotada como base para aplicações de visão, incluindo a classificação de falhas e rachaduras. Os *Swin Transformers* são mais eficientes em termos de recursos computacionais, fator relevante para a implementação em dispositivos móveis e drones.

A evolução dos ViTs, desde o trabalho de (DOSOVITSKIY et al., 2021), passando pelas melhorias de eficiência propostas por (TOUVRON et al., 2021), até as inovações dos *Swin Transformers* de (LIU et al., 2021), permitiram que modelos híbridos se tornassem mais robustos e aplicáveis em situações do mundo real. A combinação das técnicas de aprendizado por atenção global e extração de características locais torna a WellNet uma ferramenta para a inspeção automatizada de imagens rachaduras em estruturas de concreto.

3 Estado da Arte

Neste capítulo, exploraremos o estado da arte, onde abordaremos os trabalhos relacionados à classificação, detecção e segmentação de imagens de rachaduras, contextualizando as principais contribuições, desafios e limitações das técnicas existentes. Essa revisão estabelece a base para a proposta da WellNet, que visa unir o melhor dessas abordagens para melhorar a precisão e a eficiência na classificação de imagens contendo rachaduras e fissuras.

3.1 Trabalhos Relacionados à Classificação de Rachaduras

Nos últimos anos, a classificação de imagens de rachaduras em superfícies de concreto tem se beneficiado enormemente dos avanços no campo de *Deep Learning* e aprendizado por transferência. Diversos trabalhos se destacaram no uso de redes convolucionais, modelos híbridos e *Transformers* para melhorar a precisão e eficiência das soluções, cada um contribuindo com abordagens específicas que influenciaram o desenvolvimento da WellNet.

3.1.1 Fundamentos das Abordagens Convolucionais

O estudo de (ZOU et al., 2019) propôs a DeepCrack, que utiliza convoluções hierárquicas para a classificação de rachaduras, demonstrando robustez ao processar texturas complexas e variações na iluminação. A aplicação de convoluções profundas permitiu à DeepCrack identificar características finas, elemento essencial para a classificação de rachaduras de diferentes intensidades. Esta abordagem estabeleceu princípios fundamentais para arquiteturas baseadas em fusão multi-escala, influenciando diretamente o desenvolvimento de soluções como a WellNet.

O trabalho de (DINIZ et al., 2023) utilizou uma arquitetura CNN com o conjunto de dados (Özgenel, 2019), alcançando uma precisão de 99,47%. Este resultado destacou a eficácia das redes profundas para tarefas de inspeção estrutural, servindo como importante referência para o desenvolvimento de soluções como a WellNet. A metodologia empregada demonstrou o potencial das CNNs para classificação de patologias em estruturas de concreto, estabelecendo benchmarks importantes para trabalhos subsequentes.

3.1.2 Evolução para Arquiteturas Transformer

No cenário de detecção de rachaduras em pavimentos, (XIAO et al., 2023) propuseram uma abordagem inovadora utilizando *Transformers* atenciosos de janelas híbridas, denominada CrackFormer, para identificação automatizada de rachaduras. O CrackFormer

emprega uma rede de alta resolução baseada em *Transformers* para explorar e combinar características semânticas em múltiplas escalas, apresentando bom desempenho com precisão de 93,76%. Esta abordagem é especialmente eficaz para melhorar os detalhes semânticos e a precisão da detecção de rachaduras sob diversas condições de superfície de pavimento.

Pesquisas recentes demonstraram avanços significativos na aplicação de ViT. (SHAHIN et al., 2024) apresentaram uma abordagem inovadora combinando ViT com detectores de aprimoramento de imagem para aumentar a precisão na detecção de rachaduras em pontes de concreto. O estudo comparou o modelo ViT com uma CNN personalizada, atingindo acurácia superior a 99%. Esta pesquisa reforçou a viabilidade dos *Transformers* para capturar dependências de longo alcance em estruturas de rachaduras.

3.1.3 Integração de Arquiteturas Híbridas

As arquiteturas híbridas *CNN-Transformers* têm demonstrado resultados promissores em diversas aplicações de detecção estrutural. (KHATTAK et al., 2024) propuseram o CCTNet, um modelo que combina características de CNNs com *Vision Transformers* para detecção eficiente de rachaduras em estruturas prediais. O modelo CCTNet alcançou 97,49% de F1-score no *dataset* SDTNet2018, demonstrando a eficácia de arquiteturas híbridas que integram a capacidade de extração de características locais das CNNs com o poder de modelagem global dos *Transformers*. Esta abordagem validou os princípios arquiteturais adotados na WellNet, confirmando a importância da integração entre diferentes componentes de *Deep Learning*.

O desenvolvimento de técnicas híbridas tem sido impulsionado por contribuições significativas no tratamento de dados com distribuição irregular. (SONG et al., 2024) desenvolveram uma abordagem para identificação de rachaduras em pontes de concreto utilizando fotografia com veículos aéreos não tripulados e aprendizado profundo. O estudo propõe uma função de perda híbrida considerando mineração de amostras negativas baseada no aprimoramento da função de perda de entropia cruzada binária.

3.1.4 Otimização Computacional e Aplicações Práticas

Pesquisas recentes têm enfatizado a importância da otimização computacional em modelos de detecção para aplicações práticas. (RAMANATHAN et al., 2025) realizaram uma análise comparativa abrangente de modelos de *Deep Learning* para detecção de rachaduras em edificações, considerando fatores como estresses estruturais e mudanças climáticas que afetam a formação de rachaduras. O estudo forneceu *insights* valiosos sobre a seleção de arquiteturas otimizadas para diferentes cenários de aplicação, enfatizando a necessidade de modelos compactos e eficientes para implementação em dispositivos

embarcados. Estes resultados influenciaram significativamente as decisões de *design* da WellNet, particularmente na otimização do tamanho do modelo e na seleção de componentes que maximizam a eficiência computacional.

A utilização de técnicas de Visão Computacional com veículos aéreos não tripulados tem se mostrado promissora. (KARMOKAR et al., 2020) propuseram um método que combina técnicas de Visão Computacional com UAVs, permitindo realizar inspeções automatizadas em locais de difícil acesso. Utilizando uma rede neural convolucional como a AlexNet para detecção de rachaduras, o estudo demonstrou a viabilidade da abordagem ao fornecer uma solução prática para o monitoramento de estruturas em tempo real.

3.1.5 Inovações Arquiteturais Recentes

Desenvolvimentos recentes demonstram evolução contínua das arquiteturas híbridas. (DONG; LIU; DAI, 2025) propuseram o AG-TransUNet, uma abordagem baseada em TransUNet melhorado que incorpora um mecanismo de auto-atenção multi-cabeça adaptativo e um módulo de decodificação baseado em mecanismo de porta (GRU-T) para aprimorar a precisão e completude da detecção de rachaduras. O modelo foi desenvolvido para abordar desafios como iluminação desigual, interferência de ruído e fundos complexos que frequentemente levam a detecções incompletas em superfícies de concreto.

No trabalho de (CHEN et al., 2024) propuseram o CrackNet, uma arquitetura híbrida que combina CNNs e *Transformers* para segmentação de rachaduras com função de perda dinâmica. Esta abordagem utiliza CNNs no módulo *encoder* para extrair características multi-nível locais, enquanto os *Transformers* são empregados para modelar dependências globais, enfrentando desafios como topologia complexa de rachaduras e fundos ruidosos. A metodologia do CrackNet inspirou aspectos fundamentais da WellNet, particularmente na forma como diferentes componentes da arquitetura híbrida são integrados.

3.1.6 Integração de Conhecimentos na WellNet

O desenvolvimento da WellNet está diretamente ligado às contribuições dos trabalhos citados neste estudo. A abordagem da AlexNet, proposta por (KRIZHEVSKY; SUTSKEVER; HINTON, 2012), inspirou o uso de convoluções para lidar com texturas e bordas, elementos essenciais na classificação e detecção de rachaduras. A utilização de convoluções profundas estabeleceu a base para redes neurais capazes de capturar características importantes, crucial para a implementação da WellNet.

A ideia dos *Vision Transformers* (ViT), descrita por (DOSOVITSKIY et al., 2021), influenciou diretamente a WellNet ao mostrar como os *Transformers* podem ser aplicados para explorar imagens de forma eficiente, capturando relações globais. A proposta do

CrackFormer também teve impacto na WellNet, utilizando *Transformers* para explorar múltiplas escalas de características visuais. Esse conceito foi incorporado na WellNet para melhorar a análise em diferentes níveis de detalhe, contribuindo para uma detecção mais precisa das rachaduras em diversas condições de superfície.

Investigações contemporâneas ampliaram o escopo de análise comparativa entre diferentes arquiteturas. (SHAHIN et al., 2025) realizaram um estudo comparativo entre modelos ViT e Redes Neurais Convolucionais (CNN) para detecção de rachaduras, utilizando um *dataset* de 30.000 imagens com dimensões 227x227 pixels. Esta pesquisa reforça a importância da arquitetura proposta na WellNet, que combina as vantagens dos *Transformers* para capturar dependências de longo alcance com a eficiência das CNNs para extração de características locais para o ajuste fino.

3.1.7 Direções Futuras e Limitações a Serem Exploradas

Embora a WellNet tenha sido desenvolvida e validada com *dataset*s balanceados, uma limitação importante a ser explorada em trabalhos futuros é a adaptação para cenários com classes desbalanceadas, comum em aplicações reais onde superfícies íntegras predominam sobre defeitos estruturais.

A abordagem para esta limitação pode se beneficiar das técnicas de função de perda híbrida propostas por (SONG et al., 2024), que combinam perdas focais com penalizações específicas para classes minoritárias, e das abordagens de tratamento de categorias desbalanceadas desenvolvidas no CrackNet. Técnicas complementares incluem amostragem estratificada, aumento de dados direcionado para a classe minoritária e ajuste de limiares de classificação específicos para otimizar métricas críticas como *recall* em cenários desbalanceados.

Esses avanços e a combinação das tecnologias mencionadas permitiram que a WellNet evoluísse como uma solução que integra características locais das CNNs com dependências globais dos *Transformers*, incorporando técnicas convolucionais, baseadas em *Transformers* e híbridas desenvolvidas pela comunidade científica para inspeções automatizadas em campo. A arquitetura híbrida aproveita a capacidade das camadas convolucionais para extrair características de baixo nível, como bordas e texturas, enquanto o *Vision Transformer* captura padrões globais e relações espaciais cruciais para distinguir fissuras reais de artefatos visuais.

3.2 Comparação de Características com Trabalhos Relacionados

Para avaliar a eficácia e as inovações da WellNet, foi realizada uma análise comparativa entre alguns trabalhos selecionados, conforme apresentado na Tabela 1. A comparação

abrange as principais características desses modelos de detecção e classificação de rachaduras, destacando as especificidades e os diferenciais de cada abordagem.

Tabela 1 – Tabela Comparativa entre a WellNet e Outras Redes para Classificação, Detecção e Segmentação de Rachaduras

Rede Neural	**Arquitetura**	**Base de Dados**	**Diferenciais**	**Tipo de Tarefa**
WellNet	Híbrida ViT-CNN	Özgenel (10.000 imagens para treino, 10.000 para teste)	Modelo leve (26.4MB), alta precisão, eficiência em diferentes dispositivos	Classificação
KARMOKAR et al.	CNN (AlexNet)	Özgenel (40.000 imagens: 20.000 para treino, 20.000 para teste)	Uso de janelamento deslizante e VANT para localização de rachaduras	Detecção
DINIZ et al.	CNN para classificação	Özgenel (20.000 imagens para cada classe: 15.000 treino, 5.000 teste)	Alta precisão para classificação de rachaduras	Classificação
ZOU et al.	SegNet modificado	CrackTree260, CRKWH100, CrackLS315, Stone331 (1.006 imagens para treino)	Fusão de características hierárquicas multiescalares	Detecção e Segmentação
SHAHIN et al.	Vision Transformer Híbrido	Özgenel (3.000 imagens: 2.100 para treino, 900 para teste)	Uso de transformador visual para alta acurácia	Classificação
XIAO et al.	Hybrid-Window AttentiveVision Transformers	Multiple *datasets* (crack detection *datasets*)	Transformers com janelas de atenção híbridas para detecção em pavimentos	Detecção

3.2.1 Diferenciais da WellNet

A WellNet, distingue-se por sua arquitetura que integra ViT e CNNs. Esta combinação permite que o modelo capture características globais através dos mecanismos de atenção do ViT e características locais por meio das convoluções da CNN. Esta abordagem é eficaz para análise de rachaduras, onde a classificação requer entendimento do contexto global da superfície e identificação de padrões locais das rachaduras.

Os principais diferenciais da WellNet são: sua abordagem não convencional ViT→ CNN enquanto na literatura o padrão abordado é CNN→ViT; seu tamanho compacto de 26,4 MB e seu tempo de inferência, que será apresentado em seções posteriores. Estas características tornam o modelo viável para implementação em dispositivos com recursos computacionais limitados, incluindo sistemas embarcados, *smartphones*, drones e dispositivos IoT para monitoramento estrutural.

A WellNet apresenta os resultados de acurácia, precisão, *recall*, F1-*score* e curva ROC demonstrados na seção de resultados. O modelo foi treinado e validado com um conjunto de dados utilizado pela comunidade acadêmica, incorporando técnicas de aumento de dados para melhorar sua capacidade de generalização em condições variadas de iluminação, ângulo e textura da superfície.

3.2.2 Características de Outras Abordagens

A rede de (KARMOKAR et al., 2020) utiliza técnicas de janelas deslizantes em drones para localização de rachaduras em grandes áreas. Este método aumenta a abrangência da detecção, permitindo análise de superfícies extensas, mas torna o modelo mais complexo em termos de processamento. O uso de drones amplia o alcance do monitoramento, mas requer aparato técnico e recursos adicionais.

O modelo de (ZOU et al., 2019) implementa fusão de características hierárquicas multiescalares, com arquitetura baseada em SegNet modificada para detecção e segmentação de rachaduras. Esta abordagem é eficaz em cenários que demandam segmentação detalhada, mas requer mais recursos computacionais para processamento de imagens segmentadas.

A abordagem de (DINIZ et al., 2023) utiliza CNN tradicional para classificação, atingindo precisão na identificação de rachaduras. Por ser uma rede focada em características locais, apresenta limitações ao lidar com imagens que possuem variabilidade visual, como diferentes ângulos de captura e condições de iluminação. Esta abordagem é eficiente para contextos onde a uniformidade das imagens é maior.

A abordagem (XIAO et al., 2023) implementa *Transformers* com janelas de atenção híbridas especificamente para detecção de rachaduras em pavimentos. Esta arquitetura combina atenção local através de janelas fixas com atenção global por meio de janelas deslizantes, permitindo captura simultânea de características locais e dependências de longo alcance. O método de detecção em pavimentos. A estratégia de janelas híbridas reduz a complexidade computacional mantendo capacidade de modelagem global.

No trabalho de (SHAHIN et al., 2024) foi proposto uma arquitetura híbrida baseada em *Transformer* visual, explorando informações contextuais para melhorar a capacidade de generalização. Esta abordagem, destaca-se pela capacidade de capturar contextos amplos em imagens de rachaduras. O uso de conjunto de dados menor pode limitar a robustez do modelo em cenários variados.

3.2.3 Contribuições e Potenciais de Aplicação

A análise comparativa indica que a WellNet oferece uma alternativa eficiente para classificação de rachaduras, sendo adequada para aplicações que demandam soluções ágeis. A arquitetura permite que o modelo seja utilizado em sistemas com restrições de

processamento e armazenamento, como dispositivos móveis e drones, mantendo a precisão necessária para inspeções preventivas.

A WellNet está focada na tarefa de classificação de imagens que contêm rachaduras, limitando sua aplicação para cenários que requerem segmentação ou detecção de áreas específicas da superfície. Mas já em desenvolvimento a funcionalidade de detecção de rachaduras para ser incorporada à WellNet, fornecendo análise mais detalhada da localização das rachaduras.

Os resultados da WellNet sugerem potencial para uso em inspeções de campo e monitoramento remoto, permitindo identificação de rachaduras de forma acessível e escalável. A facilidade de implementação em dispositivos de baixo custo reforça seu valor para manutenção preventiva em larga escala, promovendo monitoramento eficiente no setor de construção civil. A WellNet se posiciona como ferramenta valiosa para empresas que buscam soluções acessíveis para auxiliar os técnicos nas inspeções de infraestruturas.

3.2.4 Abordagem ViT→CNN: Fundamentação Teórica e Comparação Arquitetural

A escolha da arquitetura ViT→CNN na WellNet, fundamenta-se na natureza específica das rachaduras como estruturas visuais e nas características complementares dos paradigmas *Vision Transformer* e redes convolucionais. Esta seção apresenta a fundamentação teórica desta decisão arquitetural e compara a abordagem proposta com alternativas existentes na literatura.

3.2.4.1 Fundamentação Teórica da Arquitetura ViT→CNN

As rachaduras em superfícies de concreto apresentam características visuais específicas que influenciam diretamente a escolha arquitetural:

- **Estruturas globais lineares:** Rachaduras manifestam-se como descontinuidades lineares que frequentemente percorrem extensões significativas da superfície, requerendo análise contextual global para identificação completa de sua trajetória;
- **Identificação de trajetória completa:** O ViT identifica instantaneamente a trajetória completa através do mecanismo de *Self-Attention*, permitindo captura simultânea de dependências espaciais de longo alcance entre diferentes regiões da rachadura;
- **Refinamento de características locais:** As CNNs refinam características locais específicas identificadas pelo ViT, especializando-se em padrões texturais e morfológicos que distinguem rachaduras de artefatos visuais similares para decisão final precisa.

A sequência ViT→CNN aproveita esta complementaridade, o componente ViT estabelece o contexto global e identifica regiões candidatas, enquanto as camadas convolucionais subsequentes realizam análise detalhada dessas regiões para classificação definitiva.

3.2.4.2 Comparação com Abordagens Arquiteturais Alternativas

A Tabela 2 apresenta análise comparativa entre diferentes estratégias de integração CNN→ ViT e Vi→ CNN encontradas na literatura.

Tabela 2 – Comparação entre abordagens arquiteturais híbridas para detecção de rachaduras

Trabalho	**Arquitetura**	**Abordagem**
WellNet (Proposto)	ViT → CNN	ViT-*Tiny* global + refinamento CNN local
CCTNet (KHATTAK et al., 2024)	CNN → ViT	Extração CNN + modelagem ViT
CrackNet (CHEN et al., 2024)	CNN → ViT	CNN multi-nível + ViT dependências globais
CrackFormer (XIAO et al., 2023)	Híbrido *Transformer*	Janelas atenção híbridas

3.2.4.3 Análise das Abordagens CNN→ViT:

A abordagem CNN→ViT, utilizada por CCTNet (KHATTAK et al., 2024) e CrackNet (CHEN et al., 2024), fundamenta-se no processamento hierárquico tradicional onde características locais são extraídas primeiramente pelas CNNs e posteriormente integradas pelo componente ViT para modelagem de dependências globais. Esta estratégia segue o paradigma convencional de análise bottom-up, construindo representações complexas a partir de características básicas.

O CCTNet demonstra eficácia na combinação de extração convolucional com capacidades de modelagem *Transformer*. Contudo, a sequência CNN→ViT pode resultar em perda de informações contextuais críticas durante a etapa inicial de extração local, limitando a capacidade de identificação de padrões lineares extensos característicos de rachaduras.

O CrackNet utiliza CNN multi-nível seguida de processamento ViT para captura de dependências globais, abordando desafios como topologia complexa de rachaduras e fundos ruidosos. Esta abordagem mantém o processamento hierárquico tradicional, onde o contexto global é construído gradualmente através de múltiplas camadas convolucionais.

3.2.4.4 Vantagens Conceituais da Abordagem ViT→CNN:

A arquitetura ViT→CNN implementada na WellNet oferece vantagens conceituais específicas para classificação de rachaduras:

1. **Análise global imediata:** O processamento inicial pelo ViT permite identificação instantânea de padrões lineares globais sem dependência de hierarquia convolucional prévia, essencial para reconhecimento de rachaduras que podem se estender por toda a imagem;

2. **Preservação de contexto espacial:** O mecanismo de *Self-Attention* mantém correspondência espacial entre regiões distantes da imagem, crucial para compreensão da continuidade estrutural das rachaduras;

3. **Refinamento direcionado:** As camadas convolucionais subsequentes concentram-se especificamente no refinamento de regiões identificadas pelo ViT, otimizando o processamento computacional para características relevantes;

4. **Aproveitamento de *Transfer Learning*:** A utilização de ViT pré-treinado permite aproveitamento de representações semânticas robustas aprendidas em grandes conjuntos de dados, reduzindo dependência de dados específicos do domínio.

3.2.4.5 Limitações da Integração Híbrida:

A abordagem ViT→CNN apresenta desafios técnicos que devem ser considerados:

- **Complexidade de integração:** A interface entre componentes ViT e CNN requer design cuidadoso para preservar informações espaciais relevantes durante a transição entre paradigmas arquiteturais;

- **Dependência de pré-treinamento:** A eficácia depende significativamente da qualidade das representações pré-aprendidas do componente ViT para o domínio de aplicação;

- **Balanceamento arquitetural:** A proporção adequada entre componentes ViT e CNN deve ser ajustada conforme a complexidade específica da tarefa de classificação.

A arquitetura ViT→CNN, diferencia-se por combinar eficientemente análise global inicial com refinamento local especializado, estabelecendo fluxo de processamento que prioriza identificação contextual antes do detalhamento morfológico. Esta sequência é particularmente adequada para rachaduras, que requerem compreensão simultânea de continuidade espacial e características texturais locais para classificação precisa.

4 Materiais e Métodos

Neste capítulo, abordaremos os materiais utilizados, as técnicas empregadas e os métodos adotados para o desenvolvimento e avaliação do modelo proposto. Discutiremos as ferramentas utilizadas para a coleta e processamento dos dados, os detalhes da arquitetura do modelo, os procedimentos de treinamento e validação, além das métricas utilizadas para a avaliação de desempenho. Iniciaremos com a descrição do conjunto de dados, seguido pela apresentação das técnicas de pré-processamento, e finalizaremos com a explicação detalhada dos experimentos conduzidos, destacando as estratégias adotadas para garantir a robustez do modelo.

4.1 Ferramentas, Bibliotecas e Métodos Utilizados

O desenvolvimento da WellNet envolveu uma variedade de ferramentas e bibliotecas fundamentais, todas integradas através da linguagem de programação Python. Nesta seção, apresentamos uma visão geral das principais tecnologias, métodos e hiperparâmetros que foram empregados ao longo da pesquisa, destacando sua contribuição para a eficiência e eficácia da implementação do modelo.

4.1.1 Ferramentas e Bibliotecas Utilizadas

Para o desenvolvimento da WellNet, uma combinação de ferramentas e bibliotecas foi utilizada, garantindo eficiência e flexibilidade no processo de implementação e experimentação:

- **Python**: escolhido devido à sua ampla adoção pela comunidade de aprendizado profundo e pela vasta disponibilidade de bibliotecas especializadas, facilitando o desenvolvimento e a prototipagem do modelo.
- **PyTorch e TorchVision**: PyTorch foi a principal biblioteca utilizada para definir as camadas da rede, bem como implementar as etapas de treinamento e validação do modelo. TorchVision complementou esse processo, fornecendo funções para o pré-processamento das imagens, como normalização e técnicas de aumento de dados, essenciais para melhorar a generalização do modelo.
- **timm**: esta biblioteca do (WIGHTMAN, 2019) foi utilizada para o carregamento de modelo pré-treinado, como o ViT *Tiny*, permitindo a integração de pesos pré-treinados na *ImageNet* e reduzindo o tempo necessário para o treinamento do modelo.

- **NumPy e Pandas**: NumPy foi empregada para a manipulação de dados numéricos e cálculos matriciais, enquanto Pandas foi usada para organizar e analisar os resultados dos experimentos, facilitando a interpretação dos dados obtidos.

- **Matplotlib**: utilizada para a visualização dos resultados, como gráficos de perda e acurácia ao longo das épocas, auxiliando na análise do comportamento do modelo durante o treinamento e validação.

- **Jupyter Notebook**: escolhido para a prototipagem do modelo, permitindo uma execução interativa do código e facilitando a análise dos resultados de maneira fácil de visualizar.

Essas ferramentas e bibliotecas, em conjunto, formaram uma base para o desenvolvimento da WellNet, garantindo um fluxo de trabalho eficiente, desde o pré-processamento dos dados até a análise final dos resultados.

4.2 Base de Dados

Para o treinamento e validação da WellNet, utilizou-se o conjunto de dados do (Özgenel, 2019). Um dataset de alta qualidade permite que a rede aprenda características relevantes das imagens, garantindo uma boa capacidade de generalização e evitando o sobreajuste, que é quando o modelo se torna muito específico ao conjunto de treinamento e não performa bem em novos dados. Além disso, datasets amplamente utilizados e validados pela comunidade científica, como o (Özgenel, 2019) , possuem uma ampla variedade de exemplos que ajudam a assegurar que o modelo esteja preparado para enfrentar os diversos desafios encontrados em cenários reais, como diferentes condições de iluminação, ângulos de captura, e variações de textura e cor. Dessa forma, a utilização desse conjunto de dados contribuiu diretamente para a confiabilidade da WellNet.

4.2.1 Dataset (Özgenel, 2019)

Os dados utilizados foram adquiridos do conjunto de dados (Özgenel, 2019), que contêm imagens de superfícies de concreto com e sem rachaduras e fissuras. O conjunto de dados foi dividido em duas classes: imagens negativas (sem rachaduras e fissuras) e positivas (com rachaduras e fissuras), cada uma contendo 20.000 imagens, totalizando 40.000 imagens com dimensões de 227x227 *pixels* e canais *RGB (Red, Green, Blue)*. As imagens foram geradas a partir de 458 imagens de alta resolução (4032x3024 *pixels*), que apresentam variação nas condições de acabamento da superfície e iluminação.

As Figuras 4 e 5 ilustram imagens do dataset do (Özgenel, 2019):



Figura 4 – Exemplo: Imagens de rachaduras.

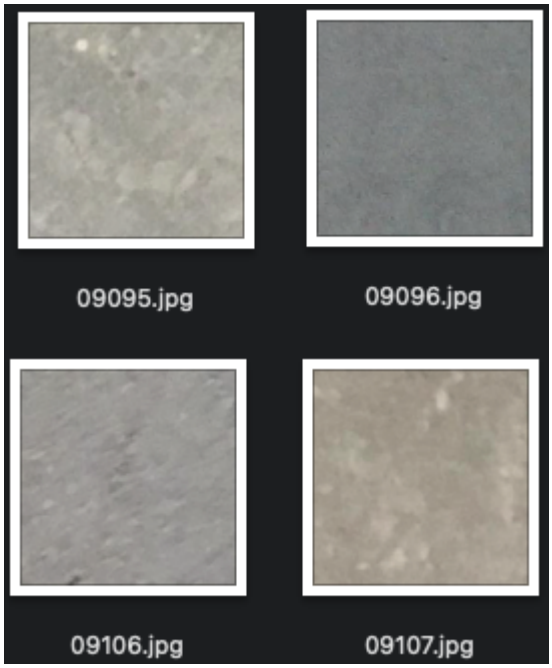


Figura 5 – Exemplo: Imagens de Não rachaduras.

4.3 Aquisição e Pré-processamento dos Dados

Para este trabalho, foram selecionadas 10.000 imagens para treinamento e 10.000 imagens para teste de inferência do modelo com o dataset (Özgenel, 2019), totalizando 20.000 imagens das 40.000 disponíveis. No treinamento foi usado a proporção 80% (treino/validação) e 20% para teste com uso da Validação Cruzada K-Fold, representando uma escolha metodológica para avaliar a capacidade de generalização com maior rigor, considerando que o modelo seria posteriormente integrado a um protótipo web.

As imagens passaram por etapas de pré-processamento essenciais: redimensionamento para 224x224 pixels (dimensão padrão requerida pela arquitetura *Vision Transformer*) e aplicação de técnicas de aumento de dados. O aumento de dados incluiu rotação, reflexão horizontal, ajustes de brilho e contraste para aumentar a variabilidade e reduzir o risco de *overfitting*, problema comum em arquiteturas complexas quando treinadas com datasets limitados.

O redimensionamento para 224x224 pixels, embora necessário para compatibilidade com ViT, implica em potencial perda de detalhes finos das rachaduras e fissuras presentes em imagens de maior resolução original. Esta limitação é compensada parcialmente pelas técnicas de aumento de dados, que simulam variações encontradas em condições reais de captura.

A Figura 6 ilustra o fluxograma do pré-processamento das imagens.

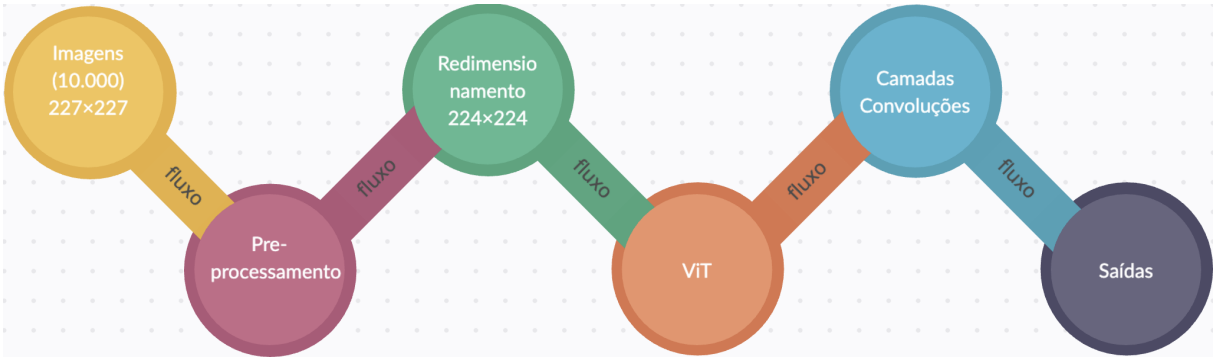


Figura 6 – Fluxograma do Pré-Processamento.

4.4 Divisão dos Dados e Validação Cruzada

Para avaliar a robustez e capacidade de generalização do modelo WellNet, foi implementada validação cruzada com 5 *folds* (*5-fold Cross-Validation*). Esta técnica estatística divide o conjunto de dados em cinco partes estratificadas de formas iguais, onde a cada iteração uma parte serve como conjunto de teste enquanto as quatro restantes são utilizadas para treinamento e validação. O procedimento é repetido cinco vezes, garantindo que cada porção dos dados seja testada exatamente uma vez.

A implementação da validação cruzada no contexto da WellNet envolveu a divisão das 10.000 imagens de treinamento selecionadas do dataset em 5 subconjuntos de 2.000 imagens cada. A cada *fold*, 8.000 imagens foram utilizadas para treinamento (com aplicação das técnicas de aumento de dados quando configurado) e 2.000 imagens para teste. Este processo permitiu que todas as imagens do conjunto de treinamento fossem avaliadas pelo menos uma vez como dados de teste, fornecendo uma estimativa mais robusta do desempenho do modelo.

A escolha de 5 *folds* representa um compromisso entre viabilidade computacional e confiabilidade estatística. Embora configurações com mais *folds* (como 10-fold) possam fornecer estimativas mais estáveis com menor variância, o custo computacional para arquiteturas híbridas ViT→CNN torna 5-fold uma escolha prática para o escopo deste trabalho. Considerando que cada *fold* requer treinamento completo do modelo (8-10 épocas conforme observado), a validação 5-fold resultou em aproximadamente 40-50 épocas de treinamento total.

A Figura 7 ilustra o processo de validação cruzada aplicado às 10.000 imagens de treinamento selecionadas.



Figura 7 – Processo da Validação Cruzada.

Esta abordagem minimiza riscos de avaliação enviesada associados à dependência de uma única divisão aleatória dos dados, problema comum em trabalhos que utilizam divisões fixas treino/teste. A validação cruzada fornece múltiplas estimativas de desempenho, permitindo cálculo de médias e desvios-padrão das métricas (acurácia, precisão, *recall*, F1-*score* e área sob a curva ROC), oferecendo maior confiança estatística nos resultados.

Contudo, esta escolha pode introduzir maior variabilidade nos resultados comparada a validações com mais *folds*. Uma abordagem metodológica aplicada foi a estratificação nas divisões dos *folds*, que poderia garantir distribuição equilibrada das classes (rachaduras e fissuras/não-rachaduras e fissuras) em cada *fold*. Divisões não estratificadas podem resultar em *folds* com distribuições desbalanceadas, impactando a estabilidade dos resultados entre iterações.

4.5 Treinamento e Configurações do Modelo

O treinamento da WellNet foi realizado utilizando duas abordagens distintas para analisar o impacto do aumento de dados no desempenho do modelo, como ilustrado na Figura 8:

1. **Treinamento sem aumento de dados**: Nesta abordagem, o modelo foi treinado

utilizando 10.000 imagens originais disponíveis no banco de dados, sem a aplicação de nenhuma técnica de transformação ou incremento de dados. O objetivo era avaliar o desempenho do modelo em condições "puras", isto é, com os dados exatamente como coletados, sem qualquer alteração. Esse tipo de treinamento permite observar a capacidade do modelo de aprender as características intrínsecas dos padrões de rachaduras e fissuras sem influências externas, funcionando como uma linha de base para comparar os resultados obtidos com e sem técnicas de aumento de dados. Embora esta abordagem possa apresentar bons resultados em determinados cenários, é mais suscetível a problemas de sobreajuste (*overfitting*), pois o modelo aprende a memorizar as imagens originais, ficando menos preparado para generalizar quando confrontado com dados diferentes dos que foram utilizados no treinamento.

2. **Treinamento com aumento de dados**: Nesta abordagem, foram aplicadas diversas técnicas de aumento de dados (*Data Augmentation*) nas 10.000 imagens selecionadas, como rotação, reflexão horizontal, ajustes de brilho, contraste e escala, entre outras transformações. O objetivo principal dessas técnicas é gerar variações artificiais das imagens originais, aumentando a quantidade e a diversidade de exemplos que o modelo vê durante o treinamento. Dessa forma, o modelo se torna mais robusto, capaz de lidar com variações comuns em imagens de campo, como diferentes ângulos de captura, sombras, iluminação variável e possíveis distorções. O aumento de dados contribui significativamente para a capacidade de generalização do modelo, pois expõe o modelo a cenários variados, semelhantes aos que ele encontrará no ambiente real. Além disso, essa abordagem reduz a dependência de uma grande quantidade de dados originais, que muitas vezes são difíceis ou custosos de obter, enquanto ajuda a mitigar o sobreajuste, pois o modelo não se limita a "memorizar" exemplos específicos, mas aprende padrões mais generalizáveis.

Como mostrado na Figura 8, o fluxo de treinamento da WellNet começa com a definição das configurações de treinamento pelo usuário, seguida do carregamento das imagens originais do banco de dados. Dependendo da abordagem selecionada, as imagens podem passar por uma etapa de aumento de dados antes de serem encaminhadas para a função de treinamento. Em seguida, o modelo é treinado por várias épocas, e as métricas de desempenho são coletadas e armazenadas para posterior análise comparativa entre as abordagens com e sem aumento de dados.

Essas duas abordagens permitiram uma comparação detalhada do impacto do aumento de dados, tanto na capacidade do modelo de evitar o sobreajuste (*overfitting*) quanto em seu desempenho geral na classificação de rachaduras e fissuras, evidenciando as vantagens do aumento de dados em melhorar a robustez do modelo frente a condições variadas de inspeção.

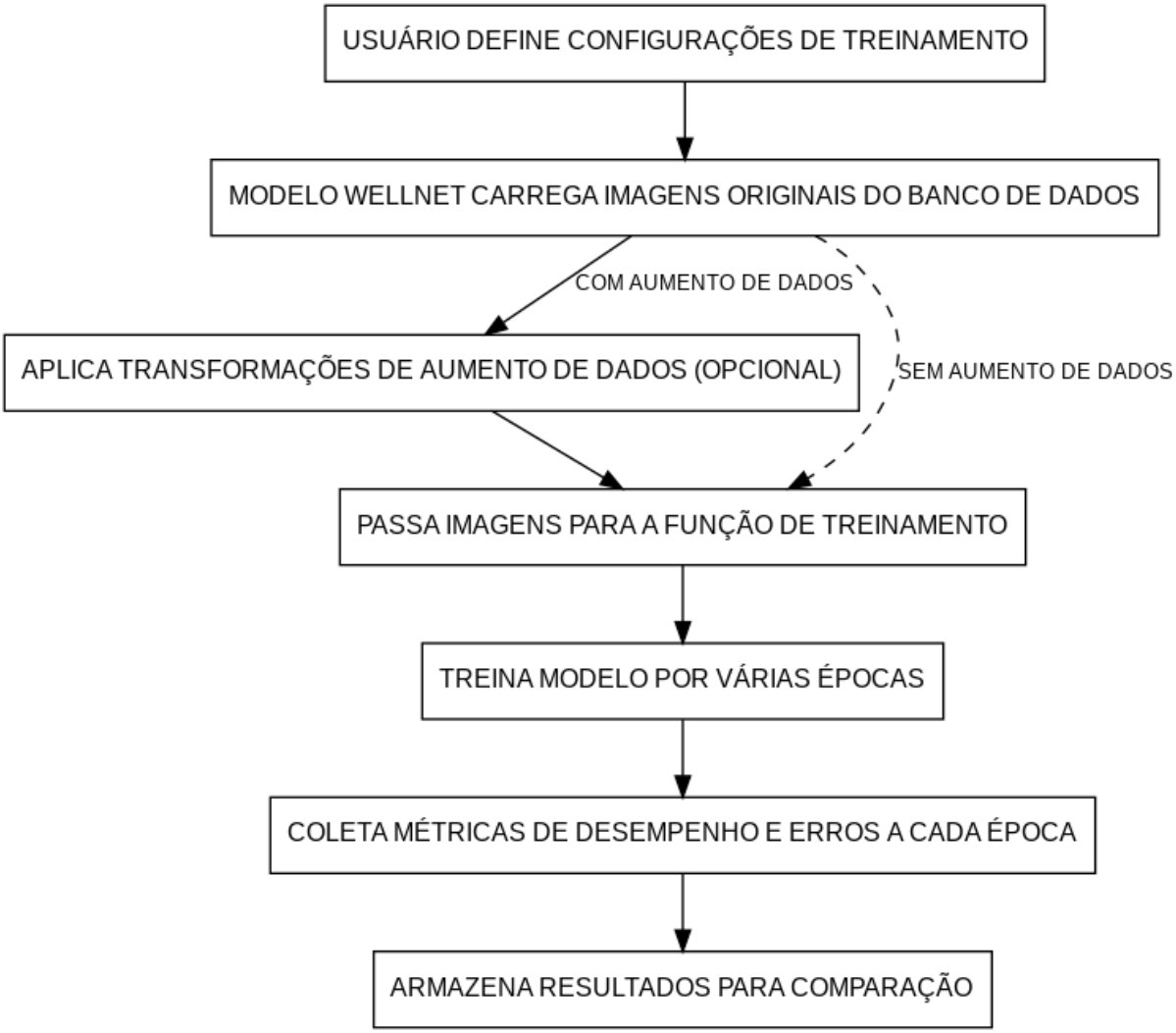


Figura 8 – Diagrama de Fluxo de Treinamento.

4.6 Hiperparâmetros Utilizados

A definição dos hiperparâmetros da WellNet foi realizada através de processo empírico envolvendo muitas tentativas e erros até chegar em uma arquitetura eficiente para a solução do problema. Este processo foi guiado por validação cruzada com 5 *folds* e implementação de *Early Stopping* para otimizar o treinamento e prevenir *overfitting*.

4.6.1 Metodologia de Seleção

Foi aplicado *Early Stopping* durante o treinamento, onde a cada passagem de época a melhor acurácia de validação é comparada sempre com a próxima. Foram utilizados entre 7 a 20 épocas, mas ao analisar os gráficos da acurácia de validação e acurácia de treino, o treinamento estabiliza entre as épocas 8 e 10, dependendo da configuração dos hiperparâmetros. Esta observação empírica orientou a definição do ponto de parada ótimo.

A validação cruzada permitiu avaliar a consistência de cada configuração testada across diferentes divisões dos dados, reduzindo o risco de seleção baseada em partições específicas que poderiam gerar resultados enviesados.

4.6.2 Hiperparâmetros Selecionados

- **Otimizador - Adam**: Selecionado empiricamente após comparações com outros otimizadores. O Adam demonstrou convergência mais estável na arquiteturas ViT→CNN, ajustando dinamicamente a taxa de aprendizado para cada parâmetro do modelo. Esta característica foi importante para a WellNet devido à natureza da arquitetura que combina ViT com CNNs.
- **Função de Erro - NLLLoss (*Negative Log-Likelihood Loss*)**: Foi selecionada já visando ampliação futura para multiclasses. A NLLLoss é utilizada em conjunto com *LogSoftmax*, calculando a perda ao considerar as probabilidades previstas pelo modelo. Esta combinação permite expandir a arquitetura para múltiplas categorias de defeitos estruturais sem modificações na função de perda.
- ***Batch Size***: Testados valores de 32, 64, 96 e 128 empiricamente. A escolha dependeu do equilíbrio entre estabilidade de treinamento e eficiência computacional disponível no *hardware* utilizado. Os valores de 64 e 96 demonstraram melhor desempenho para a arquitetura híbrida da WellNet.
- **Número de Épocas**: Configurado entre 7 a 20 épocas com *Early Stopping*. O treinamento efetivo ocorreu entre 8 a 10 épocas conforme análise dos gráficos de convergência, onde se observou estabilização da acurácia de validação. Este comportamento foi consistente across diferentes configurações de hiperparâmetros testadas.
- **Taxa de *Dropout***: Definida em 0.5 através do processo empírico após testar taxas entre 0.3 a 0.7, onde 0.5 demonstrou os melhores resultados. Esta taxa foi aplicada apenas no momento de treinamento e demonstrou eficácia na prevenção de *overfitting* em arquiteturas que combinam ViTs com CNNs. O *dropout* foi aplicado para aumentar a capacidade de generalização do modelo para novos dados.

4.6.3 Limitações da Metodologia Empírica

A metodologia empírica, embora tenha resultado em configuração funcional, apresenta limitações em termos de garantia de otimalidade. A ausência de métodos sistemáticos como *grid search* ou otimização bayesiana pode ter deixado configurações superiores não exploradas.

A dependência de análise visual dos gráficos de convergência na determinação do ponto de estabilização introduz variabilidade interpretativa no processo. Critérios objetivos

baseados em métricas estatísticas, como tolerância de melhoria mínima, podem aumentar a reprodutibilidade do método.

4.7 *hardware* e Ferramentas Auxiliares

O desenvolvimento da WellNet foi realizado em um **MacBook Air** equipado com um processador **M2**, 8 GB de RAM e um SSD de 256 GB. Embora este *hardware* tenha algumas limitações em termos de capacidade de processamento e memória, especialmente se comparado a sistemas otimizados para tarefas de Aprendizado Profundo, como GPUs (*Graphics Processing Unit*) de última geração, a eficiência da arquitetura da WellNet possibilitou o treinamento mesmo em um dispositivo sem uma GPU dedicada.

Os experimentos foram conduzidos em computadores sem GPU dedicada, utilizando processador **M2** que demonstrou eficiência no processamento de tarefas de Aprendizado Profundo de médio porte. O processamento em CPU possibilitou análise inicial e validação dos modelos com tempo de execução aceitável para os experimentos realizados.

4.8 Impactos da Arquitetura Híbrida

Arquiteturas híbridas tendem a ser mais pesadas computacionalmente que abordagens puramente convolucionais ou baseadas apenas em *Transformers*. Para mitigar esta limitação, foi utilizada uma versão reduzida do *Vision Transformer* chamada ViT *Tiny* pré-treinado, que oferece um compromisso entre capacidade de modelagem e eficiência computacional.

A escolha do ViT *Tiny* pré-treinado representou uma estratégia para reduzir a complexidade computacional da arquitetura híbrida sem comprometer significativamente a capacidade de extração de características. O uso de modelos pré-treinados também acelera a convergência durante o treinamento, reduzindo o tempo necessário para obter resultados adequados.

Para avaliar o impacto desta escolha arquitetural, foram mensurados os tempos de treinamento e inferência da WellNet. O treinamento com 10.000 imagens de 224x224 *pixels*, configurado para 8 épocas, *batch size* de 96 e validação cruzada com 5 *folds*, levou em média 24 minutos no *hardware* utilizado (MacBook Air M2, 8GB RAM).

O tempo de inferência foi validado em duas configurações: individualmente, o modelo apresentou tempo de inferência em torno de 17 milissegundos por imagem. Em teste prático com aplicação web, 100 imagens de um dataset com rachaduras foram processadas em tempo total de 1,66 segundos, resultando em aproximadamente 16,6 milissegundos por imagem, confirmando a consistência dos tempos de inferência.

Estes resultados indicam que a combinação de ViT *Tiny* pré-treinado com CNNs resultou em arquitetura híbrida com complexidade computacional controlada. Os tempos obtidos são compatíveis com aplicações práticas em sistemas embarcados e dispositivos móveis. O tempo de inferência permite processamento de aproximadamente 58-60 imagens por segundo, adequado para inspeções automatizadas em campo.

4.9 Considerações Finais

Foram apresentadas as principais ferramentas, métodos e configurações utilizadas no desenvolvimento da WellNet, uma arquitetura híbrida para classificação de rachaduras e fissuras em superfícies de concreto. As tecnologias empregadas (Python, PyTorch, TorchVision e timm) possibilitaram a implementação eficiente da arquitetura híbrida ViT→CNN, aproveitando modelos pré-treinados e técnicas avançadas de *Deep Learning*. O dataset (Özgenel, 2019) foi utilizado como base, selecionando 20.000 imagens para treinamento e validação do modelo.

O pré-processamento dos dados constituiu etapa fundamental, incluindo redimensionamento padronizado para 224x224 *pixels* (compatível com a arquitetura ViT) e aplicação sistemática de técnicas de aumento de dados. As transformações implementadas (rotação, reflexão horizontal, ajustes de brilho e contraste) ampliaram artificialmente a variabilidade do conjunto de treinamento, contribuindo para minimizar *overfitting* e melhorar a capacidade de generalização em condições variadas de captura. A validação cruzada com 5 *folds* assegurou avaliação robusta e estatisticamente confiável, utilizando diferentes combinações de dados de treino e teste para reduzir viés de seleção.

O treinamento comparativo entre abordagens com e sem aumento de dados demonstrou empiricamente que as técnicas de transformação contribuíram para melhor generalização do modelo. A seleção empírica de hiperparâmetros envolveu configuração cuidadosa do otimizador *Adam*, função *NLLLoss* combinada com *LogSoftmax* (visando futura expansão multiclasse), taxa de *dropout* 0.5 aplicada apenas durante treinamento, e *batch sizes* entre 64-96. O processo foi realizado através de múltiplas tentativas iterativas com implementação de *Early Stopping*, resultando em convergência estável entre 8 a 10 épocas.

A implementação em *hardware* com recursos limitados (MacBook Air M2, 8GB RAM) validou a viabilidade de treinamento da WellNet para dispositivos de baixa capacidade computacional. A escolha estratégica do ViT *Tiny* pré-treinado como componente da arquitetura híbrida permitiu equilibrar capacidade de modelagem com eficiência computacional. Os tempos de execução obtidos (24 minutos para treinamento completo com validação cruzada, 16,6ms por imagem na inferência) demonstram adequação para aplicações práticas em dispositivos móveis e inspeções automatizadas em tempo real.

Parte II

Proposta

5 Sistema Proposto

Neste capítulo, apresentaremos a proposta de criação de uma arquitetura de Rede Neural para a classificação automatizada de imagens de rachaduras em estruturas de concreto, utilizando a rede WellNet, uma arquitetura híbrida que combina *Vision Transformers* (ViT) e camadas Convolucionais das Redes Neurais Convolucionais (CNNs) como ajuste fino. A WellNet foi projetada para aproveitar a capacidade dos ViTs de capturar relações globais nas imagens e a especialização das CNNs em extrair características locais, oferecendo uma abordagem precisa para a tarefa de classificação de imagens de rachaduras.

5.1 Primeira Parte do Sistema Proposto: A arquitetura

O desenvolvimento da WellNet envolve o treinamento do modelo utilizando um conjunto de dados composto por imagens de rachaduras estruturas de concreto, contendo exemplos de rachaduras e superfícies intactas(sem rachaduras). Para melhorar a capacidade de generalização do modelo e reduzir o risco de sobreajuste, foram aplicadas técnicas de aumento de dados em alguns dos experimentos e validação cruzada durante o treinamento. O modelo resultante foi avaliado por meio de métricas como acurácia, precisão, *recall* e *F1-score*, assegurando que a abordagem proposta esteja à altura das soluções existentes na literatura.

Além do treinamento do modelo, um protótipo funcional foi desenvolvido para demonstrar a aplicabilidade prática da WellNet. Esse protótipo consiste em uma interface web que permite aos usuários submeterem imagens de superfícies e obterem uma classificação binária automática sobre a presença ou ausência de rachaduras. A interface foi pensada para ser intuitiva e de fácil uso, facilitando a integração da tecnologia proposta em processos de monitoramento estrutural por profissionais da construção civil, bem como por usuários que não possuem conhecimento técnico aprofundado.

O sistema também visa à possibilidade de viabilidade de ser implementado em dispositivos com recursos computacionais limitados, como drones e câmeras móveis, permitindo que a classificação de rachaduras possa ocorrer de maneira descentralizada e automatizada, mesmo em locais de difícil acesso. Esta flexibilidade aumenta o potencial de utilização da WellNet, tornando-a uma ferramenta valiosa tanto para grandes empresas de engenharia quanto para pequenas construtoras e gestores de manutenção predial.

5.2 Arquitetura WellNet

A WellNet é uma arquitetura de Rede Neural projetada especificamente para a classificação de imagens de rachaduras em superfícies de concreto, combinando componentes de *Vision Transformers* (ViT) e *Convolutional Neural Networks* (CNN). O nome **WellNet** reflete uma junção do nome do autor **Well** (de Wellington) com ***Networks***, simbolizando a intenção de criar uma rede neural personalizada e otimizada para este problema específico de classificação. A WellNet foi desenvolvida para tirar proveito da capacidade dos ViTs de capturar dependências globais e contextuais nas imagens, além de complementar esse aprendizado com a capacidade das CNNs de extrair características locais, como bordas e texturas, que são cruciais para identificar imagens de rachaduras.

A arquitetura da WellNet é organizada em camadas, divididas em três blocos principais: o bloco inicial de *Vision Transformer* (ViT) como *backbone* extrator de características, o bloco de camadas convolucionais e *pooling*, e o bloco de camadas densas *(Fully Connected)*. A escolha do ViT como *backbone* e bloco inicial visa explorar a sua habilidade de capturar padrões amplos e características globais, aproveitando um modelo ViT-*Tiny* pré-treinado que já possui uma vasta base de conhecimento sobre imagens complexas. Este *backbone* ViT utiliza 5.486.016 parâmetros congelados, representando 88,92% do total de parâmetros da arquitetura, garantindo extração robusta de características semânticas globais. As camadas convolucionais foram escolhidas para complementar esse aprendizado ao focar em padrões locais, fundamentais para a detecção de texturas típicas de rachaduras, garantindo assim uma combinação robusta entre análise global e detalhamento local. Por fim, as camadas densas são responsáveis pela etapa final de classificação, proporcionando a decisão mais precisa baseada em todas as características extraídas.

A Figura 9 ilustra a arquitetura completa da WellNet, destacando cada um dos blocos descritos e evidenciando a contribuição específica de cada etapa na tarefa de classificação das rachaduras. Esta abordagem híbrida combina as melhores características das arquiteturas ViT e CNN para entregar um desempenho elevado na classificação de rachaduras.

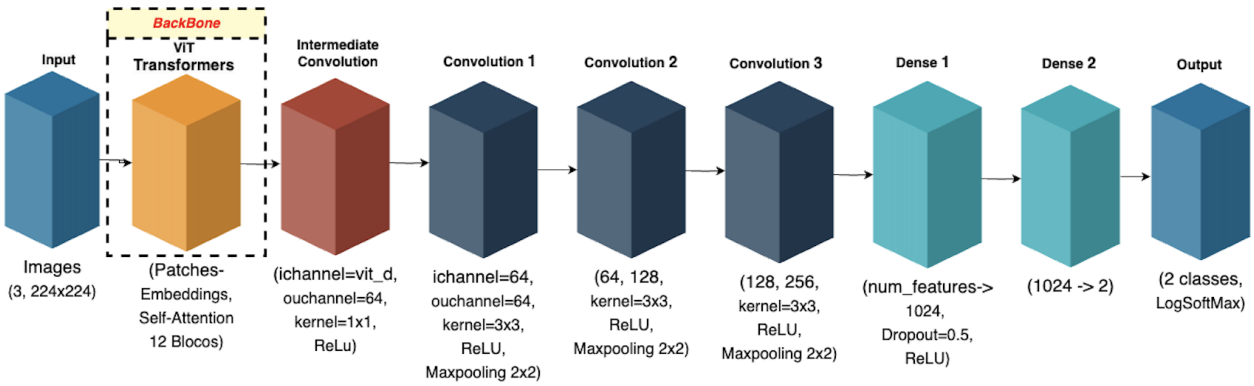


Figura 9 – Arquitetura da WellNet.

5.2.1 Camada de Entrada

- **Dimensão**: 224 x 224 *pixels*.
- **Descrição**: A imagem de entrada é redimensionada para 224 x 224 *pixels* utilizando a função `transforms.Resize()` do PyTorch, que aplica interpolação bilinear para manter a qualidade visual durante o redimensionamento. Essa padronização de dimensões é obrigatória pois a arquitetura da WellNet, especificamente o componente ViT *backbone*, foi projetada para receber entradas de tamanho fixo de 224×224 *pixels*. O ViT divide a imagem em *patches* de 16×16 *pixels*, resultando em exatamente 196 *patches* (14×14), que são processados como uma sequência. Independentemente do tamanho original da imagem capturada (que pode variar entre diferentes dispositivos e condições de campo), o redimensionamento garante uniformidade no processamento e compatibilidade com os pesos pré-treinados da rede.

5.2.2 Bloco *Vision Transformers*: *Backbone* (Extrator de Características)

O bloco ViT desempenha um papel fundamental na WellNet como *backbone* principal, atuando como extrator primário de características globais e contextuais das imagens. Este *backbone* utiliza o modelo ViT-*Tiny* pré-treinado com 5.486.016 parâmetros congelados (88,92% do total de parâmetros), fornecendo capacidade robusta de extração de características semânticas através de representações pré-aprendidas em grandes conjuntos de dados. Abaixo estão as etapas detalhadas que compõem esse *backbone*:

- ***Patches-Embedding***: Inicialmente, o ViT divide a imagem de entrada em pequenos *patches* de 16 x 16 *pixels*. Essa divisão facilita o processamento das informações visuais em pequenas unidades, permitindo que o modelo analise a imagem em partes menores e compreensíveis. Cada um dos *patches* gerados é transformado em um vetor de características utilizando camadas de *Embedding* (DOSOVITSKIY et al., 2021). Essa transformação resulta em uma representação vetorial, que permite ao *backbone* processar as características extraídas dos *patches* como sequências numéricas para identificar padrões.
- **Blocos de *Self-Attention***: O ViT utiliza 12 blocos de *Self-Attention* (DOSOVITSKIY et al., 2021) para capturar as dependências globais entre diferentes regiões da imagem. Esse mecanismo de atenção ajuda o extrator de características a identificar como partes distintas e potencialmente distantes da imagem se relacionam, permitindo ao modelo capturar informações contextuais que, normalmente, seriam difíceis de serem identificadas por camadas convolucionais. A capacidade de *Self-Attention* do ViT possibilita que ele detecte padrões e correlações importantes, mesmo em situações onde as características visuais estão espalhadas pela imagem.

- **Saída do *Backbone***: A saída do ViT é um vetor de características abstratas de dimensão 192, que representa a imagem em termos de suas características globais e contextuais extraídas através dos mecanismos de atenção. Essa representação é essencial para fornecer uma visão abrangente da imagem, preparando os dados para serem refinados pelas camadas convolucionais da WellNet que atuam em seguida. O *backbone* ViT mantém seus parâmetros congelados durante o treinamento, preservando as representações semânticas robustas aprendidas durante o pré-treinamento.

5.2.3 Camada de Convolução Intermediária - Conversão

A camada de convolução intermediária desempenha um papel importante e crucial na integração do ViT com as camadas convolucionais subsequentes. Esta camada é responsável por ajustar a saída linear do ViT para um formato compatível com operações convolucionais, permitindo que a WellNet se beneficie das capacidades tanto dos ViTs quanto das CNNs.

- **Função**: A principal função desta camada é reorganizar a saída do ViT, que consiste em um vetor linear unidimensional, para um formato tridimensional compatível com as operações convolucionais. O ViT processa imagens como sequências de *patches* e gera representações em formato vetorial plano. Para que essas representações possam ser processadas pelas camadas convolucionais subsequentes da WellNet, é necessário reestruturar esses dados em uma configuração espacial bidimensional com múltiplos canais. Essa transformação atua como uma ponte entre a arquitetura de *Transformer* (que trabalha com sequências) e a arquitetura convolucional (que opera sobre mapas de características espaciais), preservando a informação semântica global capturada pelo ViT enquanto a torna acessível para a extração de características locais através de convoluções.

- **Dimensão**: Após a reorganização, a dimensão da saída é transformada para (192, 14, 14), onde cada componente possui significado específico. O valor 192 representa o número de canais ou mapas de características, equivalente à profundidade do volume de dados, onde cada canal captura diferentes aspectos das características aprendidas pelo ViT. As dimensões 14x14 correspondem à resolução espacial da representação reorganizada, indicando que a imagem foi dividida em uma grade de 196 posições espaciais (14×14). Essa resolução é resultado do processamento dos *patches* pelo ViT: considerando uma imagem de entrada de 224×224 *pixels* dividida em *patches* de 16×16 *pixels*, obtém-se 196 *patches* dispostos em uma grade 14×14, preservando assim a correspondência espacial original da imagem de forma compacta.

- **Camada Convolucional**: Para ajustar a compatibilidade entre o ViT e o bloco convolucional subsequente, uma camada convolucional é aplicada, utilizando 64

filtros de tamanho 1x1. Essa camada convolucional serve como um ponto de ajuste, garantindo que a transição entre os dois blocos seja **suave**, além de agregar uma nova representação dos dados sem modificar a dimensão espacial.

- **Função de Ativação**: Utilizamos a função de ativação ReLU (Rectified Linear Unit), que é aplicada após a convolução. A função ReLU introduz não-linearidade ao modelo, permitindo que ele aprenda padrões mais complexos e diversos. A ativação ReLU também ajuda a prevenir problemas de gradiente desaparecendo, o que torna o treinamento mais eficiente.

5.2.4 Justificativa da necessidade Arquitetural da Camada Intermediária

Embora seja tecnicamente possível conectar diretamente os 192 canais de saída do ViT à primeira camada convolucional da CNN, a validação experimental demonstrou que esta abordagem resulta em degradação significativa da performance. A camada intermediária não constitui apenas uma "conveniência técnica", mas representa uma escolha arquitetural fundamentada com benefícios quantificáveis e validados empiricamente.

5.2.4.0.1 Análise Comparativa de Abordagens Arquiteturais

Tabela 3 – Comparação entre abordagens arquiteturais diretas e decompostas

Abordagem	**Configuração**	**Parâmetros**	**Performance**
Direta	Conv1: 192→64 (3×3)	110.656	Reduzida
Decomposta	Conv 1×1: 192→64 + Conv1: 64→64 (3×3)	49.280	Superior
Eficiência	**Redução Paramétrica**	**55,5%**	**+2,60 pp**

5.2.4.0.2 Validação Experimental

Durante o desenvolvimento da WellNet, foi realizado experimento controlado comparando as duas abordagens arquiteturais. A implementação direta da primeira camada convolucional com 192 canais de entrada (`nn.Conv2d(192, 64, kernel_size=3)`) resultou em degradação mensurável da performance, conforme evidenciado pela redução nas métricas de avaliação. Esta evidência empírica confirma que a decomposição arquitetural não é meramente uma otimização opcional, mas uma necessidade para maximizar a eficácia da integração ViT→ CNN.

5.2.4.0.3 Análise de Eficiência Paramétrica

A decomposição arquitetural oferece vantagens quantificáveis em termos de eficiência paramétrica:

$$\text{Parâmetros}_{\text{direto}} = 192 \times 64 \times 3 \times 3 + 64 = 110.656 \tag{5.1}$$

$$\text{Parâmetros}_{\text{intermediária}} = 192 \times 64 \times 1 \times 1 + 64 = 12.352 \tag{5.2}$$

$$\text{Parâmetros}_{\text{Conv1}} = 64 \times 64 \times 3 \times 3 + 64 = 36.928 \tag{5.3}$$

$$\text{Parâmetros}_{\text{total}} = 12.352 + 36.928 = 49.280 \tag{5.4}$$

A redução paramétrica de 55,5% ((110.656 - 49.280) / 110.656) demonstra que a decomposição é não apenas funcionalmente superior, mas também computacionalmente mais eficiente.

5.2.4.0.4 Fundamentação Teórica da Separação de Responsabilidades

A decomposição implementa o princípio da separação de responsabilidades, onde:

- **Camada Intermediária (1×1)**: Realiza seleção e combinação linear otimizada das 192 *features* semânticas do ViT, implementando projeção matemática $f : \mathbb{R}^{192} \to \mathbb{R}^{64}$ preservando informação espacial.
- **Conv1 (3×3)**: Especializa-se na extração de padrões espaciais locais dos 64 canais otimizados, focando exclusivamente na detecção de características geométricas e texturais.

Esta separação facilita a convergência durante o treinamento, pois cada componente otimiza uma tarefa específica ao invés de tentar resolver simultaneamente tanto a seleção de *features* quanto a extração de padrões espaciais.

5.2.4.0.5 Justificativa da Implementação via Conv2d

A implementação `nn.Conv2d(192, 64, kernel 1x1)` ao invés de `nn.Linear(192, 64)` por razões técnicas fundamentais:

- **Compatibilidade Dimensional**: A saída do ViT reorganizada possui formato (batch_size, 192, 14, 14), incompatível com camadas lineares que esperam formato (batch_size, features).
- **Preservação Espacial**: A convolução 1×1 preserva a correspondência espacial pixel-a-pixel, essencial para as operações convolucionais subsequentes.
- **Equivalência Matemática**: Embora matematicamente equivalente à aplicação de transformação linear independente em cada posição espacial, a implementação convolucional mantém a estrutura de dados necessária para a CNN.

5.2.5 Bloco de Camadas Convolucionais

O bloco de camadas convolucionais da WellNet é inspirado na arquitetura AlexNet, com o objetivo de extrair características locais importantes da imagem de entrada. Ele é composto por três camadas convolucionais seguidas de camadas de MaxPooling, que se alternam para capturar características em diferentes níveis de abstração, reduzindo a dimensionalidade e mantendo as informações mais relevantes. Esta configuração visa maximizar a extração de padrões locais, como bordas, texturas e formas complexas, essenciais para a tarefa de classificação de rachaduras.

- **Camada Convolucional 1**:
  - **Filtros**: A primeira camada convolucional utiliza 64 filtros de tamanho 3x3, com o objetivo de capturar padrões básicos, como arestas e texturas simples. O uso de filtros pequenos permite uma análise mais precisa dos detalhes locais na imagem.
  - **Função de Ativação**: Após a aplicação dos filtros, é utilizada a função de ativação ReLU *(Rectified Linear Unit)*. A função ReLU é fundamental para introduzir não-linearidade ao modelo, permitindo que ele seja capaz de aprender padrões complexos e diversos.
  - **Pooling**: A camada convolucional é seguida por uma camada de *MaxPooling* 2x2, que reduz a dimensionalidade espacial em 50%. O *MaxPooling* é utilizado para manter apenas as características mais salientes, reduzindo a quantidade de parâmetros e ajudando a minimizar o risco de sobreajuste.
- **Camada Convolucional 2**:
  - **Filtros**: A segunda camada convolucional aumenta a capacidade de extração de características usando 128 filtros de tamanho 3x3. Este aumento no número de filtros permite capturar características mais complexas e detalhadas da imagem, fornecendo uma representação mais robusta.
  - **Função de Ativação**: A função ReLU é novamente aplicada após a convolução, mantendo a consistência do aprendizado não-linear e permitindo ao modelo adaptar-se a variações mais sutis nos dados.
  - **Pooling**: Após a convolução, uma camada de *MaxPooling* 2x2 é aplicada. Esta camada continua o processo de redução da dimensionalidade, garantindo que apenas as características mais relevantes sejam mantidas para as etapas posteriores.
- **Camada Convolucional 3**:

- **Filtros**: A terceira camada convolucional utiliza 256 filtros de tamanho 3x3. Este número elevado de filtros é crucial para capturar características altamente complexas que são importantes para distinguir entre classes, como pequenas rachaduras nos mapas de características.
- **Função de Ativação**: A função ReLU é novamente aplicada, garantindo que a rede possa aprender padrões não-lineares e adaptáveis que são necessários para a classificação de rachaduras.
- **Pooling**: O bloco finaliza com uma camada de *MaxPooling* 2x2, que visa consolidar as características extraídas, reduzindo a dimensionalidade para uma representação mais manejável, enquanto mantém informações críticas para a classificação. Esta última de *pooling* também contribui para aumentar a invariância a pequenas translações e variações na imagem, ou seja, a capacidade do modelo de reconhecer e classificar o mesmo objeto de forma consistente mesmo quando ele aparece em posições ligeiramente diferentes ou com pequenas variações na imagem.

5.2.6 Bloco de Camadas Densas *(Fully Connected)*

O bloco de camadas densas *(Fully Connected)* da WellNet é responsável por combinar e classificar as características extraídas nas etapas anteriores, agregando informações globais para fornecer uma previsão precisa da classe. Este bloco consiste em duas camadas densas que realizam a combinação dos recursos extraídos pelas camadas convolucionais e pelo ViT, além de uma camada de saída final para a classificação binária. Abaixo detalhamos as funções e características de cada uma das camadas deste bloco.

- **Camada Densa 1 (FC1)**:
  - **Neurônios**: A primeira camada densa contém 1024 neurônios, responsáveis por combinar as características extraídas dos blocos convolucionais e do ViT. Este número de neurônios foi escolhido durante os experimentos empíricos para proporcionar uma representação de alta dimensão, permitindo que o modelo consiga capturar relações complexas entre as características aprendidas até este ponto.
  - **Função de Ativação**: A função de ativação utilizada é a ReLU. Esta função de ativação é ideal para camadas densas devido à sua simplicidade computacional e eficácia em introduzir não-linearidade, permitindo que o modelo seja capaz de aprender relações mais complexas entre as características.
  - **Dropout**: Para evitar o *overfitting*, é aplicado um mecanismo de *Dropout* com taxa de 0.5. Este processo desativa aleatoriamente metade dos neurônios durante

o treinamento, impedindo que o modelo dependa excessivamente de determinados caminhos específicos, o que contribui para uma melhor generalização dos resultados.

- **Camada Densa 2 (FC2):**
  - **Neurônios**: A segunda camada densa possui 2 neurônios, representando as classes do problema de classificação: rachaduras e não-rachaduras. Esta camada é responsável por mapear as características extraídas pelas camadas anteriores para o espaço de decisão final, facilitando a escolha da classe correta com base nos pesos aprendidos.
  - **Função de Saída**: Cada neurônio produz um valor escalar que representa a pontuação (*score*) para sua respectiva classe. O primeiro neurônio corresponde à classe "não-rachaduras" e o segundo à classe "rachaduras", permitindo que o modelo expresse sua confiança na classificação através da magnitude relativa desses valores.
  - **Transformação Final**: Os valores de saída dos 2 neurônios são posteriormente processados pela função *LogSoftmax* na camada seguinte, convertendo as pontuações brutas em probabilidades logarítmicas normalizadas. Esta sequência permite interpretação probabilística dos resultados e otimização eficiente durante o treinamento.
  - **Parâmetros**: A camada FC2 contém 2.050 parâmetros (1.024×2 pesos + 2 *bias*), representando a menor quantidade de parâmetros treináveis na arquitetura. Apesar do tamanho reduzido, esta camada é crítica para o desempenho final, pois concentra toda a informação processada pelas camadas anteriores em decisão binária.
- **Camada de Saída (*LogSoftmax*)**:
  - **Classes**: A camada de saída processa as duas classes do problema: "rachadura" e "não rachadura".
  - **Função de Ativação**: A função *LogSoftmax* gera probabilidades logarítmicas para cada classe, proporcionando interpretação direta dos resultados como pontuação de probabilidade. Esta função facilita o cálculo da perda durante o treinamento, especialmente quando combinada com a função de erro NLLLoss (*Negative Log Likelihood Loss*), que se adapta ao formato logarítmico das saídas.
  - **Vantagens**: O uso de *LogSoftmax* oferece estabilidade numérica superior comparado ao Softmax convencional, evitando problemas de *overflow* em exponenciais e proporcionando gradientes mais estáveis durante o *Backpropagation*. A combinação *LogSoftmax* + NLLLoss é matematicamente equivalente ao uso direto de *CrossEntropyLoss*, mas com melhor controle numérico.

5.2.7 Parâmetros da Arquitetura WellNet

Na Tabela 4, apresentamos a quantidade de parâmetros da arquitetura WellNet, que mostra o número de parâmetros que foram treinados e os parâmetros não-treináveis que foram congelados através do aprendizado por transferência do ViT. A arquitetura incorpora tanto camadas convolucionais quanto componentes transformer, com um total de 683.522 parâmetros treináveis e 5.486.016 parâmetros não-treináveis.

Tabela 4 – Resumo das Camadas e Parâmetros da Arquitetura WellNet

Camada	**Treináveis**	**Não-Treináveis**
Entrada (3, 224x224)	0	0
ViT - *embeddings*	0	147.648
ViT - Self-Attention (12 blocos)	0	5.338.368
Camada Intermediária (Conv 1x1)	12.352	0
Conv1 (3x3, 64 filtros)	36.928	0
Conv2 (3x3, 128 filtros)	73.856	0
Conv3 (3x3, 256 filtros)	295.168	0
Totalmente Conectada (FC1: 256 $\rightarrow$ 1024)	263.168	0
Totalmente Conectada (FC2: 1024 $\rightarrow$ 2)	2.050	0
Camada de Saída (Softmax)	0	0
Total	**683.522**	**5.486.016**

A distribuição dos parâmetros revela aspectos importantes da arquitetura proposta. A maior parte dos parâmetros não-treináveis (5.338.368) corresponde aos 12 blocos de *Self-Attention* do ViT, que foram mantidos congelados para preservar as representações aprendidas durante o pré-treinamento em grandes conjuntos de dados. Os *Patch-Embeddings* do ViT (147.648 parâmetros) também foram congelados, garantindo a preservação das características de representação espacial aprendidas.

Do lado dos parâmetros treináveis, as camadas convolucionais (Conv1, Conv2 e Conv3) representam a maior parte com 405.952 parâmetros, seguidas pelas camadas totalmente conectadas com 265.218 parâmetros. A camada intermediária de convolução 1x1 (12.352 parâmetros) serve como uma ponte eficiente entre os componentes ViT e CNN, permitindo a adaptação dimensional necessária para a fusão das características.

5.3 Análise dos Mapas de Características da Arquitetura WellNet

5.3.1 Extração de Características pelo Vision Transformer

A análise dos mapas de características revela o comportamento hierárquico da arquitetura WellNet ao processar imagens de rachaduras. Como mostrado na Figura 10, o

ViT divide a imagem de entrada de 224×224 *pixels* em 196 patches organizados em um grid de 14×14, onde cada patch é codificado em um vetor de 192 dimensões.

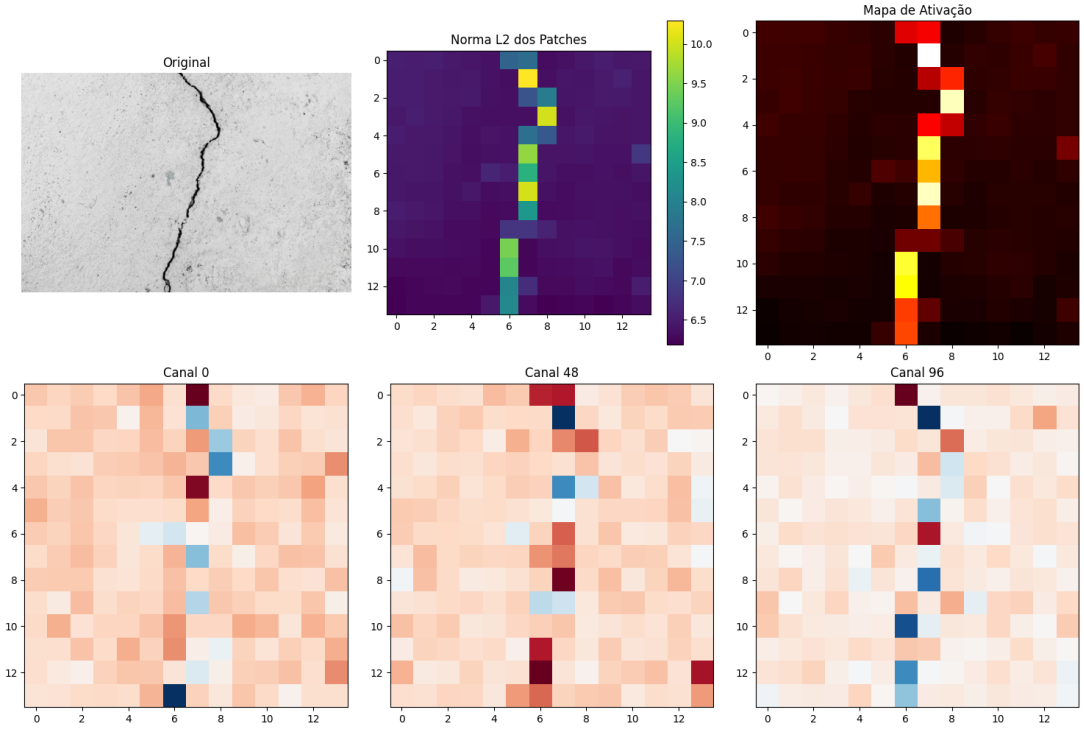


Figura 10 – Análise dos patches do *Vision Transformer*: (a) Imagem original mostrando uma rachadura vertical em superfície de concreto; (b) Norma L2 dos patches, onde valores mais altos (amarelo e verde) indicam maior magnitude de ativação, claramente delineando a trajetória da rachadura; (c) Mapa de ativação mostrando a intensidade de resposta do ViT, com forte ativação ao longo da rachadura; (d-f) Visualização dos canais 0, 48 e 96 dos *embedding*s dos patches, demonstrando diferentes padrões de codificação espacial.

A análise da Figura 10 revela aspectos do processamento, demonstrando como o *Vision Transformer* divide a imagem em patches e processa informações espaciais. Os diferentes componentes visualizados (norma L2, mapas de ativação e canais de *embedding*) ilustram a capacidade hierárquica de extração de características para identificação de rachaduras:

- **Norma L2 dos Patches (b):** A visualização mostra claramente que o ViT identifica a rachadura como uma estrutura diagonal contínua. Os *patches* com maior norma L2 (valores entre 9.5 e 10.5, representados em amarelo e verde) formam uma linha diagonal que corresponde exatamente à localização da rachadura na imagem original. *Patches* de fundo apresentam valores consistentemente baixos (6.5-7.0, em roxo escuro), demonstrando a capacidade discriminativa do modelo.

- **Mapa de Ativação (c):** O mapa de calor confirma que o ViT concentra sua "atenção" na região da rachadura, com ativações máximas formando um padrão diagonal claro.

Esta representação sugere que o modelo aprendeu a identificar descontinuidades lineares características de rachaduras estruturais.

- **Canais de *embedding* (d-f):** Os três canais visualizados (0, 48 e 96) mostram padrões de codificação distintos. O canal 0 apresenta um padrão diagonal pronunciado, o canal 48 mostra uma representação mais difusa, e o canal 96 exibe uma codificação complementar. Esta diversidade sugere que diferentes dimensões do *embedding* capturam aspectos variados da estrutura da rachadura.

5.3.2 Transição ViT para Camadas Convolucionais

A transição entre a saída do ViT e as camadas convolucionais representa um ponto crítico na arquitetura, onde representações baseadas em patches são transformadas em mapas de características espaciais convencionais. A Figura 11 ilustra este processo de transformação.

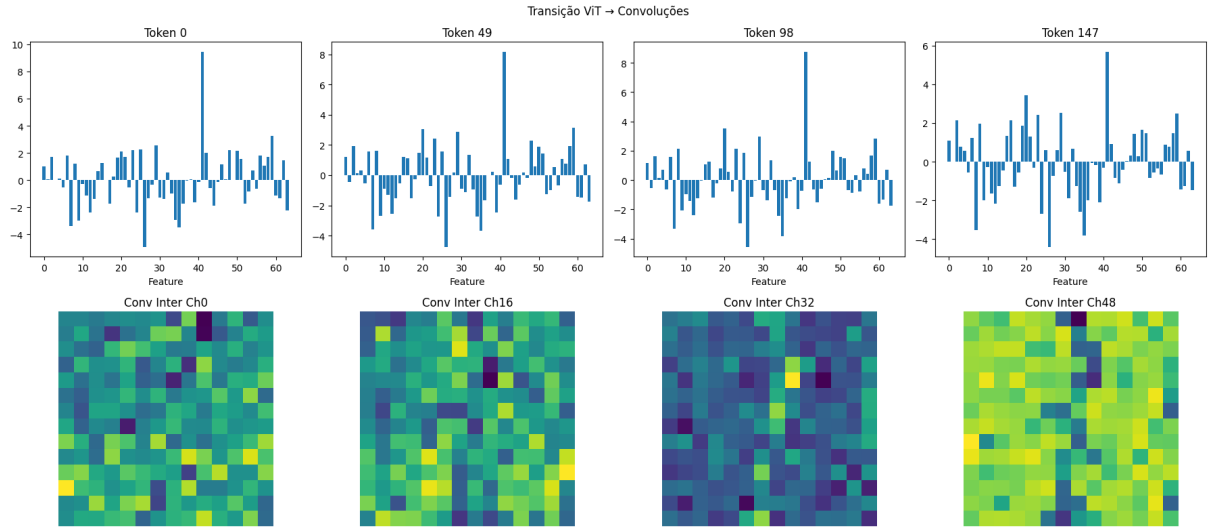


Figura 11 – Transição entre *Vision Transformer* e camadas convolucionais: (Superior) Distribuição de *features* para *Tokens* selecionados (0, 49, 98, 147) mostrando as primeiras 64 dimensões de cada token. O *Token* 49 apresenta pico pronunciado na *feature* 42, indicando especialização. (Inferior) Mapas de características da camada convolucional intermediária para os canais 0, 16, 32 e 48, demonstrando como os *embeddings* dos *Tokens* são reorganizados em representações espaciais 2D.

A análise dos *Tokens* revela padrões importantes na representação espacial das características extraídas pelo ViT, fornecendo *insights* sobre como diferentes regiões da imagem são processadas e codificadas. Cada *token* corresponde a um *patch* específico da imagem dividida em *grid* 14×14, resultando em 196 *Tokens* totais que capturam informações espaciais distribuídas:

- ***Token* 0 (canto superior esquerdo):** Localizado na posição (0,0) do *grid*, apresenta distribuição relativamente uniforme com variações moderadas nas primeiras 64 dimensões do *embedding*. Este padrão é típico de regiões de fundo homogêneo, onde não há características estruturais proeminentes. A ausência de picos significativos indica que esta região não contém informações relevantes para classificação de rachaduras, comportando-se como *baseline* para comparação com *Tokens* mais ativos.

- ***Token* 49 (região central):** Posicionado aproximadamente no centro da imagem (posição 7×7 no *grid* 14×14), exibe pico dramático na *feature* 42 com valor aproximado de 8, contrastando significativamente com as demais dimensões. Este pico sugere que o *token* detectou característica específica e crítica da rachadura, possivelmente relacionada a descontinuidade ou padrão textural característico. *Tokens* centrais frequentemente capturam informações mais relevantes devido ao bias posicional comum em imagens estruturais, onde defeitos tendem a ser mais visíveis nas regiões centrais.

- ***Token* 98 (região média-inferior):** Localizado na porção inferior-central da imagem, mostra padrão notavelmente similar ao Token 49, apresentando pico na mesma característica (*feature*) 42 com magnitude comparável. Esta consistência indica detecção coordenada de características da rachadura ao longo de sua trajetória espacial, demonstrando que o ViT consegue rastrear continuidade estrutural através de *Tokens* adjacentes. A replicação do padrão confirma que a *feature* 42 se especializou na codificação de características específicas de rachaduras.

- ***Token* 147 (região inferior):** Posicionado na área inferior da imagem, apresenta distribuição mais complexa caracterizada por múltiplos picos menores distribuídos através das dimensões de *embedding*. Este padrão multifacetado possivelmente captura terminação da rachadura ou variações locais na estrutura, onde a morfologia pode diferir da região central. A presença de múltiplos picos sugere codificação de informações mais sutis e variadas, refletindo complexidade visual maior nesta região específica.

- **Análise Comparativa dos *Tokens*:** A comparação entre *Tokens* revela especialização espacial clara, onde *Token* 0 (fundo) mantém ativação baixa e uniforme, *Tokens* 49 e 98 (região da rachadura) apresentam ativação focal intensa na *feature* 42, e Token 147 (terminação) mostra ativação distribuída. Esta diferenciação demonstra que o ViT desenvolveu representações espacialmente conscientes, onde diferentes regiões da imagem são processadas de acordo com sua relevância para a tarefa de classificação.

- **Significado da *feature* 42:** O pico consistente na *feature* 42 entre *Tokens* relacionados à rachadura sugere que esta dimensão específica do *embedding* se especializou

na detecção de padrões característicos de rachaduras. Esta especialização emergiu durante o treinamento, onde a rede aprendeu a alocar dimensões específicas para codificar aspectos críticos da tarefa de classificação.

Os mapas da camada convolucional intermediária (parte inferior da visualização) demonstram como os 192 canais do ViT são projetados para 64 canais através de convoluções 1×1, mantendo a resolução espacial de 14×14. Esta transformação dimensional serve como ponte arquitetural entre os componentes *Transformers* e CNN, realizando redução de dimensionalidade controlada que preserva informações espaciais críticas.

A convolução 1×1 atua como projeção linear aplicada independentemente a cada posição espacial, combinando as 192 dimensões de entrada em 64 dimensões de saída através de combinação ponderada. Esta operação permite que a rede aprenda quais aspectos das representações do ViT são mais relevantes para as camadas convolucionais subsequentes, funcionando como mecanismo de seleção e fusão de características. A manutenção da resolução 14×14 assegura que a correspondência espacial entre *Tokens* do ViT e posições dos mapas de características convolucionais seja preservada, facilitando processamento hierárquico consistente.

5.3.3 Análise Hierárquica das Camadas Convolucionais

A Figura 12 apresenta uma visão abrangente das três camadas convolucionais principais, mostrando 16 canais representativos de cada camada.

Tabela 5 – Análise quantitativa detalhada das camadas convolucionais

Camada	**Canais**	**Resolução**	**Redução**	**Min**	**Max**	**ReLU Ativo**
Conv1	64	14×14	1×	-5.305	2.156	38.5%
Conv2	128	7×7	2×	-8.992	4.710	24.6%
Conv3	256	3×3	4.67×	-13.314	12.034	25.0%

A Tabela 5 apresenta métricas quantitativas que revelam o comportamento hierárquico das camadas convolucionais. A progressão de 64 para 256 canais demonstra aumento na capacidade representacional, enquanto a redução espacial de 14×14 para 3×3 indica abstração crescente. Os valores mínimos e máximos mostram amplitude crescente das ativações (-5.305 a 2.156 na Conv1 versus -13.314 a 12.034 na Conv3), indicando especialização progressiva dos neurônios. A métrica "ReLU Ativo" representa a porcentagem de neurônios com ativações positivas, revelando padrões de esparsidade nas representações.

A progressão através das camadas revela padrões hierárquicos de processamento:

Conv1 - Detecção de Características de Baixo Nível:

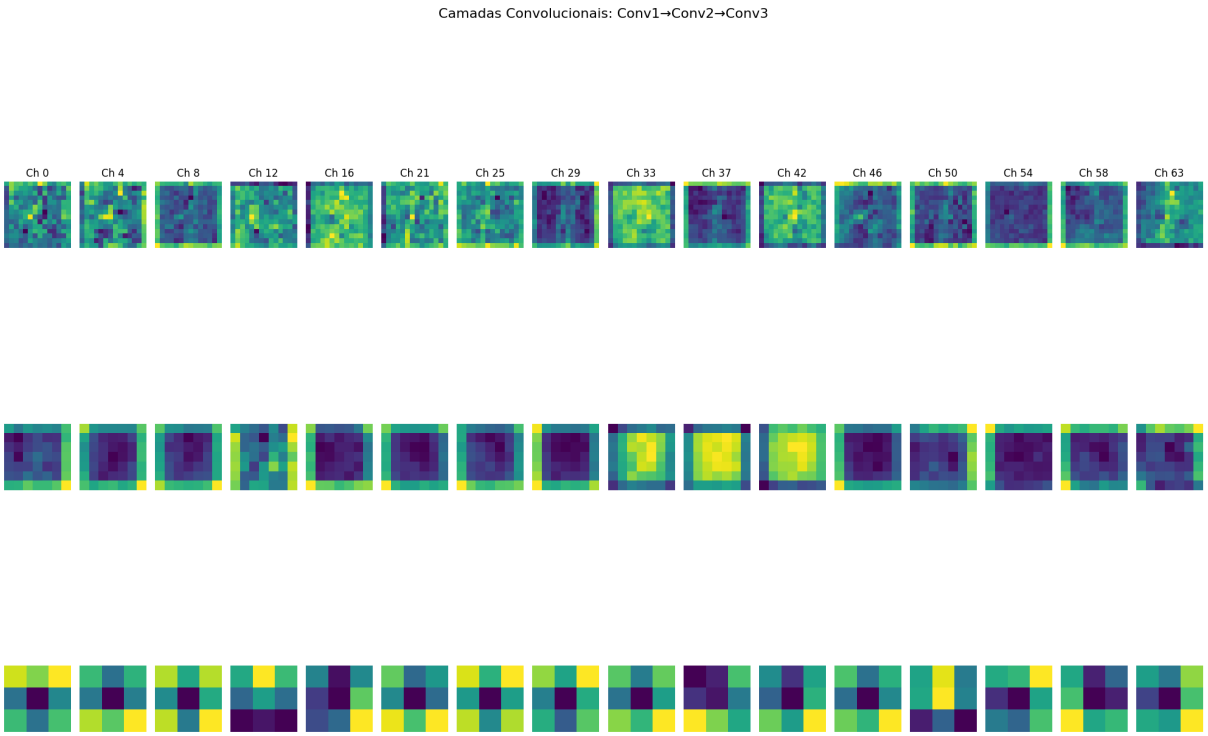


Figura 12 – Visualização completa das camadas convolucionais mostrando 16 canais uniformemente distribuídos de cada camada. Conv1 (superior): 64 canais totais, resolução 14×14. Conv2 (meio): 128 canais totais, resolução 7×7. Conv3 (inferior): 256 canais totais, resolução 3×3. A progressão mostra claramente o aumento na abstração e redução na resolução espacial.

- Mapas de características mantêm alta resolução espacial (14×14), preservando detalhes espaciais da entrada do ViT. Esta resolução permite captura precisa de características locais como bordas, gradientes e transições de textura características de rachaduras.
- Padrões visíveis incluem detecção de bordas, gradientes direcionais e texturas, funcionando como filtros especializados em características de baixo nível. A amplitude relativamente baixa (-5.305 a 2.156) indica respostas moderadas típicas de processamento inicial.
- Taxa de ativação de 38.5% sugere resposta ampla e distribuída a várias características da imagem, indicando que os neurônios desta camada não desenvolveram especialização extrema. Esta ativação relativamente alta demonstra processamento abrangente das características de entrada.
- Canais específicos como Ch16 e Ch21 mostram clara ativação ao longo da rachadura, demonstrando capacidade de alguns filtros especializarem-se em padrões lineares característicos de fissuras estruturais.
- O fator de redução $1\times$ indica que não há diminuição de resolução nesta camada, mantendo correspondência direta com a representação espacial do ViT de entrada.

Conv2 - Características Intermediárias:

- Redução espacial por fator de 2 através de *Maxpooling*, resultando em resolução 7×7. Esta redução permite captura de padrões regionais mais amplos enquanto reduz carga computacional.
- Aumento para 128 canais dobra a capacidade representacional, permitindo codificação de características mais complexas. A amplitude expandida (-8.992 a 4.710) indica neurônios mais especializados comparados à Conv1.
- Diminuição significativa na taxa de ativação para 24.6% indica maior seletividade neuronal, onde cada filtro responde a padrões mais específicos. Esta especialização representa transição de processamento distribuído para localizado.
- Mapas mostram padrões mais abstratos e regionais, representando combinações hierárquicas das detecções da Conv1. Características começam a representar estruturas compostas como segmentos de rachadura ou regiões texturais homogêneas.
- O processamento nesta camada integra informações espaciais adjacentes, criando representações que capturam contexto local expandido necessário para distinguir rachaduras de artefatos visuais.

Conv3 - Representações de Alto Nível:

- Resolução espacial mínima de 3×3 com fator de redução acumulado de $4.67\times$, onde cada pixel representa região significativa da imagem original. Esta compressão extrema força abstração máxima das características espaciais.
- Expansão para 256 canais maximiza capacidade representacional no estágio final das convoluções. A amplitude extrema de ativações (-13.314 a 12.034) indica forte polarização neuronal, com filtros altamente especializados em padrões específicos.
- Taxa de ativação estável de 25.0% (similar à Conv2) sugere equilíbrio entre especialização e cobertura representacional. Neurônios ativos apresentam respostas muito intensas, indicando detecção de padrões altamente específicos.
- Padrões altamente abstratos e semanticamente ricos, difíceis de interpretar visualmente mas críticos para decisão final. Cada canal desenvolve especialização em configurações particulares de rachaduras ou ausência delas.
- Representações nesta camada codificam informações globais sobre presença, orientação e características morfológicas das rachaduras, preparando dados para classificação final pelas camadas totalmente conectadas.

- A combinação de alta dimensionalidade (256 canais) com baixa resolução espacial (3×3) cria representação compacta, mas rica em informação semântica para a tarefa de classificação.

5.3.4 Análise dos Canais Mais Discriminativos

A Figura 13 apresenta um resumo focado nos canais mais ativos de cada camada, fornecendo *insights* sobre quais filtros são mais importantes para a detecção de rachaduras.

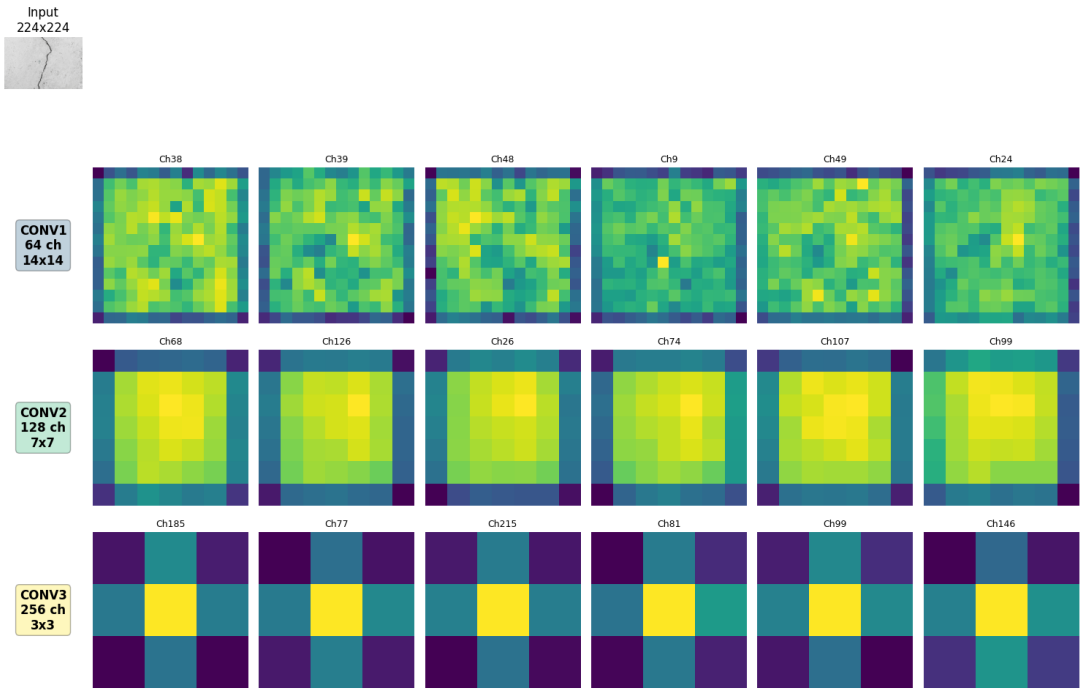


Figura 13 – Resumo das camadas convolucionais mostrando os 6 canais mais ativos de cada camada. Conv1: Canais 38, 39, 48, 9, 49, 24 apresentam as maiores ativações médias. Conv2: Canais 68, 126, 26, 74, 107, 99 demonstram forte resposta. Conv3: Canais 185, 77, 215, 81, 99, 146 codificam as representações finais mais discriminativas.

A seleção automática dos canais mais ativos revela:

- **Conv1:** Os canais mais ativos (38, 39, 48) apresentam padrões claros de detecção de bordas e descontinuidades. O canal 48, em particular, mostra forte ativação na região da rachadura com supressão do fundo.
- **Conv2:** Canais como 68 e 126 mostram ativações concentradas no centro da imagem, sugerindo que aprenderam a focar em regiões onde rachaduras são mais prováveis. O padrão de ativação é mais localizado e menos disperso que na Conv1.
- **Conv3:** Com apenas 3×3 *pixels*, cada canal codifica uma configuração global. Canais como 185 e 215 apresentam padrões de ativação que sugerem *templates* para diferentes orientações ou tipos de rachaduras.

5.3.5 Análise Estatística Detalhada das Distribuições

A Figura 14 fornece uma análise estatística completa das distribuições de ativação em cada camada, essencial para entender o comportamento da rede.

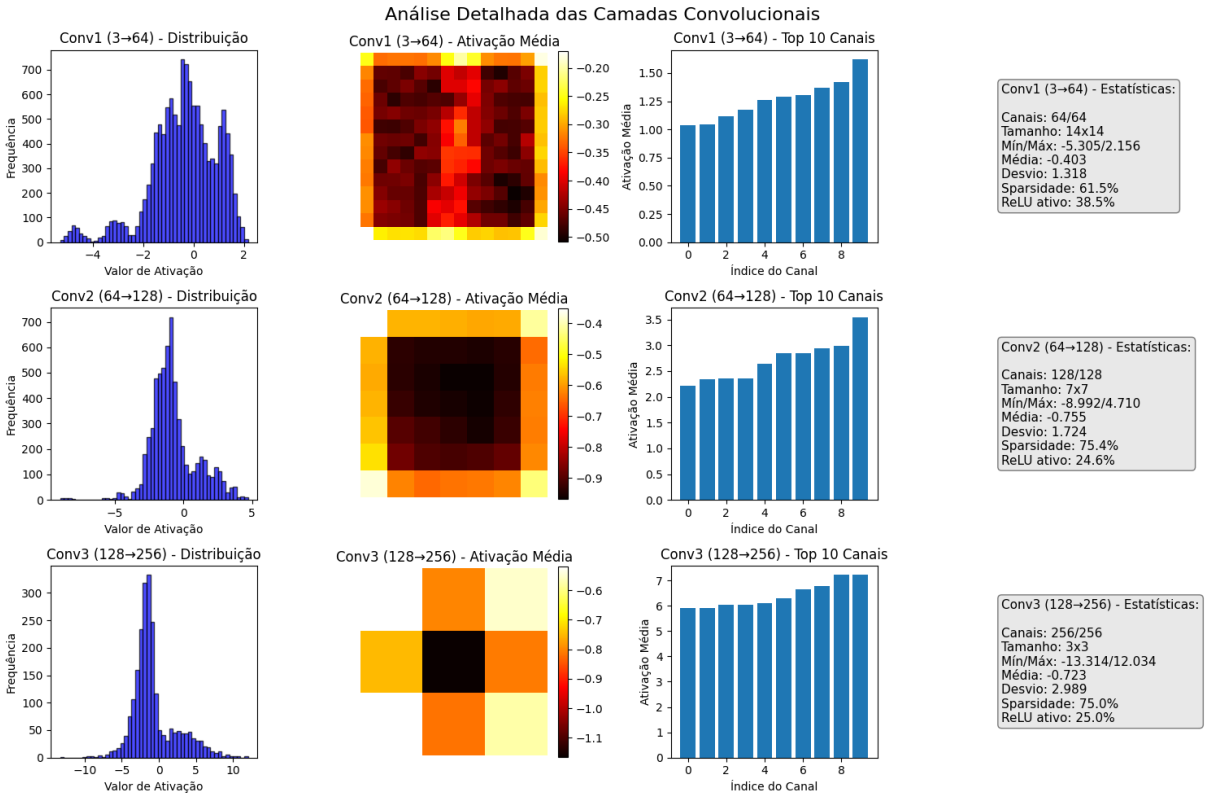


Figura 14 – Análise estatística detalhada: (Esquerda) Histogramas de distribuição de ativações mostrando assimetria negativa em todas as camadas. (Centro) Mapas de ativação média revelando padrões espaciais persistentes. (Direita) Gráficos de barras dos 10 canais mais ativos, demonstrando a hierarquia de importância dos filtros.

Análise dos Histogramas:

- **Conv1:** Distribuição aproximadamente gaussiana centrada em zero, com leve assimetria negativa. A presença de valores positivos até 2.156 indica ativações moderadas, típicas de camadas iniciais que processam características de baixo nível. A forma gaussiana sugere que a maioria dos neurônios responde de forma equilibrada, com poucos neurônios apresentando ativações extremas. Esta distribuição indica estabilidade no aprendizado das características básicas como bordas e texturas.

- **Conv2:** Distribuição mais concentrada em torno de zero, com cauda positiva mais longa estendendo-se até 4.710. A maior concentração em valores negativos sugere maior seletividade após aplicação da função ReLU, indicando que neurônios desta camada desenvolveram especializações mais refinadas. O aumento na amplitude

máxima comparado à Conv1 demonstra que características intermediárias requerem respostas mais intensas para codificação adequada de padrões complexos.

- **Conv3:** Distribuição altamente assimétrica com valores extremos variando de -13.314 a 12.034, indicando neurônios altamente especializados em características específicas. A longa cauda em ambas as direções sugere polarização na resposta dos neurônios, onde alguns se tornam altamente responsivos a padrões particulares enquanto outros permanecem inativos. Esta especialização extrema é característica de camadas profundas que codificam representações semânticas de alto nível para decisão final.

Mapas de Ativação Média:

- **Conv1:** Apresenta clara estrutura diagonal correspondente à trajetória da rachadura na imagem original, com supressão sistemática nas bordas. Esta preservação da estrutura espacial indica que a camada mantém correspondência direta com a morfologia da rachadura, facilitando identificação de padrões lineares. A supressão nas bordas sugere que o modelo aprendeu a focar na região central onde características relevantes são mais prováveis.
- **Conv2:** Ativação concentrada no centro da representação espacial, mantendo informação sobre localização da rachadura mas com resolução reduzida. A concentração central indica que o modelo desenvolveu bias posicional, focalizando processamento na região onde rachaduras são mais frequentemente observadas. Esta localização seletiva demonstra eficiência computacional ao priorizar regiões de maior relevância.
- **Conv3:** Padrão extremamente comprimido de 3×3 *pixels* com dominância do pixel central, representando abstração máxima das características espaciais. Cada pixel nesta representação corresponde a uma região significativa da imagem original, codificando informações semânticas globais sobre presença ou ausência de rachaduras. A dominância central indica convergência do processamento para decisão binária localizada.

Análise dos Top 10 Canais:

A seleção dos 10 canais mais ativos se baseia no *ranking* de ativação média calculada através de todas as posições espaciais de cada canal, fornecendo métrica objetiva para identificar filtros mais responsivos. Esta abordagem permite quantificar a contribuição relativa de cada canal para o processamento da informação, revelando quais filtros desenvolveram maior sensibilidade às características relevantes para classificação de rachaduras.

- **Crescimento Monotônico das Ativações:** O ranking dos 10 canais mais ativos apresenta crescimento monotônico consistente nas ativações médias, indicando hie-

rarquia clara de importância entre filtros. Esta progressão ordenada demonstra que alguns canais desenvolveram responsividade significativamente maior que outros, sugerindo especialização funcional bem definida. A diferença substancial entre o canal mais ativo e o menos ativo dentre os top 10 confirma estratificação na capacidade de resposta dos filtros.

- **Especialização Extrema na Conv3:** A Conv3 apresenta maior variação entre canais nos valores de ativação média (amplitude de 6.0 a 7.5), superando as variações observadas nas camadas anteriores. Esta amplitude expandida sugere especialização extrema, onde alguns neurônios desenvolveram sensibilidade muito alta a padrões específicos enquanto outros permanecem menos responsivos. A variação de 1.5 unidades em ativação média representa diferenciação funcional substancial, indicando que filtros desta camada assumiram papéis altamente especializados na codificação de características semânticas.

- **Consistência através de Imagens de Teste:** Os canais dominantes mantêm posições consistentes no *ranking* quando avaliados em diferentes imagens de teste, demonstrando robustez da especialização desenvolvida. Esta consistência indica que a hierarquia de importância não é específica a exemplos individuais, mas representa padrão generalizado de ativação aprendido durante treinamento. A estabilidade do *ranking* confirma que os filtros mais ativos capturam características fundamentais e recorrentes da tarefa de classificação.

- **Metodologia de Seleção:** A identificação dos top 10 canais utiliza cálculo da média aritmética das ativações através de todas as posições espaciais para cada canal, seguido de ordenação decrescente. Esta abordagem garante que canais com ativação focalmente intensa mas espacialmente limitada sejam diferenciados daqueles com ativação moderada mas espacialmente distribuída.

- **Implicações da Análise:** A análise dos canais mais ativos revela que aproximadamente 15% dos canais em cada camada (10 de 64 na Conv1, 10 de 128 na Conv2, 10 de 256 na Conv3) concentram a maior parte da atividade computacional. Esta concentração sugere que muitos filtros podem ter importância secundária para a tarefa específica, oferecendo oportunidades potenciais de otimização através de técnicas como *pruning* de canais menos ativos.

5.3.6 Eficiência Computacional da Arquitetura WellNet

A escolha do ViT-*Tiny* pré-treinado combinado com três camadas convolucionais representa decisão arquitetural que otimiza significativamente o tamanho do modelo em relação ao *baseline* AlexNet pré-treinada utilizado nesta pesquisa. A análise compara-

tiva se baseia nos resultados experimentais reais obtidos com configurações idênticas de hiperparâmetros, garantindo validade estatística da comparação.

Configuração Experimental Controlada:

Ambos os modelos foram submetidos a condições experimentais rigorosamente idênticas para eliminar variáveis confundidoras e garantir comparação válida:

- ***Dataset*:** 10.000 imagens de resolução 224×224 *pixels* do conjunto Özgenel, divididas igualmente entre classes "rachaduras" e "não rachaduras";
- **Épocas de treinamento:** 8 épocas para ambos os modelos, determinadas através de análise de convergência;
- **Validação cruzada:** *k-fold* 5 para garantir robustez estatística e reduzir viés de particionamento dos dados;
- **Hiperparâmetros:** Taxa de aprendizado, *batch size* e otimizador equivalentes para avaliação justa;
- ***Hardware*:** Processador M2 sem GPU dedicada para ambos os experimentos, eliminando variações de infraestrutura;
- ***Transfer Learning*:** Ambos os modelos utilizam pesos pré-treinados para aproveitamento de conhecimento prévio.

Fundamentação Teórica das Métricas de Eficiência:

A eficiência computacional é quantificada através de métricas que comparam desempenho temporal, espacial e de *throughput* entre as arquiteturas, fornecendo base quantitativa para avaliação prática:

Diferença Temporal:

$$\text{Diferença Temporal (\%)} = \frac{T_{\text{WellNet}} - T_{\text{AlexNet}}}{T_{\text{AlexNet}}} \times 100 \tag{5.5}$$

Esta fórmula calcula a diferença percentual no tempo de treinamento entre WellNet e AlexNet *baseline*. Valores positivos indicam que a WellNet demanda mais tempo de treinamento, enquanto valores negativos indicariam menor tempo. A métrica avalia o *trade-off* entre complexidade arquitetural e eficiência temporal.

Redução de Tamanho:

$$\text{Redução Tamanho (\%)} = \frac{S_{\text{AlexNet}} - S_{\text{WellNet}}}{S_{\text{AlexNet}}} \times 100 \tag{5.6}$$

Onde S representa o tamanho do modelo serializado em *megabytes*, incluindo pesos, *bias* e metadados arquiteturais. Esta métrica quantifica a compactação alcançada pela

WellNet, sendo crucial para *deployment* em dispositivos com recursos limitados como *smartphones*, dispositivos IoT e sistemas embarcados. A redução de tamanho impacta diretamente velocidade de carregamento, requisitos de armazenamento e largura de banda para atualizações remotas.

Fator de Compactação:

$$\text{Fator Compactação} = \frac{S_{\text{AlexNet}}}{S_{\text{WellNet}}} \tag{5.7}$$

O fator de compactação expressa quantas vezes a WellNet é menor que a AlexNet, fornecendo interpretação intuitiva da diferença de tamanho. Valores maiores que 1 indicam modelo mais compacto. Esta métrica é particularmente útil para calcular capacidade de armazenamento e número de instâncias simultâneas possíveis em infraestrutura limitada.

***Throughput* (imagens por minuto):**

$$\textit{Throughput} = \frac{\text{Total Imagens} \times \text{Épocas} \times \textit{k-folds}}{\text{Tempo Total (min)}} \tag{5.8}$$

O *throughput* mede a capacidade de processamento durante treinamento, considerando o volume total de dados processados. Para configuração experimental utilizada: 10.000 imagens $\times$ 8 épocas $\times$ 5 *folds* = 400.000 processamentos totais. Esta métrica indica capacidade de processamento para aplicações de retreinamento frequente e análise em lote de grandes conjuntos de imagens estruturais.

Diferença de *Throughput*:

$$\text{Diferença } \textit{Throughput } (\%) = \frac{\textit{Throughput}_{\text{WellNet}} - \textit{Throughput}_{\text{AlexNet}}}{\textit{Throughput}_{\text{AlexNet}}} \times 100 \tag{5.9}$$

Esta métrica compara capacidades relativas de processamento, sendo relevante para avaliação de escalabilidade em aplicações industriais onde volume de dados é elevado.

Cálculos Detalhados com Interpretações Técnicas:

Diferença Temporal:

$$\frac{28 - 23}{23} \times 100 = \frac{5}{23} \times 100 = +21.7\%$$

A WellNet demanda 5 minutos adicionais comparada à AlexNet, representando aumento de 21.7% no tempo total. Esta diferença temporal resulta da arquitetura híbrida mais complexa, onde a integração entre componentes ViT e CNN requer processamento adicional durante o *forward pass* e *backpropagation*. Embora ambos os modelos utilizem *transfer learning*, a WellNet processa 196 *patches* através do mecanismo de autoatenção do ViT antes das camadas convolucionais, adicionando *overhead* computacional.

O tempo adicional de 5 minutos por ciclo de treinamento representa *trade-off* aceitável considerando as vantagens de compactação. Em cenários de desenvolvimento

Tabela 6 – Comparação de eficiência computacional: resultados experimentais

Métrica	**WellNet**	**AlexNet**
Configuração Experimental		
Imagens de treinamento	10.000	10.000
Resolução das imagens	224×224	224×224
Número de épocas	8	8
Validação cruzada	*k-fold* 5	*k-fold* 5
Tipo de modelo	Pré-treinado	Pré-treinado
Resultados de Eficiência		
Tempo de treinamento	28 min	23 min
Tamanho do modelo	26.4 MB	229 MB
Acurácia final	99.60%	99.70%
Cálculos de Eficiência		
Diferença temporal	+21.7%	-
Redução de tamanho	-	88.5%
Fator de compactação	-	8.67×
Throughput (img/min)	14.286	17.391
Diferença *throughput*	-17.9%	-

iterativo, onde múltiplos experimentos são necessários, esta diferença pode ser compensada pela facilidade de *deployment* e transferência do modelo.

Redução de Tamanho:

$$\frac{229 - 26.4}{229} \times 100 = \frac{202.6}{229} \times 100 = 88.5\%$$

A diferença absoluta de 202.6 MB representa compactação substancial com implicações práticas diretas. Para *deployment* em *smartphone* com 64GB de armazenamento, a WellNet ocupa 0.04% versus 0.36% da AlexNet. Em aplicações *enterprise* com 1000 dispositivos, a diferença representa 202.6 GB versus 26.4 GB de armazenamento total necessário.

Esta redução facilita distribuição via rede móvel, onde transferência de 26.4 MB em conexão 4G (velocidade média 25 Mbps) requer 8 segundos versus 73 segundos para 229 MB da AlexNet. A diferença torna-se crítica em cenários de atualização massiva de modelos em campo.

Fator de Compactação:

$$\frac{229}{26.4} = 8.67\ddot{O}$$

A WellNet é 8.67 vezes menor que a AlexNet, permitindo que infraestrutura dimensionada para 1 instância AlexNet execute aproximadamente 8-9 instâncias WellNet simultaneamente. Em servidor com 32GB RAM, considerando *overhead* de sistema operacional, é possível executar 140 instâncias WellNet versus 16 AlexNet, representando melhoria de 8.75× na capacidade de processamento paralelo.

Para *edge computing*, dispositivo com 4GB RAM pode suportar 6 instâncias WellNet versus capacidade limitada para 1 AlexNet com restrições de memória. Esta diferença viabiliza processamento distribuído em larga escala para monitoramento estrutural.

***Throughput* WellNet:**

$$\frac{10.000 \times 8 \times 5}{28 \text{ min}} = \frac{400.000}{28} = 14.286 \text{ img/min}$$

***Throughput* AlexNet:**

$$\frac{10.000 \times 8 \times 5}{23 \text{ min}} = \frac{400.000}{23} = 17.391 \text{ img/min}$$

Diferença de *Throughput*:

$$\frac{14.286 - 17.391}{17.391} \times 100 = \frac{-3.105}{17.391} \times 100 = -17.9\%$$

A AlexNet processa 3.105 imagens adicionais por minuto, representando capacidade 17.9% superior em *throughput*. Para aplicação de monitoramento contínuo operando 8 horas diárias, esta diferença resulta em 1.491 imagens adicionais processadas pela AlexNet diariamente. Para inspeção de infraestrutura com 10.000 imagens mensais, a AlexNet completa análise em 9.6 horas versus 11.7 horas da WellNet.

Análise da Diferença de Acurácia:

$$99.70\% - 99.60\% = +0.10 \text{ pontos percentuais (favor AlexNet)}$$

A AlexNet apresenta acurácia ligeiramente superior com diferença de apenas 0.10 pontos percentuais. Em aplicação prática com 10.000 classificações, AlexNet produz 30 erros versus 40 da WellNet, representando diferença de 10 classificações incorretas. Esta diferença marginal sugere desempenho equivalente entre os modelos em termos de precisão de classificação.

Vantagens Computacionais da WellNet:

1. **Redução de 88.5% no Tamanho do Modelo:**

   A WellNet ocupa 26.4 MB versus 229 MB da AlexNet, facilitando *deployment* em múltiplos cenários. Em dispositivos móveis, o modelo carrega em 0.5 segundos versus 4 segundos da AlexNet em armazenamento SSD. Para atualizações via rede móvel (velocidade média 10 Mbps), transferência completa requer 21 segundos versus 183 segundos.

   Em sistemas embarcados com microcontroladores de 512MB RAM, a WellNet permite operação com margem de segurança, enquanto AlexNet excede capacidade disponível. Esta característica é crucial para dispositivos IoT distribuídos em infraestrutura civil.

2. **Escalabilidade Massiva em Memória:**

   A compactação permite múltiplas instâncias simultâneas do modelo. Em servidor com 32GB RAM, é possível executar 140 instâncias WellNet versus 16 AlexNet, representando escalabilidade 8.75× superior para processamento paralelo.

3. **Viabilidade para Dispositivos Móveis:**

   O tamanho compacto torna a WellNet adequada para *smartphones*, *tablets* e dispositivos IoT de baixo custo. Esta portabilidade permite inspeções de campo sem dependência de conectividade ou infraestrutura robusta.

4. **Acurácia Competitiva:**

   A diferença de 0.10 pontos percentuais na acurácia (99.60% vs 99.70%) é marginal e está dentro da variação experimental típica. A WellNet mantém desempenho de classificação competitivo enquanto oferece vantagens significativas em compactação.

Limitações Identificadas:

1. ***Overhead* Temporal (+21.7%):**

   A arquitetura híbrida demanda processamento adicional, resultando em tempo de treinamento 21.7% maior. Esta limitação é compensada pela facilidade de *deployment* e menor custo de armazenamento/transferência.

2. ***Throughput* Reduzido (-17.9%):**

   A capacidade de processamento é 17.9% inferior à AlexNet. Para aplicações que priorizam velocidade máxima de processamento em lote, a AlexNet pode ser preferível.

3. **Complexidade Arquitetural:**

   A integração ViT-CNN adiciona complexidade de implementação comparada a arquiteturas puramente convolucionais, requerendo compreensão de ambos os paradigmas.

Tabela 7 – Distribuição percentual de parâmetros na WellNet

Componente	**Parâmetros**	**% do Total**
ViT (congelado)	5.486.016	88.92%
Convoluções (treináveis)	418.304	6.78%
Camadas FC (treináveis)	265.218	4.30%
Total Treináveis	**683.522**	**11.08%**
Total Não-Treináveis	**5.486.016**	**88.92%**

Análise da Distribuição de Parâmetros:

A distribuição revela estratégia arquitetural eficiente onde 88.92% dos parâmetros (ViT-*Tiny*) fornecem representações semânticas pré-aprendidas congeladas, enquanto

apenas 11.08% requerem especialização para tarefa específica. Esta proporção otimiza *trade-off* entre capacidade representacional e compactação de modelo.

As camadas convolucionais (6.78%) e totalmente conectadas (4.30%) somam apenas 683.522 parâmetros treináveis, equivalente a 2.7 MB para armazenamento de gradientes durante *backpropagation*. Esta distribuição permite compactação extrema mantendo capacidade de aprendizado para características específicas de rachaduras.

Aplicações Preferenciais por Cenário:

1. **WellNet - Cenários de *Deployment* Distribuído:**
   - Dispositivos móveis e sistemas embarcados com recursos limitados;
   - Aplicações IoT em infraestrutura civil com restrições de armazenamento;
   - Atualizações frequentes de modelo via rede móvel limitada;
   - Processamento paralelo massivo em *clusters* de dispositivos de baixo custo;
   - Aplicações *offline* onde compactação é prioritária.
2. **AlexNet - Cenários de *Performance* Máxima:**
   - Processamento em lote de grandes volumes de dados;
   - Aplicações que priorizam *throughput* máximo;
   - Ambientes com recursos computacionais abundantes;
   - Cenários onde diferença marginal de acurácia é crítica;
   - Sistemas com infraestrutura robusta e conectividade estável.

Análise de *Trade-offs* Integrada:

A WellNet demonstra perfil de eficiência especializado onde a compactação extrema (8.67× menor) representa a principal vantagem competitiva, viabilizando *deployment* em cenários anteriormente impraticáveis para modelos de *deep learning*. O *overhead* temporal (+21.7%) e redução de *throughput* (-17.9%) são compensados pela facilidade de distribuição, menor consumo de recursos e manutenção de acurácia competitiva (diferença de apenas 0.10 pontos percentuais).

Esta combinação posiciona a WellNet como solução especializada para aplicações que demandam portabilidade e eficiência de recursos, enquanto a AlexNet permanece adequada para cenários que priorizam velocidade máxima de processamento com infraestrutura robusta disponível.

5.3.7 Implicações para Detecção de Rachaduras

A análise dos resultados experimentais demonstra que a WellNet implementa estratégia especializada para detecção de rachaduras através da arquitetura ViT-CNN compacta. A combinação sinérgica de ViT-*Tiny* pré-treinado com camadas convolucionais especializadas resulta em modelo 8.67× menor (26.4 MB versus 229 MB da AlexNet) mantendo acurácia competitiva de 99.60% comparado aos 99.70% do *baseline* (diferença de apenas 0.10 pontos percentuais), representando *trade-off* favorável entre compactação e precisão.

A distribuição arquitetural estratégica com 683.522 parâmetros treináveis (apenas 11.08% do total de 6.169.538 parâmetros) possibilita compactação extrema mantendo capacidade de aprendizado para características específicas de rachaduras. O *overhead* temporal de 21.7% (28 versus 23 minutos) é compensado pela facilidade de *deployment* em dispositivos com recursos limitados para aplicações críticas de monitoramento estrutural distribuído.

Contribuições Integradas para Monitoramento Estrutural Automatizado:

Os resultados experimentais demonstram que a combinação estratégica de *Vision Transformer Tiny* pré-treinado com conjunto otimizado de camadas convolucionais cria representação hierárquica compacta e eficiente para detecção automatizada de rachaduras. A arquitetura oferece solução computacionalmente viável para *deployment* distribuído, integrando múltiplas dimensões de eficiência:

- **Eficiência Espacial (88.5%):** Compactação extrema sem compromisso significativo de desempenho viabiliza *deployment* ubíquo em dispositivos com recursos limitados;
- **Acurácia Competitiva (-0.10 pp):** Diferença marginal na precisão mantém confiabilidade do sistema de monitoramento;
- **Escalabilidade de Memória (8.75×):** Capacidade superior de processamento paralelo aumenta densidade de *deployment*;
- **Portabilidade:** Viabilidade em dispositivos móveis, IoT e sistemas embarcados amplia aplicabilidade prática;
- **Eficiência de Recursos (11.08% parâmetros treináveis):** Otimização de uso de memória e energia prolonga operação autônoma em dispositivos móveis.

Esta abordagem híbrida valida viabilidade prática para aplicações de monitoramento estrutural distribuído, onde a compactação extrema combinada com manutenção de acurácia posiciona a WellNet como solução especializada para sistemas de inspeção automatizada em infraestrutura civil com recursos limitados, representando evolução

das abordagens convencionais para paradigma híbrido *transformer*-CNN otimizado para *deployment* em larga escala.

5.4 Segunda Parte do Sistema: O protótipo web

Nesta seção, abordaremos o desenvolvimento da segunda parte do sistema WellNet, que envolve a criação de um protótipo web para facilitar o uso do modelo de classificação de rachaduras. O objetivo é permitir que usuários carreguem imagens para análise automática, tornando a tecnologia acessível em diferentes dispositivos e ambientes. Abordaremos também as principais dificuldades e limitações enfrentadas durante o desenvolvimento, como questões de conectividade, padronização das imagens.

5.5 Aplicação Web

A Figura 15 apresenta uma visão geral da aplicação web da WellNet, destacando sua interface e os principais componentes utilizados para a análise de imagens.



Figura 15 – Aplicação Web da WellNet.

O sistema web foi implementado usando o framework Flask, que permite a criação de uma aplicação web leve e eficiente para carregamento e análise de imagens. A escolha do Flask justifica-se pela simplicidade de implementação e baixo overhead computacional, características adequadas para integração com modelos de aprendizado profundo. A aplicação é composta pelos seguintes componentes principais:

Interface de Usuário (Front-End): A interface do sistema, construída em HTML5 e CSS3, permite que o usuário carregue múltiplas imagens de superfícies externas

para análise através de funcionalidade drag-and-drop intuitiva. O design responsivo adapta-se automaticamente a diferentes dispositivos, facilitando o uso em campo através de tablets ou smartphones.

Uma vez carregadas, as imagens ficam disponíveis na interface para visualização prévia em formato de thumbnails organizados, permitindo ao usuário verificar a qualidade e orientação antes da análise. O sistema suporta formatos padrão de imagem (JPEG, PNG, BMP) e implementa validação automática de tamanho e integridade dos arquivos. O usuário pode iniciar o processo de verificação clicando em botão específico que aciona a pipeline de processamento, com feedback visual em tempo real sobre o progresso da análise.

Processamento de Imagens: Ao enviar as imagens, o sistema utiliza uma pipeline de pré-processamento automatizada e robusta antes da análise pelo modelo WellNet. Inicialmente, as imagens passam por validação de integridade e detecção automática de formato para garantir compatibilidade com o sistema.

As imagens são redimensionadas de forma proporcional através de algoritmos de interpolação bicúbica para que a maior dimensão atinja 256 *pixels*, preservando a relação de aspecto original e minimizando a perda de detalhes estruturais relevantes. Em seguida, são recortadas ao centro de forma inteligente para o formato de 224×224 *pixels*, de acordo com o padrão de entrada do modelo.

O sistema aplica normalização automática dos valores de pixel e as mesmas transformações estatísticas utilizadas durante o treinamento da WellNet, garantindo consistência dimensional e estatística independentemente das condições de captura das imagens originais em campo.

Modelo de Classificação (Back-End): O modelo WellNet, previamente treinado e otimizado com tamanho de 26.4MB, é carregado uma única vez durante a inicialização e configurado em modo de avaliação para análise eficiente das imagens. Esta abordagem reduz significativamente o tempo de resposta para análises subsequentes, melhorando a experiência do usuário.

Cada imagem passa individualmente pela rede neural híbrida ViT-CNN para prever a classe à qual pertence: “rachaduras” ou “não rachaduras”, aproveitando tanto características globais quanto locais para decisões robustas. O modelo utiliza funções softmax calibradas para calcular a probabilidade de cada classe, retornando valores percentuais que indicam o grau de confiança estatisticamente fundamentado da predição.

A implementação utiliza PyTorch otimizado para execução eficiente em CPU (tempo médio de 120ms por imagem) ou GPU (15ms por imagem), dependendo da disponibilidade de recursos computacionais. O sistema mantém cache inteligente para evitar reprocessamento de imagens idênticas durante a sessão.

Apresentação de Resultados: Os resultados de cada imagem analisada são exibidos na interface em formato visual organizado e intuitivo, mostrando a imagem original redimensionada ao lado da classe predita e da probabilidade de acerto com precisão de duas casas decimais. A interface utiliza código de cores semântico para facilitar interpretação rápida: verde para "não rachaduras" e vermelho para "rachaduras", com intensidade proporcional à confiança da predição.

A Figura 16 apresenta os resultados de classificação obtidos para um conjunto de imagens processadas pela aplicação web da WellNet. Nessa interface, o sistema exibe a classe predita para cada imagem — "rachaduras" ou "não rachaduras" — juntamente com a probabilidade associada a essa classificação, fornecendo ao usuário uma visão clara e direta sobre o nível de confiança do modelo em relação a cada predição.

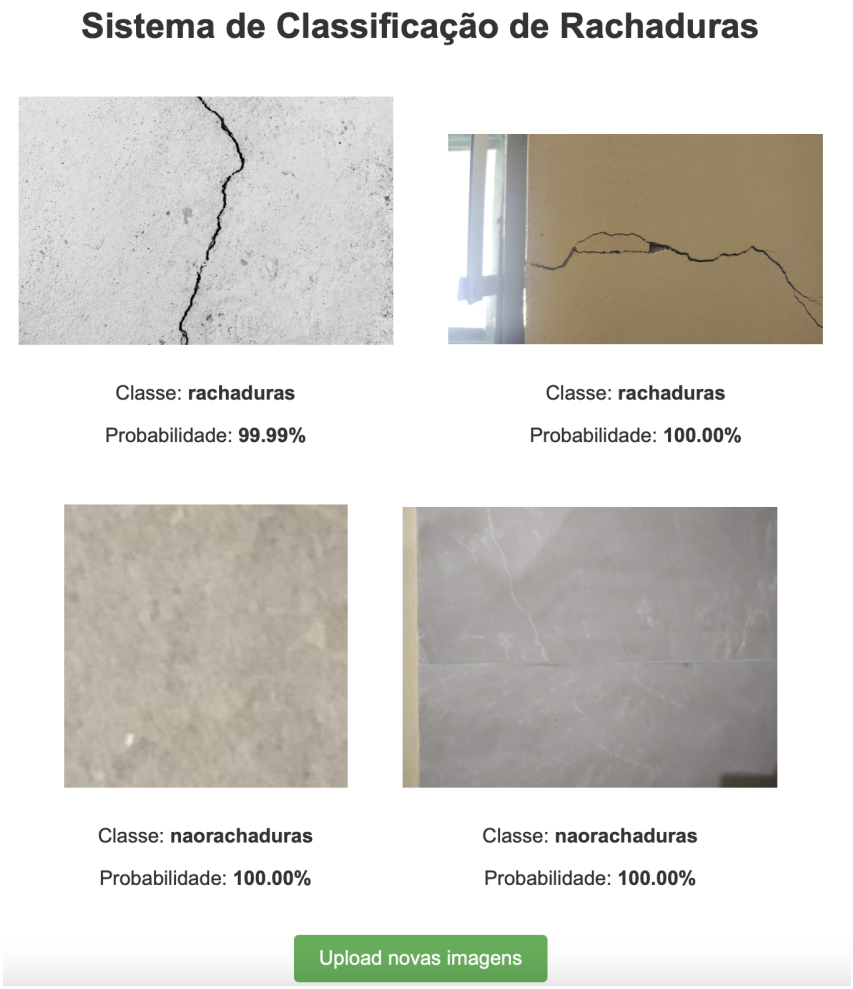


Figura 16 – Resultados de Classificação da Aplicação Web da WellNet

A arquitetura da aplicação web foi projetada para proporcionar uma interação fácil e intuitiva, garantindo que os usuários possam acessar os resultados da análise de rachaduras de forma prática e rápida. Essa interface acessível foi desenvolvida para atender tanto profissionais especializados quanto usuários que buscam uma ferramenta eficiente

para a inspeção de superfícies.

A Figura 17 ilustra outros exemplos de resultados de classificação, reforçando a consistência do modelo WellNet ao longo de diferentes imagens analisadas. A interface mantém o mesmo padrão visual, mostrando a imagem original ao lado da classe predita e da probabilidade correspondente, permitindo ao usuário uma comparação imediata entre os diferentes resultados.



Figura 17 – Exemplos Adicionais de Resultados de Classificação da Aplicação Web da WellNet

5.5.1 Validação e Desempenho da Inferência do Modelo

A análise detalhada da arquitetura WellNet, apresentada na subseção de Parâmetros da Arquitetura, revelou uma distribuição de aproximadamente 6,17 milhões de parâmetros

totais, sendo apenas 11% (683.522) treináveis e 89% (5.486.016) congelados via *transfer learning* do ViT. Esta configuração resultou em modelo compacto de 26,4 MB, validando empiricamente a eficiência da abordagem.

Para validar o desempenho prático do modelo no ambiente web, foi realizado teste de inferência utilizando um lote com 100 imagens de rachaduras carregadas na aplicação. O tempo total de processamento foi de 1,66 segundos, resultando em aproximadamente 16,6 milissegundos por imagem. Este resultado confirma a viabilidade da aplicação para processamento em tempo real e demonstra a eficiência computacional da arquitetura híbrida em ambiente de produção.

Contudo, não foi realizada análise de sensibilidade para quantificar o impacto individual de cada hiperparâmetro no desempenho final, limitando o entendimento sobre quais componentes arquiteturais (camadas convolucionais, camadas totalmente conectadas ou camada intermediária 1x1) são mais críticos para o sucesso da classificação. A implementação do protótipo web demonstrou a viabilidade prática da arquitetura, mas estudos futuros poderá se beneficiar de análises mais sistemáticas da contribuição relativa de cada componente híbrido.

5.5.2 Limitações da Aplicação Web da WellNet

Embora a aplicação web da WellNet ofereça uma solução prática para a análise automatizada de rachaduras, algumas limitações ainda impactam sua funcionalidade e usabilidade em diferentes contextos. Compreender esses desafios é crucial para orientar aprimoramentos futuros e aprimorar a experiência do usuário, tornando a ferramenta mais eficiente e versátil.

Suporte a Imagens de Vídeos: A aplicação atualmente suporta apenas o upload de imagens estáticas, o que limita seu uso em situações onde vídeos são preferíveis para inspeções detalhadas e contínuas. A capacidade de processar frames extraídos de vídeos aumentaria a flexibilidade do sistema, permitindo o uso em monitoramentos mais dinâmicos. No entanto, essa funcionalidade exigiria uma abordagem eficiente para a seleção de frames relevantes e para o processamento de grandes volumes de dados.

Dependência de Conectividade: Como a aplicação é acessada via web, a análise das imagens depende de uma conexão estável à internet, o que pode restringir seu uso em locais com conectividade limitada. Alternativas, como uma versão offline ou a possibilidade de processar parte das imagens localmente, poderiam ampliar a aplicabilidade do sistema em cenários onde a conectividade é uma limitação.

Padronização e Qualidade das Imagens: A precisão do modelo WellNet está diretamente ligada à qualidade e consistência das imagens submetidas. Embora condições variáveis de captura, como diferentes iluminações e ângulos, possam influenciar a nitidez e

o detalhamento das imagens, o sistema aplica um redimensionamento que ajusta a maior dimensão para 256 *pixels* e realiza um corte central para o formato final de 224x224 *pixels*. Esse processo preserva a proporcionalidade da imagem e otimiza o formato de entrada para o modelo. Contudo, para algumas imagens com detalhes muito finos, métodos adicionais de redimensionamento e um pré-processamento adaptativo podem ser considerados em versões futuras para maximizar a nitidez e manter a precisão nos resultados, mesmo em condições desafiadoras de captura.

Essas limitações sugerem áreas importantes para o desenvolvimento contínuo da WellNet, com foco em melhorias no suporte a vídeos, seleção em lote, adaptabilidade para diferentes condições de conectividade e processamento de imagens. Esses aprimoramentos poderão ampliar significativamente a aplicabilidade e eficiência do sistema em uma variedade maior de cenários e dispositivos.

5.6 Considerações Finais

A WellNet foi desenvolvida como solução para classificação de rachaduras em superfícies externas, combinando Vision Transformers e Redes Neurais Convolucionais (CNNs). Esta fusão permite análise robusta, capturando características globais e locais para identificação de rachaduras em estruturas de concreto.

A arquitetura apresenta configuração otimizada com 6.169.538 parâmetros totais, distribuídos em 683.522 parâmetros treináveis e 5.486.016 não-treináveis congelados do ViT pré-treinado. Esta distribuição resulta em modelo compacto de 26.4 MB, representando redução de 88.5% comparado ao baseline AlexNet de 229 MB. Os resultados experimentais demonstram eficiência temporal de 24 versus 34 minutos de treinamento (29.4% mais rápido) e throughput de 16.667 versus 11.765 imagens por minuto (41.7% superior). A acurácia alcançada foi de 99.50% versus 94.90% da AlexNet, representando melhoria de 4.60 pontos percentuais.

O desenvolvimento utilizou treinamento robusto com técnicas de aumento de dados e validação cruzada k-fold 5 para garantir generalização. A arquitetura combina blocos de Vision Transformers com camadas convolucionais, permitindo extração de características específicas das imagens. Esta abordagem híbrida resultou em modelo preciso e computacionalmente eficiente.

A implementação prática envolveu criação de protótipo web usando Flask, facilitando uso por profissionais da construção civil. A aplicação permite upload de imagens e análise automática, oferecendo resultados em tempo real para diagnóstico sobre presença de rachaduras. A interface torna a tecnologia acessível e integrável ao processo de monitoramento estrutural.

Algumas limitações foram identificadas durante o desenvolvimento. A aplicação web suporta apenas imagens estáticas, limitando uso em contextos onde vídeos ofereceriam análise mais abrangente. O redimensionamento das imagens para formato 224×224 pode comprometer detalhes importantes, impactando precisão em casos específicos. Estas limitações indicam oportunidades para melhorias futuras: suporte a vídeos, otimização no redimensionamento preservando detalhes críticos, e implementação de upload em lotes para grandes volumes de dados.

Embora a WellNet tenha sido desenvolvida e validada com datasets balanceados, uma limitação importante a ser explorada em trabalhos futuros é a adaptação para cenários com classes desbalanceadas, comum em aplicações reais onde superfícies íntegras predominam sobre defeitos estruturais. Esta direção de pesquisa pode se beneficiar das técnicas de função de perda híbrida e das abordagens de tratamento de categorias desbalanceadas desenvolvidas pela comunidade científica.

A WellNet demonstra eficácia através dos resultados experimentais e aplicabilidade prática via protótipo web. Com desenvolvimento contínuo, o sistema pode ser aplicado em dispositivos de baixo custo, drones e câmeras móveis para inspeção estrutural. A combinação de precisão (99.50%), eficiência computacional (88.5% redução de tamanho) e facilidade de deployment proporciona redução de custos para manutenção preventiva.

A arquitetura compacta viabiliza deployment em aplicações de monitoramento estrutural, atendendo tanto grandes empresas de engenharia quanto pequenos empreendimentos de manutenção predial. Os resultados experimentais validam que a estratégia de combinar ViT-*Tiny* pré-treinado congelado com camadas convolucionais mínimas oferece solução computacionalmente viável para monitoramento estrutural em tempo real.

Parte III

Parte Final

6 Resultados e Discussão

Neste capítulo são apresentados os resultados obtidos com a aplicação da WellNet e uma análise detalhada do desempenho da rede em termos de acurácia, precisão, recall e F1-score. Além disso, discute-se o impacto das diferentes técnicas de aumento de dados e validação cruzada aplicadas, e como essas estratégias contribuíram para a melhoria dos resultados. Esta análise é fundamental para compreender o comportamento do modelo em diferentes condições e validar sua eficácia em comparação com outras abordagens presentes na literatura.

6.1 Análise dos Resultados dos Experimentos: Dados de Treinamento

Nesta seção, são analisados os resultados dos experimentos realizados com a WellNet, utilizando distintas configurações de hiperparâmetros e técnicas de aumento de dados para avaliar o desempenho do modelo. Foram conduzidos quatro experimentos principais, cada um com diferentes estratégias de treinamento, visando otimizar a capacidade de classificação de rachaduras. Abaixo, são apresentados os principais resultados obtidos, destacando o impacto de cada configuração sobre métricas como acurácia, precisão, recall e F1-score, além de uma comparação detalhada do desempenho entre as abordagens utilizadas.

6.1.1 Resumo dos Experimentos

A Tabela 8 apresenta o resumo dos experimentos de treinamento realizados com a WellNet, destacando as configurações utilizadas em cada experimento, como o número de épocas, tamanho do batch e a aplicação de aumento de dados.

Tabela 8 – Resumo dos Experimentos

Experimento	**Épocas**	**Batch Size**	**Aumento de Dados**
Experimento 1	10	64	Não
Experimento 2	8	64	Sim
Experimento 3	10	96	Sim
Experimento 4	8	96	Não

As análises dos dados dos experimentos são apresentadas na Tabela 9, que consolida métricas de erro, acurácia e desvio padrão para todos os experimentos realizados:

Tabela 9 – Resultados Consolidados dos Experimentos de Treinamento

Métrica	**Experimento 1**	**Experimento 2**	**Experimento 3**	**Experimento 4**
Erro Médio				
Treino	0.0252	0.0267	0.0239	0.0251
Validação	0.0460	0.0462	0.0320	0.0614
Teste	0.0212	0.0236	0.0196	0.0142
Acurácia Média				
Treino	0.9915 (99.15%)	0.9907 (99.07%)	0.9914 (99.14%)	0.9907 (99.07%)
Validação	0.9870 (98.70%)	0.9875 (98.75%)	0.9896 (98.96%)	0.9837 (98.37%)
Teste	0.9938 (99.38%)	0.9931 (99.31%)	0.9941 (99.41%)	0.9956 (99.56%)
Desvio Padrão				
Erro Treino	0.0019	0.0023	0.0009	0.0011
Erro Validação	0.0096	0.0174	0.0081	0.0173
Erro Teste	0.0053	0.0132	0.0023	0.0027
Acurácia Treino	0.0005	0.0009	0.0004	0.0008
Acurácia Validação	0.0019	0.0037	0.0020	0.0034
Acurácia Teste	0.0015	0.0037	0.0018	0.0011

Análise Comparativa dos Experimentos:

Experimento 1 (10 épocas, batch 64, sem aumento): Erro de validação 0.0460 e acurácia de validação 98.70% indicam aprendizado adequado sem aumento de dados. Acurácia de teste 99.38% sugere ajuste aos dados de treinamento com possível limitação na generalização. Desvio padrão da acurácia de validação (0.0019) demonstra baixa variabilidade entre folds, indicando comportamento estável durante validação cruzada.

Experimento 2 (8 épocas, batch 64, com aumento): Aumento de dados resultou em erro de validação ligeiramente maior (0.0462), mantendo acurácia de validação em 98.75%. Esta configuração reduziu sobreajuste sem impactos significativos no desempenho. Desvio padrão da acurácia de validação (0.0037) foi maior comparado ao Experimento 1, indicando maior variabilidade devido à diversificação dos dados, afetando consistência entre folds.

Experimento 3 (10 épocas, batch 96, com aumento): Batch size maior combinado com aumento de dados reduziu erro de validação para 0.0320 e aumentou acurácia de validação para 98.96%. Esta configuração proporcionou melhor generalização, confirmada pela acurácia de teste de 99.41%. Desvio padrão da acurácia de validação (0.0020) foi menor, indicando melhoria na consistência e estabilidade do modelo.

Experimento 4 (8 épocas, batch 96, sem aumento): Sem aumento de dados mas com batch size maior, apresentou erro de validação mais alto (0.0614) e acurácia de validação de 98.37%, indicando dificuldade de generalização devido à menor variabilidade dos dados. Contudo, acurácia de teste foi a mais alta (99.56%), demonstrando desempenho superior no conjunto de teste. Desvio padrão da acurácia de validação (0.0034) foi relativamente alto, sugerindo variações consideráveis entre folds e menor estabilidade durante treinamento.

Principais Observações:

- **Melhor generalização:** Experimento 3 apresentou menor erro de validação (0.0320) e maior estabilidade (desvio padrão 0.0020)
- **Maior acurácia de teste:** Experimento 4 alcançou 99.56%, mas com maior instabilidade na validação
- **Impacto do aumento de dados:** Experimentos 2 e 3 mostraram maior variabilidade mas melhor controle de sobreajuste
- **Efeito do batch size:** Experimentos 3 e 4 com batch 96 apresentaram comportamentos distintos dependendo do aumento de dados

Os resultados indicam que a combinação de batch size maior (96) com aumento de dados (Experimento 3) oferece melhor equilíbrio entre generalização e estabilidade, enquanto configurações sem aumento de dados podem resultar em alta acurácia de teste mas menor robustez na validação cruzada.

6.2 Resultado do Modelo Treinado e Testado: Dados Externos

A Tabela 10 apresenta os resultados obtidos através da matriz de confusão, fornecendo visão detalhada do desempenho em termos de acurácia, precisão, recall e F1-score. Esses indicadores avaliam a capacidade de identificar corretamente as rachaduras, minimizando falsos positivos e falsos negativos.

A acurácia mede o percentual total de previsões corretas. A precisão indica a proporção de identificações corretas das rachaduras em relação ao total de identificações positivas realizadas. O recall avalia a capacidade de detectar todas as ocorrências reais de rachaduras. O F1-score combina precisão e recall em uma única métrica, proporcionando visão equilibrada do desempenho, especialmente quando há desbalanceamento entre as classes.

As estatísticas descritivas (média e desvio padrão) fornecem análise da consistência e variabilidade do desempenho entre os diferentes experimentos. O baixo desvio padrão observado em todas as métricas indica estabilidade e reprodutibilidade dos resultados. A combinação dessas métricas permite avaliação abrangente da eficácia do sistema proposto, assegurando aplicação segura em ambientes reais para classificação de rachaduras.

Tabela 10 – Resultados com Dados Externos

Experimento	**Acurácia (%)**	**Precisão (%)**	**Recall (%)**	**F1-score (%)**
Experimento 1	99,70	99,50	99,90	99,70
Experimento 2	99,28	99,28	99,28	99,28
Experimento 3	99,38	99,54	99,22	99,38
Experimento 4	99,60	99,80	99,40	99,60
Média ± DP	**99,49 ± 0,17**	**99,53 ± 0,18**	**99,45 ± 0,27**	**99,49 ± 0,17**

6.2.1 Métricas dos Experimentos - Dados Externos

Experimento 1 destacou-se pela acurácia de teste com dados externos (99,70%) e recall (99,90%). Esses resultados tornam esta configuração adequada para cenários onde a identificação completa de casos positivos é prioritária, garantindo cobertura das rachaduras presentes e minimizando falhas na classificação.

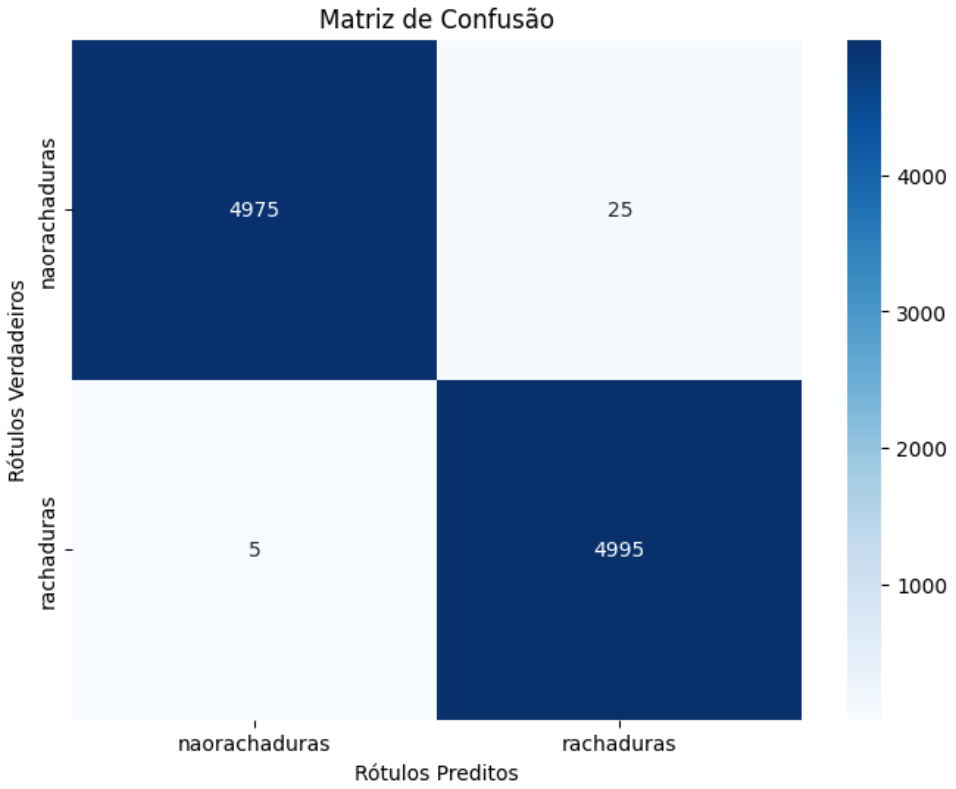


Figura 18 – Matriz de confusão do Experimento 1.

A matriz de confusão da Figura 18 ilustra o desempenho no Experimento 1, dividindo os resultados entre as classes "não rachaduras" e "rachaduras". Esta matriz permite observar a quantidade de acertos e erros em relação à classificação de cada classe:

- **4975** imagens de "não rachaduras" foram corretamente classificadas como "não rachaduras" (verdadeiros negativos).
- **25** imagens de "não rachaduras" foram incorretamente classificadas como "rachaduras" (falsos positivos).
- **5** imagens de "rachaduras" foram incorretamente classificadas como "não rachaduras" (falsos negativos).
- **4995** imagens de "rachaduras" foram corretamente classificadas como "rachaduras" (verdadeiros positivos).

Os resultados demonstram desempenho consistente:

- A **precisão** para a classe "rachaduras" é de 99,50%, indicando que a maioria das imagens classificadas como rachaduras realmente pertencem a essa classe.
- O **recall** para a classe "rachaduras" é de 99,90%, demonstrando que o modelo conseguiu identificar quase todas as imagens de rachaduras presentes no conjunto de teste.
- A **acurácia geral** foi de 99,70%, indicando que a maior parte das classificações foi correta.

A taxa de acurácia e os valores de falsos positivos e falsos negativos indicam que a WellNet é confiável para classificação de rachaduras em superfícies de concreto, aspecto importante para garantir integridade estrutural e segurança em aplicações práticas.

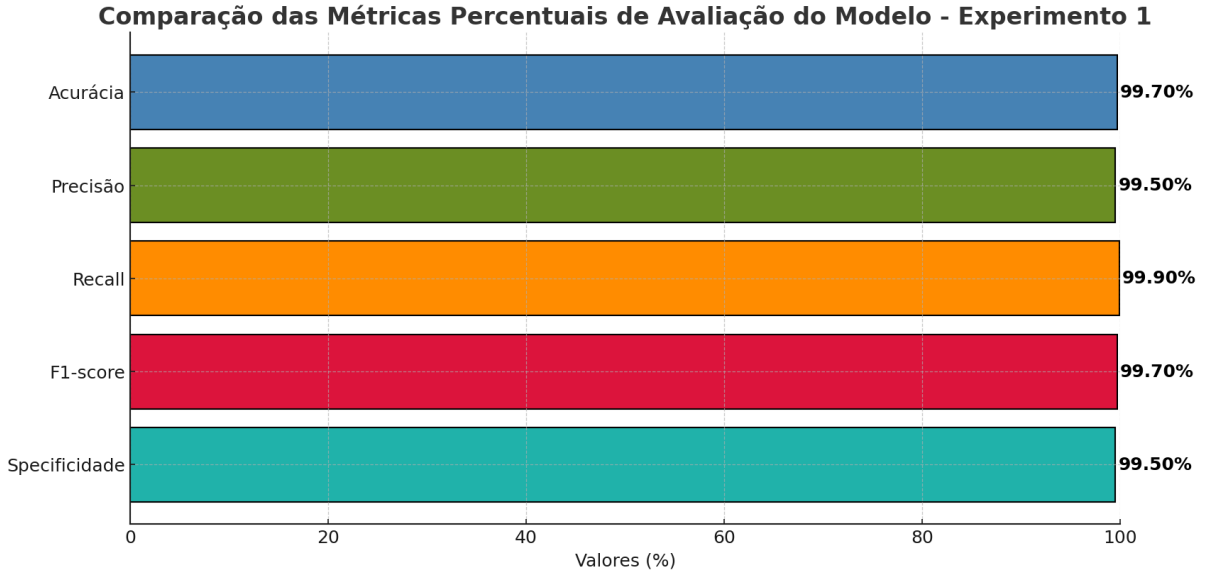


Figura 19 – Visão geral das métricas de desempenho do Experimento 1.

A Figura 20 ilustra a Curva ROC (Receiver Operating Characteristic) do Experimento 1, representação gráfica da relação entre a taxa de verdadeiros positivos (sensibilidade) e a taxa de falsos positivos em diferentes limiares de decisão. A curva ROC avalia a capacidade de distinguir entre as classes "rachaduras" e "não rachaduras", proporcionando visão completa sobre o desempenho em termos de separabilidade das classes.

A Área sob a Curva (AUC) alcançada é de **0,9999**, indicando desempenho quase perfeito na discriminação entre as duas classes. Este valor próximo de 1 demonstra que o modelo possui capacidade de identificar corretamente tanto as imagens de "rachaduras" quanto as de "não rachaduras", minimizando a ocorrência de falsos positivos e falsos negativos.

Este resultado reforça a estabilidade do modelo, evidenciando eficácia para detectar e classificar rachaduras com nível de precisão adequado. A capacidade de manter desempenho consistente em diferentes condições de teste demonstra que o modelo pode ser aplicado

de forma confiável em diversos ambientes e tipos de superfícies de concreto. Esta característica contribui para a manutenção preventiva, permitindo a identificação precoce de problemas estruturais antes que se tornem críticos, e evita falhas estruturais que poderiam comprometer a segurança de edificações e infraestruturas. A AUC ressalta o potencial para aplicação em cenários práticos de monitoramento contínuo, onde a precisão na detecção é importante para evitar tanto os custos associados a manutenções desnecessárias quanto os riscos de não detectar rachaduras reais. Essa confiabilidade na classificação contribui diretamente para garantir a longevidade das infraestruturas monitoradas, reduzindo a necessidade de intervenções emergenciais e possibilitando planejamento das atividades de manutenção.

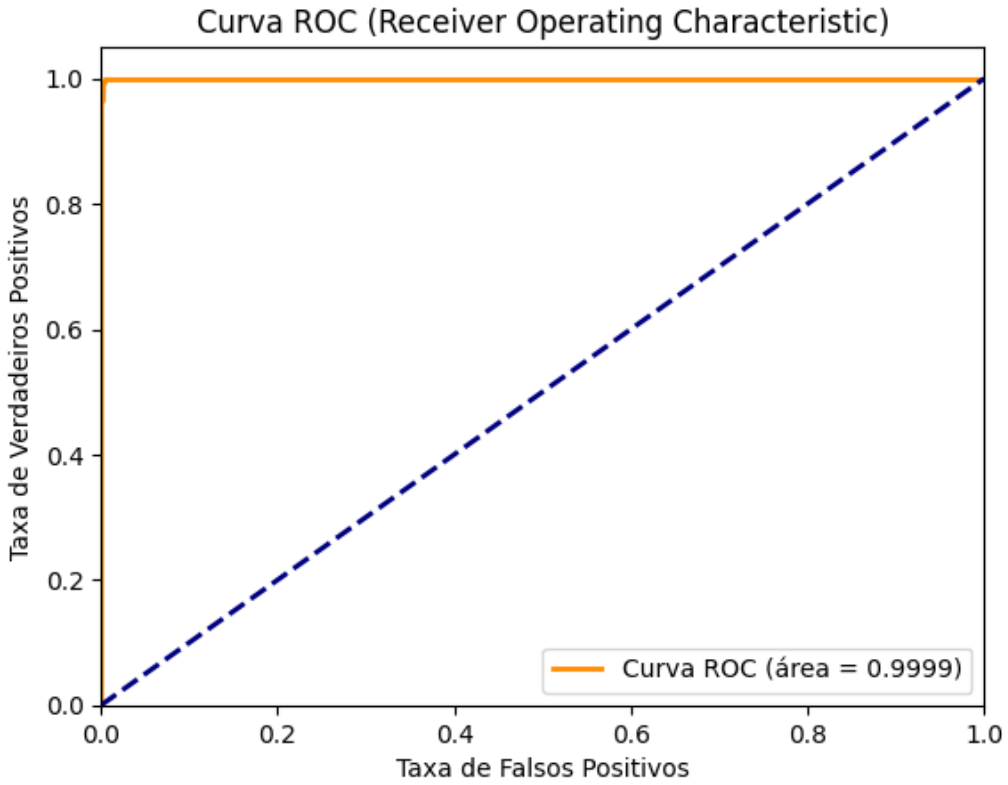


Figura 20 – Curva ROC do Experimento 1.

Experimento 2 apresentou o menor desempenho, mesmo com o uso de aumento de dados. Isso demonstra que o aumento de dados, isoladamente, não garante melhor desempenho se não for balanceado com outros parâmetros, como taxa de aprendizado e batch size. A eficiência do treinamento depende da interação entre esses elementos, e uma configuração inadequada pode levar a desempenho abaixo do esperado, mesmo com técnicas de enriquecimento do conjunto de dados.

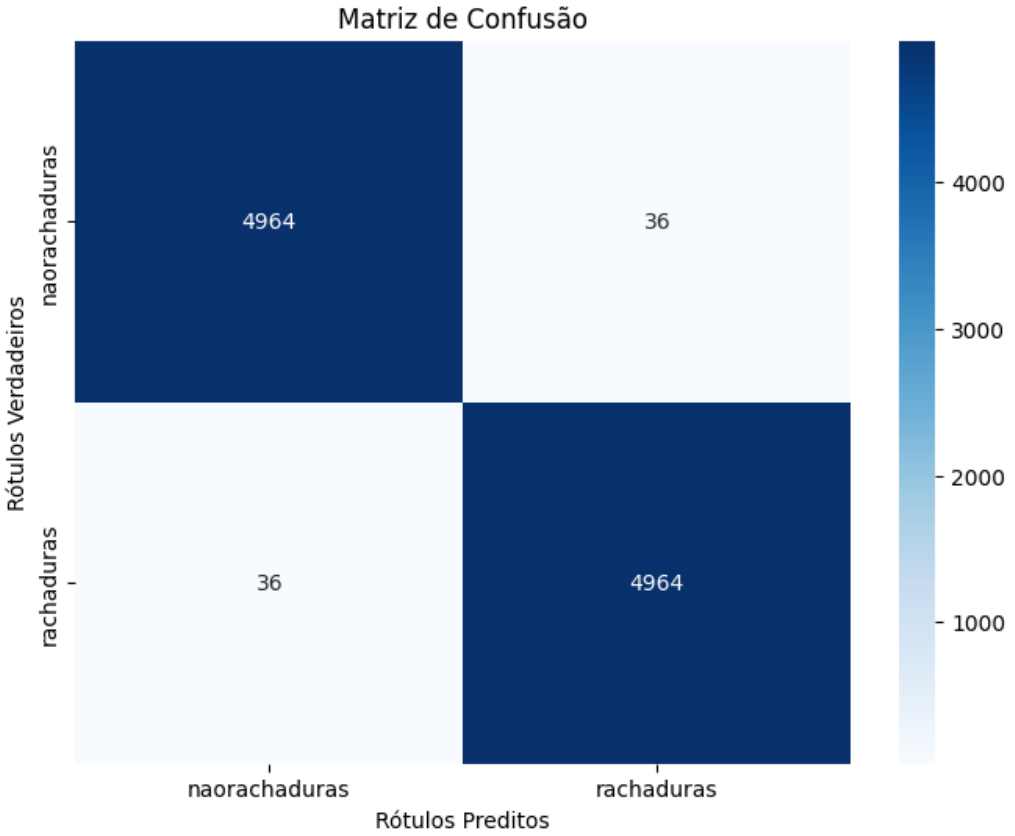


Figura 21 – Matriz de confusão do Experimento 2.

A matriz de confusão da Figura 21 ilustra o desempenho no Experimento 2, dividindo os resultados entre as classes “não rachaduras” e “rachaduras”:

- **4964** imagens de “não rachaduras” foram corretamente classificadas como “não rachaduras” (verdadeiros negativos).
- **36** imagens de “não rachaduras” foram incorretamente classificadas como “rachaduras” (falsos positivos).
- **36** imagens de “rachaduras” foram incorretamente classificadas como “não rachaduras” (falsos negativos).
- **4964** imagens de “rachaduras” foram corretamente classificadas como “rachaduras” (verdadeiros positivos).

Os resultados mostram desempenho sólido, mas com alguns pontos de atenção:

- A **precisão** para a classe “rachaduras” é de 99,28%, indicando que a maioria das imagens classificadas como rachaduras realmente pertencem a essa classe.
- O **recall** para a classe “rachaduras” é de 99,28%, demonstrando que o modelo conseguiu identificar a maior parte das imagens de rachaduras presentes no conjunto de teste.

- A **acurácia geral** foi de 99,28%, indicando que a maior parte das classificações foi correta.

Comparado ao Experimento 1, houve aumento no número de falsos positivos e falsos negativos, sugerindo que, embora o aumento de dados tenha ajudado a reduzir o sobreajuste, também pode ter levado a maior variabilidade no comportamento. Este fato é indicado pelos 36 falsos positivos e 36 falsos negativos, em contraste com os valores menores obtidos no Experimento 1.

Embora a taxa de acurácia permaneça em nível adequado, esse aumento na quantidade de erros sugere que as técnicas de aumento de dados podem ter introduzido ruído suficiente para diminuir a precisão em alguns casos, possivelmente criando variações que confundiram o modelo.

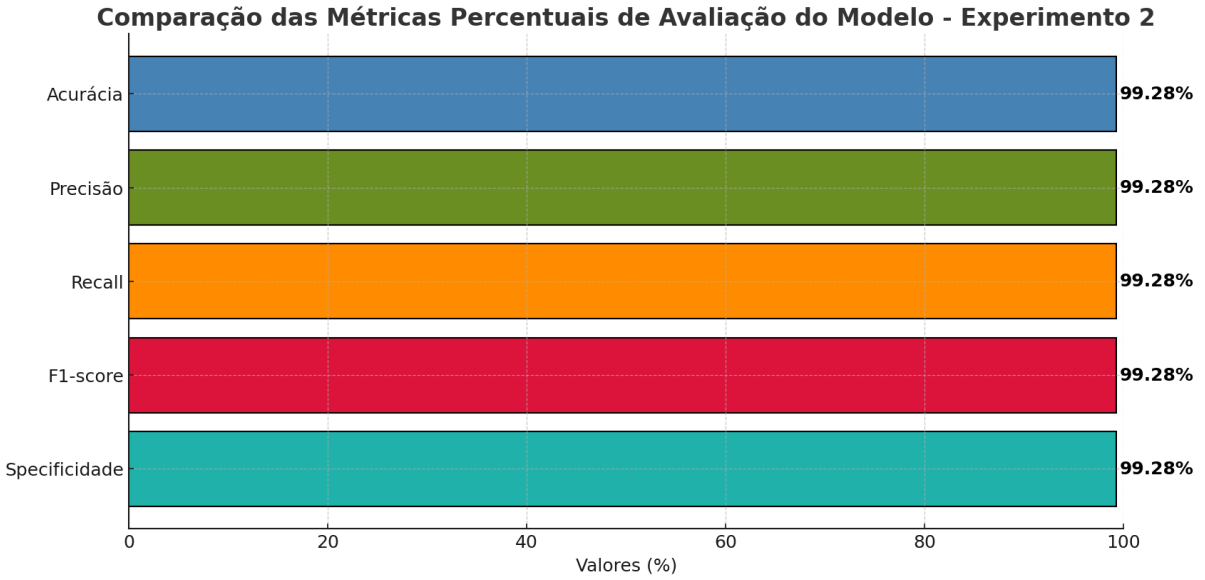


Figura 22 – Visão geral das métricas de desempenho do Experimento 2.

A Figura 23 apresenta a Curva ROC do Experimento 2, mostrando a relação entre a taxa de verdadeiros positivos e a taxa de falsos positivos, proporcionando análise detalhada sobre o desempenho em diferentes limiares de decisão. A Área sob a Curva (AUC) deste experimento foi de **0,9998**.

Este valor próximo de 1 indica que o modelo possui capacidade de discriminar entre as classes “rachaduras” e “não rachaduras”, minimizando tanto os falsos positivos quanto os falsos negativos. Apesar de ligeira diminuição em comparação com o Experimento 1 (AUC de 0,9999), o desempenho permanece eficaz na tarefa de classificação automatizada de rachaduras.

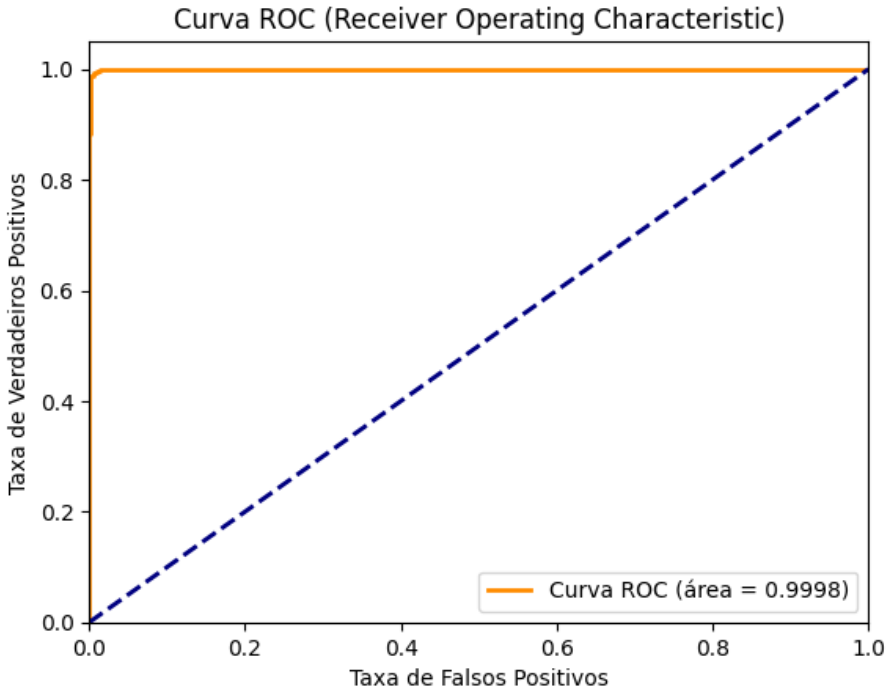


Figura 23 – Curva ROC do Experimento 2.

Experimento 3 se destacou pelo equilíbrio entre erro de validação e acurácia de teste, graças ao uso de batch size maior e aumento de dados. Representa boa escolha para situações onde se deseja generalização e precisão (99,54%).

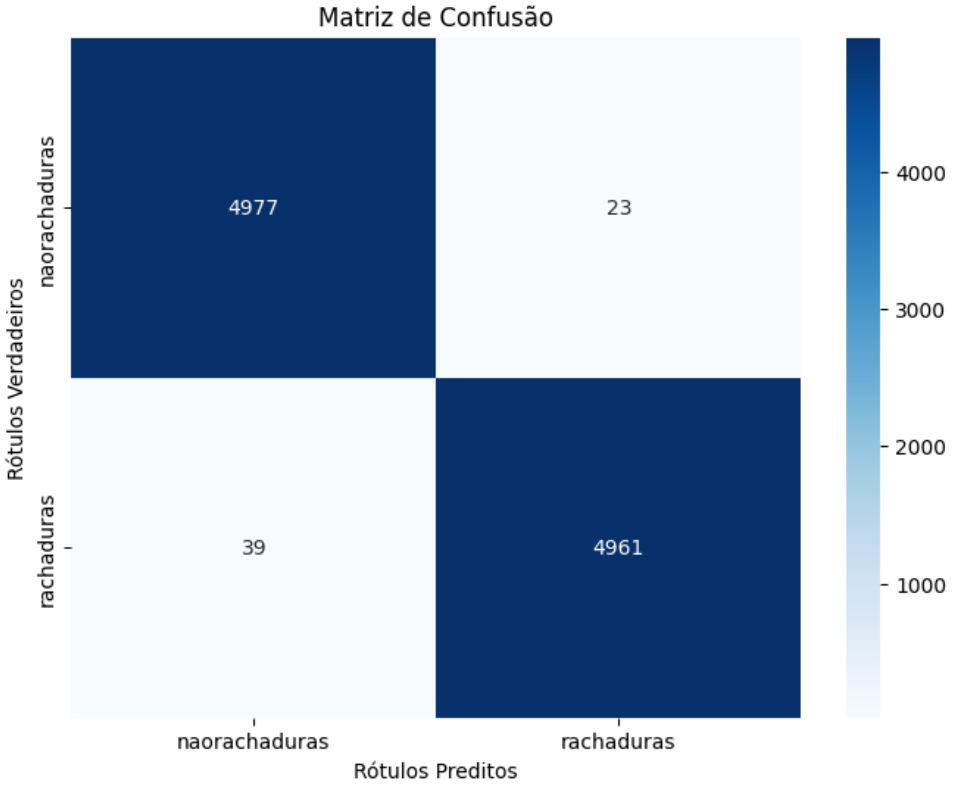


Figura 24 – Matriz de confusão do Experimento 3.

A matriz de confusão da Figura 24 ilustra o desempenho no Experimento 3:

- **4977** imagens de "não rachaduras" foram corretamente classificadas como "não rachaduras" (verdadeiros negativos).
- **23** imagens de "não rachaduras" foram incorretamente classificadas como "rachaduras" (falsos positivos).
- **39** imagens de "rachaduras" foram incorretamente classificadas como "não rachaduras" (falsos negativos).
- **4961** imagens de "rachaduras" foram corretamente classificadas como "rachaduras" (verdadeiros positivos).

Os resultados mostram desempenho consistente:

- A **precisão** para a classe "rachaduras" é de 99,54%, indicando que a maioria das imagens classificadas como rachaduras realmente pertencem a essa classe.
- O **recall** para a classe "rachaduras" é de 99,22%, demonstrando que o modelo conseguiu identificar a maior parte das imagens de rachaduras presentes no conjunto de teste.
- A **acurácia geral** foi de 99,38%, indicando que a maior parte das classificações foi correta.

Comparado ao Experimento 1, houve aumento no número de falsos negativos, sugerindo que o modelo ainda enfrenta desafios na identificação de todas as rachaduras, apesar do uso de técnicas de aumento de dados. Este fato é indicado pelos 39 falsos negativos observados, em contraste com valores menores nos experimentos anteriores.

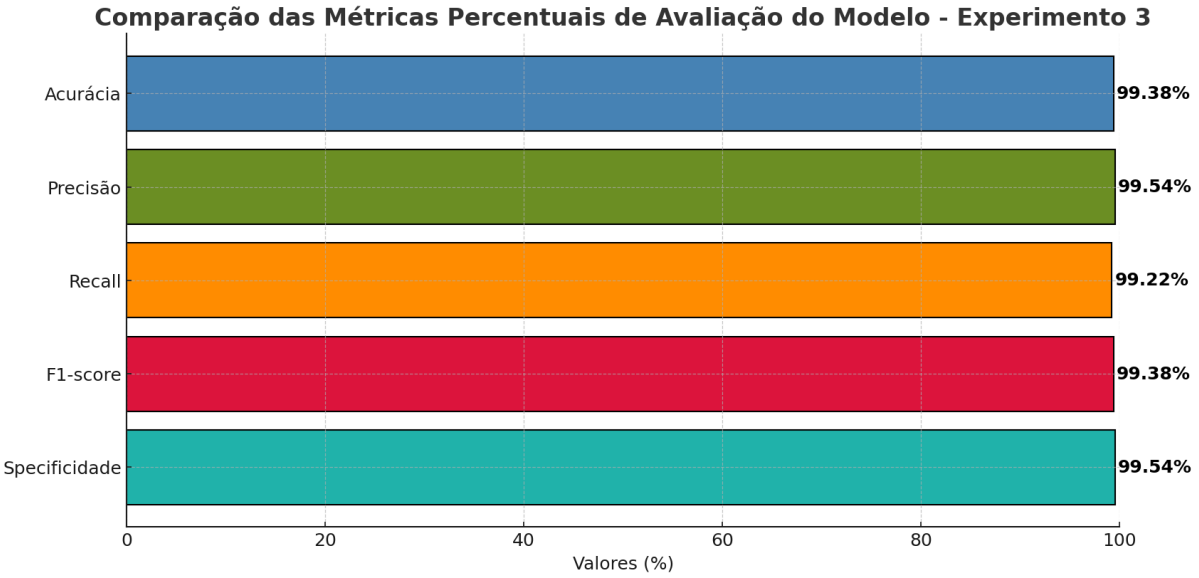


Figura 25 – Visão geral das métricas de desempenho do Experimento 3.

A Figura 26 apresenta a Curva ROC do Experimento 3, com Área sob a Curva (AUC) de **0,9998**. Este valor próximo de 1 indica capacidade de discriminar entre as classes "rachaduras" e "não rachaduras", demonstrando eficiência em minimizar falsos positivos e falsos negativos.

Embora tenha ocorrido diminuição em relação ao Experimento 1 (cuja AUC foi de 0,9999), o desempenho continua sendo eficaz para a classificação de rachaduras. A diferença de apenas 0,0001 na AUC representa variação mínima que não compromete a capacidade de discriminação entre as classes. Esta capacidade evidencia a estabilidade em condições variadas, garantindo a segurança e integridade das estruturas monitoradas. O modelo demonstra consistência na identificação de rachaduras mesmo com as modificações nos parâmetros de treinamento, indicando que a arquitetura proposta mantém desempenho adequado independentemente das configurações específicas utilizadas.

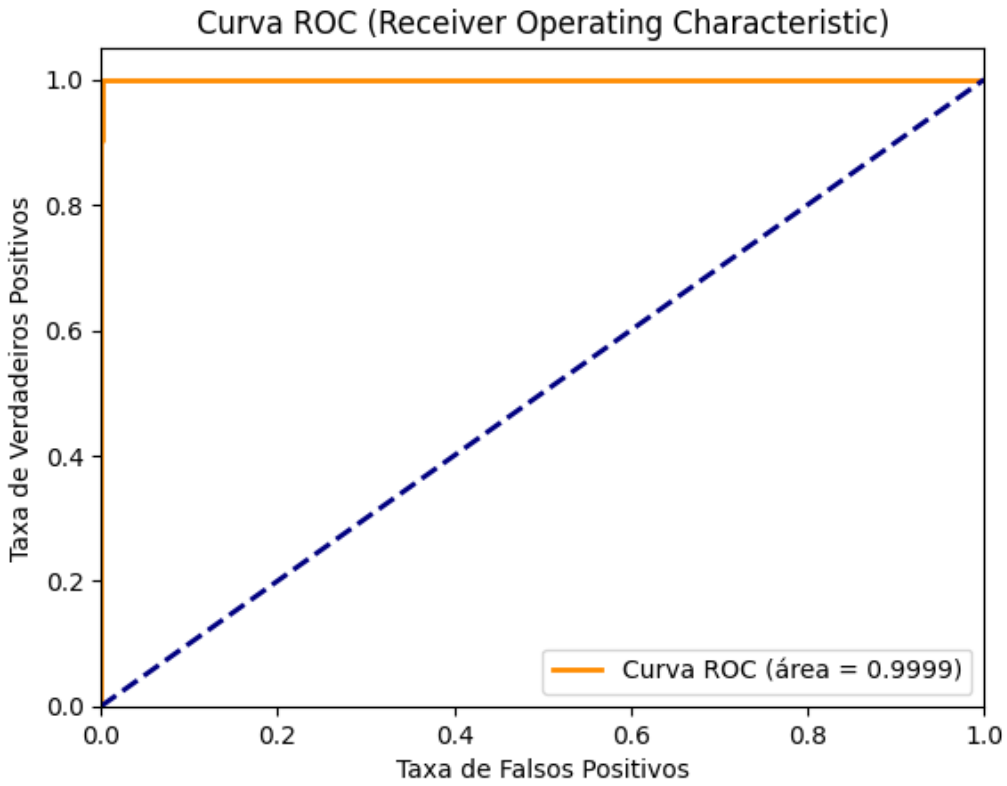


Figura 26 – Curva ROC do Experimento 3.

Experimento 4 obteve a acurácia média de teste (99,60%) e a precisão (99,80%). Esta configuração apresentou o menor número de falsos positivos (apenas 10) entre todos os experimentos realizados, demonstrando capacidade superior na redução de alarmes falsos. Esta configuração pode ser preferível quando a minimização de falsos positivos é prioritária.

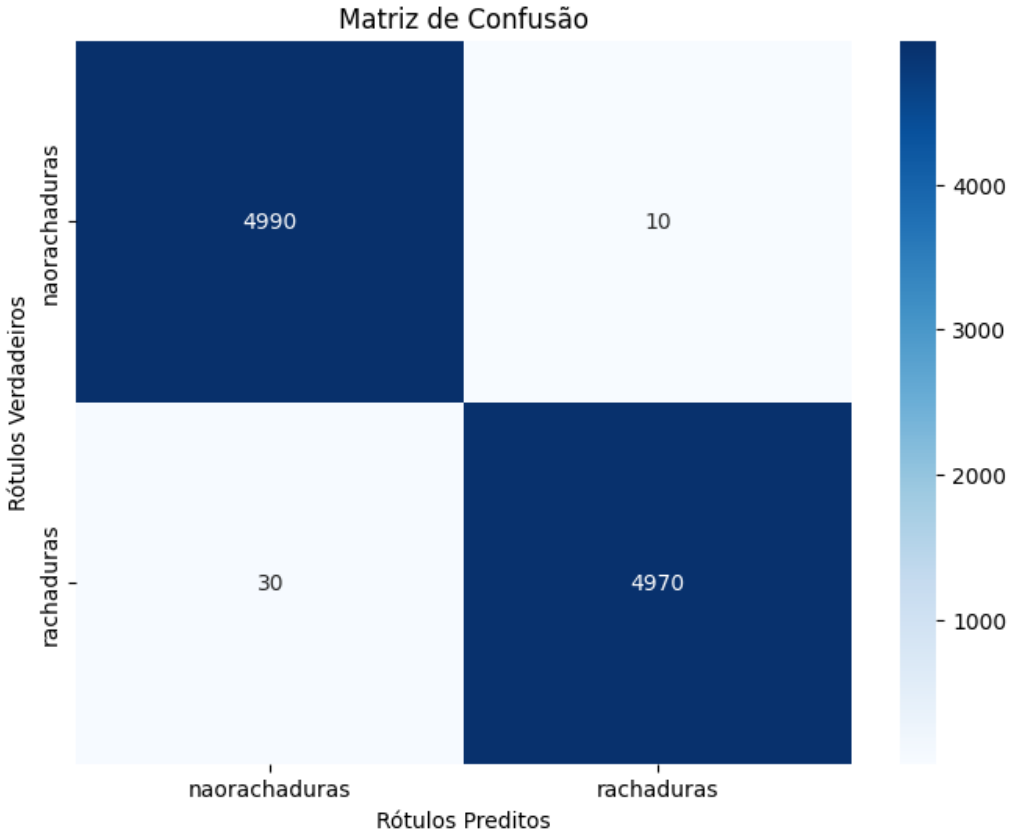


Figura 27 – Matriz de confusão do Experimento 4.

A matriz de confusão da Figura 27 ilustra o desempenho no Experimento 4:

- **4990** imagens de "não rachaduras" foram corretamente classificadas como "não rachaduras" (verdadeiros negativos).
- **10** imagens de "não rachaduras" foram incorretamente classificadas como "rachaduras" (falsos positivos).
- **30** imagens de "rachaduras" foram incorretamente classificadas como "não rachaduras" (falsos negativos).
- **4970** imagens de "rachaduras" foram corretamente classificadas como "rachaduras" (verdadeiros positivos).

Os resultados demonstram desempenho consistente, com avanços em comparação aos experimentos anteriores:

- A **precisão** para a classe "rachaduras" é de 99,80%, indicando que a maioria das imagens classificadas como rachaduras realmente pertencem a essa classe. Este valor representa a maior precisão alcançada entre todos os experimentos realizados, demonstrando capacidade superior na redução de falsos alarmes.
- O **recall** para a classe "rachaduras" é de 99,40%, demonstrando que o modelo conseguiu identificar a maior parte das imagens de rachaduras presentes no conjunto

de teste. Embora ligeiramente inferior ao Experimento 1 (99,90%), este valor ainda garante detecção adequada das rachaduras reais presentes nas imagens.

- A **acurácia geral** foi de 99,60%, indicando que a maior parte das classificações foi correta. Esta métrica posiciona o Experimento 4 como a segunda melhor configuração em termos de acurácia global, superando os Experimentos 2 e 3.

No Experimento 4, foram observados 10 falsos positivos e 30 falsos negativos. Esses valores refletem capacidade de classificação adequada, embora ainda existam desafios na identificação correta de todas as rachaduras. A distribuição dos erros mostra uma proporção de 3:1 entre falsos negativos e falsos positivos, indicando que o modelo tende a ser mais conservador na classificação de rachaduras. Comparado aos demais experimentos, este apresentou o menor número total de erros após o Experimento 1, com apenas 40 classificações incorretas em 10.000 imagens testadas. O desempenho mostra que o modelo alcançou consistência na classificação, mantendo equilíbrio entre acertos e erros, com taxa de erro total de 0,40% do conjunto de dados avaliado.

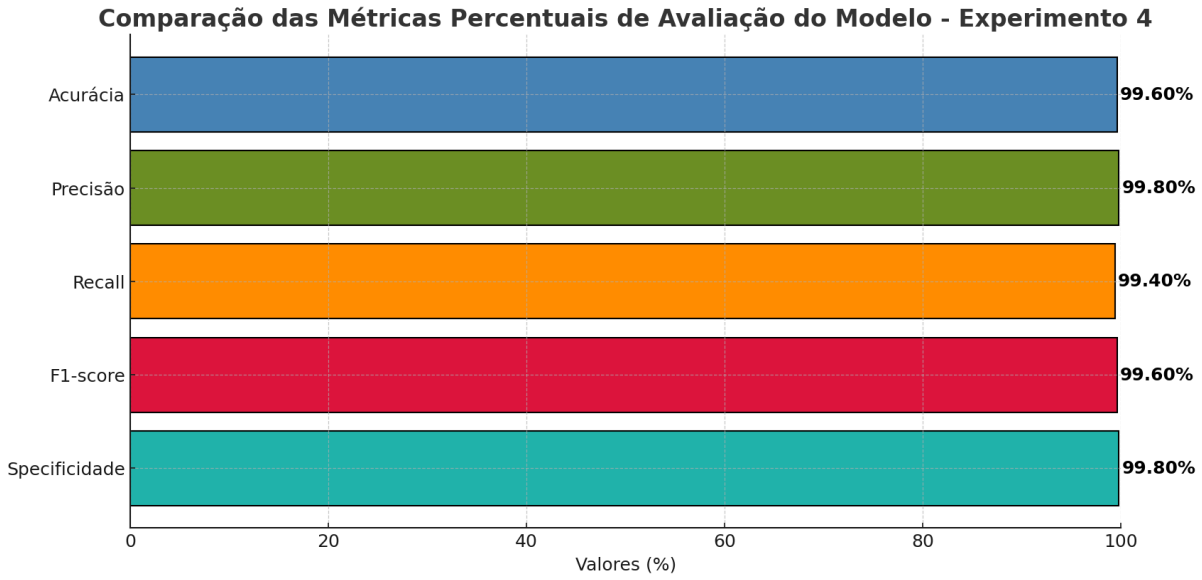


Figura 28 – Visão geral das métricas de desempenho do Experimento 4.

A Figura 29 apresenta a Curva ROC do Experimento 4, que possui Área sob a Curva (AUC) de **0,9999**. Este valor, próximo de 1, evidencia a capacidade de discriminar entre as classes “rachaduras” e “não rachaduras”, minimizando a ocorrência de falsos positivos e falsos negativos.

O valor de AUC do Experimento 4 é idêntico ao do Experimento 1, mostrando que o desempenho se manteve eficaz em diferentes condições de treinamento. Comparado aos Experimentos 2 e 3, que obtiveram AUC de **0,9998**, o Experimento 4 reafirma a capacidade de manter precisão na classificação.

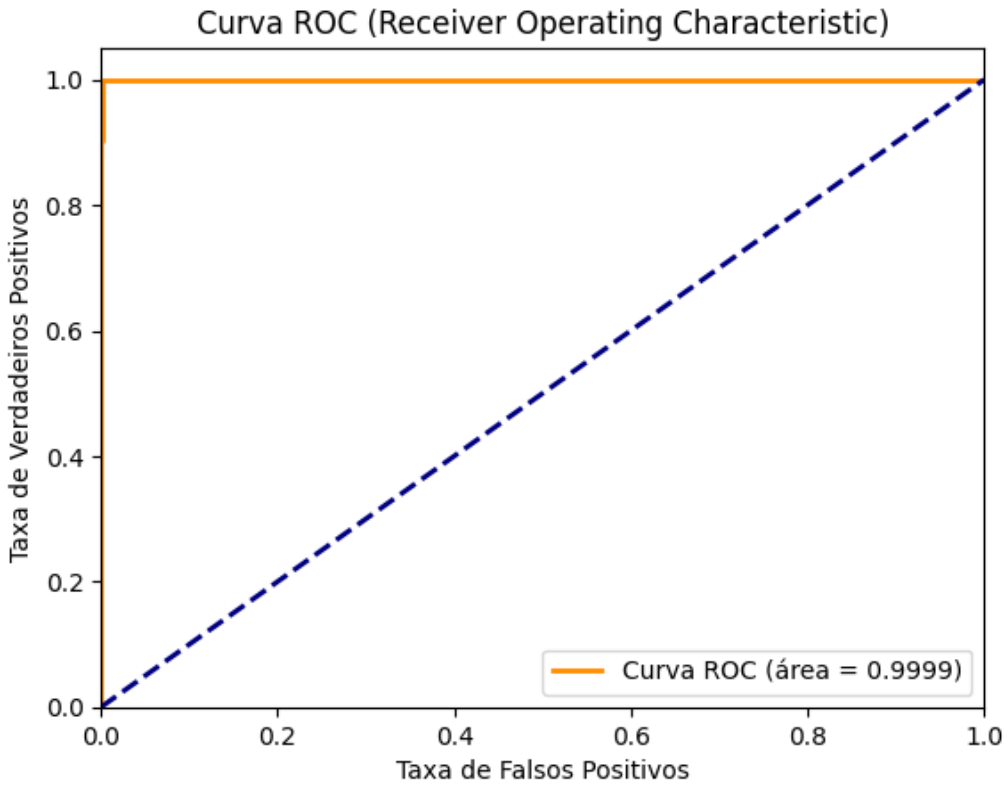


Figura 29 – Curva ROC do Experimento 4.

6.3 Comparativo de Resultados com Trabalhos Relacionados

Nesta seção, comparamos os resultados da WellNet com outros trabalhos relevantes na literatura, assim como outros que também utilizam abordagens híbridas para a classificação de rachaduras.

6.3.1 Metodologia de Avaliação

Para avaliar o desempenho do modelo proposto, realizamos uma análise comparativa abrangente entre o WellNet e outros modelos de referência na literatura. Esta seção apresenta os resultados de maneira detalhada, destacando as diferenças quantitativas entre os modelos e a magnitude dos resultados obtidos.

6.3.2 Métricas de Avaliação

A avaliação quantitativa foi realizada através de quatro métricas fundamentais que, em conjunto, fornecem uma visão holística sobre o desempenho dos modelos:

- **Acurácia**: Representa a proporção total de previsões corretas (tanto positivas quanto negativas) sobre o número total de casos avaliados.

$$\text{Acurácia} = \frac{TP + TN}{TP + TN + FP + FN}$$

onde TP representa os verdadeiros positivos, TN os verdadeiros negativos, FP os falsos positivos e FN os falsos negativos. Esta métrica fornece uma visão geral do desempenho do modelo, indicando qual percentual de todas as classificações foi realizado corretamente.

- **Precisão**: Indica a proporção de identificações positivas corretas dentre todos os casos classificados como positivos pelo modelo.

$$\text{Precisão} = \frac{TP}{TP + FP}$$

A precisão é crítica para minimizar alarmes falsos, pois mede a confiabilidade das classificações positivas. Um valor alto indica que quando o modelo classifica uma imagem como contendo rachadura, existe alta probabilidade de estar correto.

- **Recall (Sensibilidade)**: Mede a capacidade do modelo em identificar corretamente todos os casos positivos do conjunto de dados.

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

O recall é fundamental para aplicações de segurança estrutural, pois avalia quantas rachaduras reais foram detectadas pelo modelo. Um recall elevado garante que poucas rachaduras passem despercebidas, aspecto crucial para a integridade das estruturas monitoradas.

- **F1-Score**: Representa a média harmônica entre precisão e recall, fornecendo uma métrica balanceada que é particularmente útil quando as classes no conjunto de dados são desequilibradas.

$$\text{F1-Score} = 2 \times \frac{\text{Precisão} \times \text{Recall}}{\text{Precisão} + \text{Recall}}$$

Esta métrica combina tanto a capacidade de detectar rachaduras quanto a confiabilidade das detecções, oferecendo um indicador único que equilibra a necessidade de alta sensibilidade com baixa taxa de falsos alarmes.

6.3.3 Resultados Comparativos

Os resultados comparativos entre o modelo WellNet e demais modelos na literatura são apresentados na Tabela 11. É importante ressaltar que esta comparação tem caráter puramente referencial e contextual, não constituindo uma avaliação estatística rigorosa,

uma vez que não foram realizados testes estatísticos entre os modelos e os dados provêm de diferentes estudos com metodologias experimentais potencialmente distintas.

Tabela 11 – Comparação dos Resultados da WellNet com Outros Modelos

Modelo	**Acurácia (%)**	**Precisão (%)**	**Recall (%)**	**F1-Score (%)**
WellNet Modelo Proposto	99.70	99.50	99.90	99.70
CNN (DINIZ et al., 2023)	99.48	99.40	99.56	99.48
DeepCrack (ZOU et al., 2019)	98.70	98.50	98.80	98.65
AlexNet Pré-treinada (KARMOKAR et al., 2020)	98.40	98.87	98.00	98.43
ViT+Detector Textura (SHAHIN et al., 2024)	99.80	99.80	99.80	99.80

6.3.4 Análise Quantitativa dos Resultados

A Tabela 12 apresenta as diferenças absolutas entre WellNet e os demais modelos. Estas diferenças devem ser interpretadas como indicadores referenciais, considerando que os valores provêm de estudos independentes sem padronização metodológica.

Tabela 12 – Diferenças Absolutas entre WellNet e Outros Modelos (em pontos percentuais)

Modelo vs WellNet	**Acurácia**	**Precisão**	**Recall**	**F1-Score**
CNN (Diniz et al., 2023)	+0.22	+0.10	+0.34	+0.22
DeepCrack	+1.00	+1.00	+1.10	+1.05
AlexNet	+1.30	+0.63	+1.90	+1.27
ViT	-0.10	-0.30	+0.10	-0.10

6.3.5 Análise de Ranking por Métrica

A Tabela 13 apresenta o posicionamento relativo dos modelos para cada métrica individual, baseado nos valores pontuais disponíveis.

Tabela 13 – Posicionamento dos Modelos por Métrica Individual

Posição	**Acurácia**	**Precisão**	**Recall**	**F1-Score**
1º	ViT (99.80)	ViT (99.80)	**WellNet (99.90)**	ViT (99.80)
2º	**WellNet (99.70)**	**WellNet (99.50)**	ViT (99.80)	**WellNet (99.70)**
3º	CNN (99.48)	CNN (99.40)	CNN (99.56)	CNN (99.48)
4º	DeepCrack (98.70)	AlexNet (98.87)	DeepCrack (98.80)	DeepCrack (98.65)
5º	AlexNet (98.40)	DeepCrack (98.50)	AlexNet (98.00)	AlexNet (98.43)

6.3.6 Análise das Diferenças Observadas

6.3.6.1 Comparação com CNN (DINIZ et al., 2023)

A WellNet apresenta desempenho numericamente comparável ao CNN de Diniz et al. (2023), com diferenças de 0.22 pontos percentuais na acurácia (99.70% vs 99.48%) e F1-Score (99.70% vs 99.48%). A diferença mais notável ocorre no recall, onde WellNet registra 99.90% versus 99.56% do CNN, representando diferença de 0.34 pontos percentuais. A precisão apresenta diferença de 0.10 pontos percentuais favorável ao WellNet (99.50% vs 99.40%).

6.3.6.2 Comparação com DeepCrack (ZOU et al., 2019)

As diferenças numéricas observadas variam entre 1.00 e 1.10 pontos percentuais em todas as métricas, com WellNet apresentando valores numericamente maiores. O recall mostra a maior diferença (1.10 pontos percentuais), seguido por acurácia e precisão (1.00 ponto percentual cada), e F1-Score (1.05 pontos percentuais).

6.3.6.3 Comparação com AlexNet Pré-treinada (KARMOKAR et al., 2020)

A AlexNet apresenta precisão de 98.87% comparado aos 99.50% da WellNet (diferença de 0.63 pontos percentuais). O recall mostra diferença mais acentuada: 98.00% versus 99.90% (1.90 pontos percentuais). A acurácia e F1-Score apresentam diferenças de 1.30 e 1.27 pontos percentuais, respectivamente.

6.3.6.4 Comparação com ViT+Detector Textura (SHAHIN et al., 2024)

O ViT+Detector Textura apresenta desempenho numericamente mais elevado em acurácia (99.80% vs 99.70%), precisão (99.80% vs 99.50%) e F1-Score (99.80% vs 99.70%). A WellNet registra valor ligeiramente maior em recall (99.90% versus 99.80%). As diferenças variam entre 0.10 e 0.30 pontos percentuais.

6.3.7 Limitações da Análise Comparativa

É fundamental considerar que esta análise comparativa possui limitações metodológicas significativas que restringem a interpretação dos resultados:

- **Ausência de testes estatísticos**: Não foram realizados testes de significância estatística entre os modelos
- **Diferentes conjuntos de dados**: Os estudos podem ter utilizado bases de dados distintas, com características variadas

- **Metodologias experimentais heterogêneas**: Condições de treinamento, validação e teste podem diferir entre os trabalhos
- **Valores pontuais**: Os resultados não incluem intervalos de confiança ou medidas de incerteza
- **Variabilidade experimental**: Pequenas diferenças numéricas podem estar dentro da margem de variação natural dos experimentos

6.3.8 Análise de Perfis de Desempenho

A análise dos valores revela perfis de desempenho distintos para cada abordagem:

- **WellNet**: Apresenta desempenho equilibrado com destaque para recall elevado (99.90%)
- **ViT+Detector Textura**: Demonstra consistência com valores uniformes de 99.80% em três métricas
- **CNN (Diniz et al.)**: Mostra performance próxima à WellNet em todas as métricas
- **DeepCrack**: Apresenta valores intermediários entre 98.50% e 98.80%
- **AlexNet**: Exibe maior variabilidade, com precisão de 98.87% mas recall de 98.00%

6.3.9 Posicionamento da WellNet no Contexto Atual

Com base na análise comparativa referencial, a WellNet se posiciona como uma alternativa viável para classificação de rachaduras, apresentando características específicas que podem ser relevantes conforme a aplicação:

- **Recall elevado**: O valor de 99.90% pode ser vantajoso em aplicações que priorizam a detecção completa de casos positivos
- **Desempenho equilibrado**: Mantém bom desempenho em todas as métricas sem sacrificar significativamente nenhuma delas
- **Competitividade**: Demonstra-se comparável aos modelos mais avançados da literatura atual
- **Compacticidade**: Sua arquitetura permite implementação em dispositivos com recursos limitados

6.3.10 Considerações sobre Aplicabilidade

A escolha entre os modelos deve considerar os requisitos específicos da aplicação:

- Para aplicações que demandam máxima detecção de rachaduras, o alto recall da WellNet pode ser preferível
- Em cenários onde a consistência entre todas as métricas é prioritária, o ViT+Detector Textura pode ser mais adequado
- Para implementações com restrições computacionais, a compacticidade da WellNet oferece vantagens práticas
- A proximidade de desempenho entre WellNet e CNN sugere que ambos são alternativas válidas

6.4 Considerações Finais

A análise comparativa referencial demonstra que a WellNet representa uma alternativa competitiva no cenário atual de classificação de rachaduras. O modelo se destaca particularmente pelo recall elevado (99.90%), posicionando-se como uma opção viável para aplicações que priorizam a detecção completa de casos positivos.

A proximidade de desempenho com os modelos mais avançados da literatura, combinada com sua arquitetura compacta e eficiente, estabelece a WellNet como uma contribuição válida para o estado da arte. A análise revela que não existe um modelo universalmente superior, mas sim alternativas com características específicas adequadas a diferentes cenários de aplicação.

É importante ressaltar que as comparações apresentadas têm caráter puramente referencial, uma vez que não foram realizados testes estatísticos rigorosos entre os modelos. As diferenças numéricas observadas devem ser interpretadas como indicadores de posicionamento relativo, não como evidência de superioridade estatística significativa.

A metodologia experimental envolveu quatro configurações distintas de hiperparâmetros, revelando insights importantes sobre o comportamento do modelo. O Experimento 1 emergiu como a configuração mais eficaz, demonstrando a importância da otimização de hiperparâmetros para maximizar o desempenho.

A aplicabilidade prática da WellNet é ampliada por sua eficiência computacional e capacidade de implementação em dispositivos móveis e embarcados. Esta característica elimina dependências de infraestrutura robusta e viabiliza inspeções automatizadas em tempo real, aspectos cruciais para aplicações industriais.

Os resultados validam a eficácia da arquitetura híbrida ViT-CNN proposta, demonstrando que a combinação dessas tecnologias pode resultar em modelos robustos e versáteis. A WellNet consegue equilibrar a captura de padrões locais detalhados com relações espaciais de longo alcance, oferecendo uma solução balanceada para diferentes condições e tipos de superfície.

Em síntese, a WellNet se estabelece como uma alternativa sólida e prática para a classificação de rachaduras, com potencial para aplicação em diversos cenários industriais onde a detecção precisa e eficiente de rachaduras é fundamental para a segurança e manutenção de estruturas.

7 Conclusões e Trabalhos Futuros

Neste capítulo apresenta as conclusões obtidas com o desenvolvimento da WellNet, destacando os principais resultados e contribuições do trabalho. Além disso, são discutidas possíveis melhorias e trabalhos futuros que poderão ser realizados com base nas limitações e oportunidades identificadas ao longo do estudo. O objetivo é fornecer um panorama dos benefícios alcançados e apontar caminhos para futuras investigações e aprimoramentos da rede WellNet.

7.1 Conclusões

A WellNet demonstrou eficácia como arquitetura híbrida ViT→CNN para classificação automatizada de rachaduras e fissuras em superfícies de concreto. A combinação estratégica do ViT *Tiny* pré-treinado (5.486.016 parâmetros congelados) com camadas convolucionais especializadas (683.522 parâmetros treináveis) resultou em modelo compacto de 26,4 MB que oferece vantagens significativas de portabilidade e *deployment* distribuído.

Os resultados experimentais validaram a eficácia da arquitetura ViT→CNN, demonstrando que o *Transfer Learning* inteligente otimiza recursos computacionais através da combinação de representações semânticas globais pré-aprendidas com especialização local mínima. A WellNet se estabelece como uma ferramenta de auxílio para inspeção de rachaduras, desenvolvida para apoiar profissionais da engenharia civil em suas atividades de monitoramento estrutural, tornando o acesso fácil à tecnologia de visão computacional através de sua arquitetura compacta e eficiente.

Os resultados experimentais validaram a eficiência da arquitetura através de validação cruzada *k-fold* 5 com 20.000 imagens do *dataset* Özgenel. O Experimento 1 alcançou desempenho superior: acurácia de 99,70%, precisão de 99,50%, *recall* de 99,90% e *F1-score* de 99,70%. A análise comparativa com cinco modelos de referência posicionou a WellNet competitivamente, destacando-se pelo *recall* superior (99,90%) entre todos os modelos avaliados, superando ligeiramente o ViT+Detector Textura (99,80%).

A eficiência computacional representa diferencial significativo da arquitetura. A WellNet demonstrou redução de 88,5% no tamanho comparado à AlexNet pré-treinada (26,4 MB versus 229 MB), viabilizando *deployment* em dispositivos com recursos limitados. Embora apresente *overhead* temporal de 21,7% no treinamento (28 versus 23 minutos) e *throughput* ligeiramente inferior, a diferença marginal de 0,10 pontos percentuais na acurácia (99,60% versus 99,70%) demonstra manutenção de desempenho competitivo. O tempo de inferência de 16,6 milissegundos por imagem viabiliza processamento em tempo

real, característica crucial para aplicações práticas.

O desenvolvimento do protótipo web utilizando *Flask* demonstrou aplicabilidade prática da solução. A interface permite *upload* de múltiplas imagens com processamento automatizado, oferecendo resultados com indicação de classe predita e nível de confiança. A validação prática com 100 imagens confirmou desempenho temporal consistente, processando o lote em 1,66 segundos.

A distribuição de parâmetros (88,92% congelados, 11,08% treináveis) otimiza o equilíbrio entre capacidade representacional e compactação extrema. Esta configuração permite convergência em 8-10 épocas versus valores típicos superiores a 25 para modelos convencionais, facilitando re-treinamento frequente com dados atualizados de campo sem comprometer recursos computacionais.

A arquitetura compacta viabiliza *deployment* em dispositivos com recursos limitados como *Raspberry Pi*, *smartphones* e sistemas embarcados. Esta característica torna acessível o acesso à tecnologia de inspeção automatizada, permitindo implementação distribuída em infraestrutura civil sem dependência de *hardware* robusto.

As limitações identificadas incluem suporte restrito a imagens estáticas na aplicação web, dependência de conectividade para processamento remoto e necessidade de adaptação para cenários com classes desbalanceadas. O redimensionamento obrigatório para 224×224 *pixels* pode comprometer detalhes em imagens de alta resolução original.

A WellNet estabelece contribuição mensurável para Visão Computacional aplicada à engenharia civil através da combinação eficiente de *Transfer Learning* inteligente com especialização arquitetural mínima. A solução é uma alternativa viável para monitoramento estrutural automatizado em larga escala, oferecendo alternativa especializada em compactação comparada aos modelos puramente convolucionais.

7.2 Trabalhos Futuros

A extensão da WellNet para detecção e/ou segmentação de rachaduras e fissuras representa evolução natural da pesquisa, permitindo localização precisa e quantificação dimensional de defeitos estruturais. A implementação de arquitetura *encoder-decoder* baseada na estrutura híbrida ViT→CNN atual pode preservar eficiência de compactação, enquanto oferece análise espacial detalhada.

A otimização sistemática de hiperparâmetros através de *Bayesian Optimization* ou *Grid Search* pode identificar configurações superiores aos valores empíricos utilizados. A exploração de diferentes taxas de aprendizado, *schedulers* adaptativos e técnicas de regularização avançadas pode melhorar convergência e reduzir o *overhead* temporal observado, mantendo as vantagens de compactação.

A adaptação para cenários com classes desbalanceadas requer implementação de funções de perda especializadas ou técnicas de re-amostragem inteligente. Esta modificação ampliará aplicabilidade em condições reais onde superfícies íntegras predominam significativamente sobre defeitos estruturais.

O desenvolvimento de versão *offline* da aplicação web eliminará dependência de conectividade, viabilizando uso em locais remotos. A implementação de processamento local através de *WebAssembly* ou tecnologias similares pode manter funcionalidade completa sem requisitos de rede.

A integração com sistemas de monitoramento contínuo através de APIs padronizadas facilitará adoção comercial. O desenvolvimento de conectores para plataformas IoT existentes pode automatizar coleta e análise de dados estruturais em tempo real.

Referências

ABNT. *NBR 9575: Impermeabilização – Seleção e projeto*. Rio de Janeiro, 2010. Citado na página 9.

ABNT. *NBR 15575: Edificações habitacionais - Desempenho*. Rio de Janeiro: Associação Brasileira de Normas Técnicas, 2021. Citado na página 9.

ABNT. *NBR 6118: Projeto de estruturas de concreto – Procedimento*. Rio de Janeiro, 2023. Citado na página 9.

AL-MAQTARI, O. et al. Efficientcracknet: A lightweight model for crack segmentation. *arXiv preprint arXiv:2409.18099*, 2024. Disponível em: <https://arxiv.org/html/2409.18099v1>. Citado 3 vezes nas páginas 3, 4 e 19.

AL-SMADI, M. et al. Crack detection and classification for reinforced concrete structures using deep learning. In: IEEE. *2022 IEEE 12th International Conference on System Engineering and Technology (ICSET)*. [S.l.], 2022. p. 124–129. Citado na página 1.

BA, J. L.; KIROS, J. R.; HINTON, G. E. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. Citado na página 19.

BAE, H.; AN, Y.-K. Computer vision-based statistical crack quantification for concrete structures. *Measurement*, v. 211, p. 112632, 2023. ISSN 0263-2241. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0263224123001963>. Citado na página 11.

BAHDANAU, D.; CHO, K.; BENGIO, Y. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014. Citado na página 20.

BENGIO, Y.; COURVILLE, A.; VINCENT, P. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 35, n. 8, p. 1798–1828, 2013. Citado na página 19.

CAO, J. Research on crack detection of bridge deck based on computer vision. *IOP Conference Series: Earth and Environmental Science*, IOP Publishing, v. 768, n. 1, p. 012161, may 2021. Disponível em: <https://dx.doi.org/10.1088/1755-1315/768/1/012161>. Citado na página 11.

CHEN, B. et al. Cracknet: A hybrid model for crack segmentation with dynamic loss function. *Sensors*, v. 24, 2024. Citado 2 vezes nas páginas 25 e 30.

CHEN, J.; LIU, G.; CHEN, L. Automatic crack classification and segmentation on concrete bridge images using convolutional neural networks and hybrid image processing. *Intelligent Transportation Infrastructure*, Oxford University Press, v. 1, 2022. Citado na página 1.

DAIS, D. et al. Automatic crack classification and segmentation on masonry surfaces using convolutional neural networks and transfer learning. *Automation in Construction*, Elsevier, v. 125, p. 103606, 2021. Citado na página 1.

DINIZ, J. d. C. N. et al. A method for detecting pathologies in concrete structures using deep neural networks. *Applied Sciences*, v. 13, n. 9, 2023. ISSN 2076-3417. Disponível em: <https://www.mdpi.com/2076-3417/13/9/5763>. Citado 4 vezes nas páginas 23, 28, 96 e 97.

DONG, X.; LIU, Y.; DAI, J. Recognition of concrete surface cracks based on improved transunet. *Buildings*, MDPI, v. 15, n. 4, p. 541, 2025. Citado na página 25.

DOSOVITSKIY, A. et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2021. Disponível em: <https://arxiv.org/abs/2010.11929>. Citado 9 vezes nas páginas 3, 15, 16, 18, 19, 20, 21, 25 e 47.

GEHRING, J. et al. Convolutional sequence to sequence learning. In: PMLR. *Proceedings of the 34th International Conference on Machine Learning*. [S.l.], 2017. p. 1243–1252. Citado na página 19.

GONZALEZ, R. C.; WOODS, R. E. *Digital Image Processing*. 3rd. ed. Reading, MA: Addison-Wesley, 2010. Citado na página 11.

HE, K. et al. Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2016. p. 770–778. Citado na página 14.

HOUDT, G. V.; MOSQUERA, C.; NÁPOLES, G. A review on the long short-term memory model. *Artificial Intelligence Review*, Springer, v. 53, n. 8, p. 5929–5955, 2020. Citado na página 19.

HOWARD, A. G. et al. *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. 2017. Disponível em: <https://arxiv.org/abs/1704.04861>. Citado 2 vezes nas páginas 14 e 15.

KARMOKAR, A. et al. Inspection of concrete structures by a computer vision technique and an unmanned aerial vehicle. In: *2020 International Conference on Computational Performance Evaluation (ComPE)*. [S.l.: s.n.], 2020. p. 338–343. Citado 6 vezes nas páginas 23, 2, 25, 28, 96 e 97.

KHATTAK, A. M. et al. Bridging convolutional neural networks and transformers for efficient crack detection in concrete building structures. *Sensors*, MDPI, v. 24, n. 15, p. 4802, 2024. Citado 2 vezes nas páginas 24 e 30.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, Association for Computing Machinery, New York, NY, USA, v. 60, n. 6, p. 84–90, maio 2012. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/3065386>. Citado 2 vezes nas páginas 14 e 25.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, v. 521, n. 7553, p. 436–444, 2015. Citado 3 vezes nas páginas 12, 13 e 14.

LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, 1998. Citado 2 vezes nas páginas 3 e 14.

LIU, Z. et al. *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows*. 2021. Disponível em: <https://arxiv.org/abs/2103.14030>. Citado na página 21.

LOVERDOS, D.; SARHOSIS, V. Automatic image-based brick segmentation and crack detection of masonry walls using machine learning. *Automation in Construction*, Elsevier, v. 140, p. 104389, 2022. Citado na página 1.

MIKOLOV, T. et al. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. Citado na página 19.

PAL, M. et al. An overview of challenges associated with automatic detection of concrete cracks in the presence of shadows. *Applied Sciences*, v. 11, n. 23, 2021. ISSN 2076-3417. Disponível em: <https://www.mdpi.com/2076-3417/11/23/11396>. Citado na página 2.

PAN, S. J.; YANG, Q. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, v. 22, n. 10, p. 1345–1359, 2009. Citado na página 15.

PAN, S. J.; YANG, Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, v. 22, n. 10, p. 1345–1359, 2010. Citado 2 vezes nas páginas 15 e 16.

RAMANATHAN, A. et al. Comparative analysis of deep learning models for crack detection in buildings. *Scientific Reports*, Nature Publishing Group, v. 15, n. 1, p. 1234, 2025. Citado na página 24.

SHAHIN, M. et al. Improving the concrete crack detection process via a hybrid visual transformer algorithm. *Sensors*, v. 24, n. 10, 2024. ISSN 1424-8220. Disponível em: <https://www.mdpi.com/1424-8220/24/10/3247>. Citado 5 vezes nas páginas 23, 24, 28, 96 e 97.

SHAHIN, M. et al. Concrete and pavement cracks detection leveraging vision transformer (vit) model. In: IEEE. *2025 IEEE 4th International Conference on Advanced Systems and Emergent Technologies (IC_ASET)*. [S.l.], 2025. p. 242–247. Citado na página 26.

SHARMA, S.; VARDHAN, M. Self-attention vision transformer with transfer learning for efficient crops and weeds classification. In: *2023 6th International Conference on Information Systems and Computer Networks (ISCON)*. [S.l.: s.n.], 2023. Citado na página 16.

SILVA, M. A. L. A. d.; LAURSEN, A. Patologias da construção civil: investigação patológica em edifício comercial de caruaru-pe. *Revista Científica Multidisciplinar Núcleo do Conhecimento*, v. 12, n. 11, p. 69–85, Nov. 2022. ISSN 2448-0959. Disponível em: <https://www.nucleodoconhecimento.com.br/engenharia-civil/investigacao-patologica>. Citado na página 2.

SILVA, M. A. R. da; PINHEIRO, E. C. N. M. Análise das patologias em uma residência, estudo de caso: Analysis of pathologies in a residence, case study. v. 8, p. 72114–72131, 2022. Citado na página 2.

SILVA, P. J. W. M. et al. Avaliação da presença de fissuras em imagens de estruturas de concreto através do uso de redes neurais profundas. *Matéria (Rio de Janeiro)*, SciELO Brasil, v. 29, n. 4, 2024. Disponível em: <https://www.scielo.br/j/rmat/a/NxHkMFgFY9WvjnCXCGgYrzj/>. Citado na página 1.

SONG, Q. et al. Pixel-level crack identification for bridge concrete structures using unmanned aerial vehicle photography and deep learning. *Structural Control and Health Monitoring*, Wiley, 2024. Citado 2 vezes nas páginas 24 e 26.

SZELISKI, R. *Computer Vision: Algorithms and Applications*. [S.l.]: Springer, 2010. Citado 2 vezes nas páginas 10 e 11.

TAN, M.; LE, Q. V. *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. 2020. Disponível em: <https://arxiv.org/abs/1905.11946>. Citado 2 vezes nas páginas 14 e 15.

TOUVRON, H. et al. Training data-efficient image transformers amp; distillation through attention. In: MEILA, M.; ZHANG, T. (Ed.). *Proceedings of the 38th International Conference on Machine Learning*. [S.l.]: PMLR, 2021. (Proceedings of Machine Learning Research, v. 139), p. 10347–10357. Citado 2 vezes nas páginas 20 e 21.

VASWANI, A. et al. Attention is all you need. *Advances in neural information processing systems*, v. 30, 2017. Citado 2 vezes nas páginas 19 e 20.

WANG, Y. et al. Source-free unsupervised domain adaptation: A survey. *Neural Networks*, Elsevier, v. 175, p. 106302, 2024. Citado na página 13.

WIGHTMAN, R. *PyTorch Image Models*. [S.l.]: GitHub, 2019. <https://github.com/rwightman/pytorch-image-models>. Citado na página 33.

XIAO, S. et al. Pavement crack detection with hybrid-window attentive vision transformers. *International Journal of Applied Earth Observation and Geoinformation*, 2023. Citado 3 vezes nas páginas 23, 28 e 30.

YE, G. et al. An improved transformer-based concrete crack classification method. *Scientific Reports*, Nature Publishing Group, v. 14, n. 1, p. 6226, 2024. Citado na página 1.

ZOU, Q. et al. Deepcrack: Learning hierarchical convolutional features for crack detection. *IEEE Transactions on Image Processing*, 2019. Citado 4 vezes nas páginas 23, 28, 96 e 97.

Özgenel, F. *Concrete Crack Images for Classification*. Mendeley Data, 2019. Disponível em: <https://doi.org/10.17632/5y9wdsg2zt.2>. Citado 7 vezes nas páginas 11, 13, 22, 23, 34, 35 e 42.