

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO, DE CIÊNCIAS EXATAS E EDUCAÇÃO
DEPARTAMENTO DE ENG. DE CONTROLE, AUTOMAÇÃO E COMPUTAÇÃO
CURSO DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Henrique Molder Gozdziejewski da Silva

Estudo Comparativo de Redes Neurais Convolucionais para Segmentação de Cáries em Radiografias Panorâmicas

Blumenau
2025

Henrique Molder Gozdziejewski da Silva

Estudo Comparativo de Redes Neurais Convolucionais para Segmentação de Cáries em Radiografias Panorâmicas

Trabalho de Conclusão de Curso de Graduação em Engenharia de Controle e Automação do Centro Tecnológico, de Ciências Exatas e Educação da Universidade Federal de Santa Catarina como requisito para a obtenção do título de Engenheiro de Controle e Automação.
Orientador: Prof. Dr. Mauri Ferrandin

Blumenau
2025

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

Silva, Henrique Molder Gozdziejewski da Silva
Estudo Comparativo de Redes Neurais Convolucionais para Segmentação de Cáries em Radiografias Panorâmicas / Henrique Molder Gozdziejewski da Silva Silva ; orientador, Mauri Ferrandin Ferrandin, 2025.
97 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Campus Blumenau, Graduação em Engenharia de Controle e Automação, Blumenau, 2025.

Inclui referências.

1. Engenharia de Controle e Automação. 2. Segmentação Semântica. 3. Detecção de Cáries. 4. Redes Neurais Convolucionais. 5. Análise Comparativa de Arquiteturas. I. Ferrandin, Mauri Ferrandin. II. Universidade Federal de Santa Catarina. Graduação em Engenharia de Controle e Automação. III. Título.

Henrique Molder Gozdziejewski da Silva

Estudo Comparativo de Redes Neurais Convolucionais para Segmentação de Cáries em Radiografias Panorâmicas

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de "Engenheiro de Controle e Automação" e aprovado em sua forma final pelo Curso de Graduação em Engenharia de Controle e Automação.

Blumenau, 05 de 12 de 2025.

Banca Examinadora:



Documento assinado digitalmente
Mauri Ferrandin
Data: 15/12/2025 16:56:13-0300
CPF: ***.580.869-**
Verifique as assinaturas em https://v.ufsc.br

Prof. Dr. Mauri Ferrandin
Orientador Universidade Federal de Santa Catarina



Documento assinado digitalmente
Carlos Roberto Moratelli
Data: 15/12/2025 17:23:14-0300
CPF: ***.999.369-**
Verifique as assinaturas em https://v.ufsc.br

Prof. Dr. Carlos Roberto Moratelli
Avaliador Universidade Federal de Santa Catarina



Documento assinado digitalmente
Marcos Vinicius Matsuo
Data: 15/12/2025 17:00:00-0300
CPF: ***.580.029-**
Verifique as assinaturas em https://v.ufsc.br

Prof. Dr. Marcos Vinícius Matsuo
Avaliador Universidade Federal de Santa Catarina

RESUMO

A detecção precoce de cáries dentárias em radiografias panorâmicas constitui desafio crítico para diagnóstico e planejamento terapêutico efetivos na odontologia. Métodos tradicionais de análise manual enfrentam limitações significativas, incluindo variabilidade inter-observador e tempo de análise elevado, comprometendo a acurácia diagnóstica e a eficiência operacional. Este trabalho tem como objetivo abordar estas limitações através de análise comparativa sistemática de arquiteturas de segmentação semântica baseadas em aprendizado profundo para detecção automatizada de lesões cariosas. Foram avaliadas cinco arquiteturas (*U-Net*, *U-Net++*, *DeepLabV3+*, *PSPNet* e *FPN*) integradas a três *backbones* extratores de características (*ResNet-34*, *EfficientNet-B0* e *EfficientNet-B3*), totalizando quinze configurações distintas. Os modelos foram rigorosamente treinados e avaliados utilizando *dataset* de 100 radiografias panorâmicas com anotações de lesões cariosas, empregando estratégias de balanceamento de classes. Os resultados evidenciaram que arquiteturas baseadas em pirâmides de características apresentam sinergia superior com *backbones* da família *EfficientNet* em cenários de dados limitados, superando significativamente arquiteturas clássicas baseadas em conexões *skip* densas. Esta trabalho conseguiu demonstrar o potencial da integração de redes neurais em diagnósticos odontológicos, oferecendo orientações práticas para pesquisadores e clínicos no auxílio de diagnóstico mais precisos, eficientes e escaláveis.

Palavras-chave: segmentação semântica; radiografia panorâmica; detecção de cáries; redes neurais convolucionais; aprendizado profundo; análise comparativa de arquiteturas

ABSTRACT

Early detection of dental caries in panoramic radiographs constitutes a critical challenge for effective diagnosis and treatment planning in dentistry. Traditional manual analysis methods face significant limitations, including inter-observer variability and high analysis time, compromising diagnostic accuracy and operational efficiency. This work aims to address these limitations through a systematic comparative analysis of semantic segmentation architectures based on deep learning for automated detection of carious lesions. Five architectures (*U-Net*, *U-Net++*, *DeepLabV3+*, *PSPNet*, and *FPN*) integrated with three feature extraction backbones (*ResNet-34*, *EfficientNet-B0*, and *EfficientNet-B3*) were evaluated, totaling fifteen distinct configurations. The models were rigorously trained and evaluated using a dataset of 100 panoramic radiographs with annotations of carious lesions, employing class balancing strategies. The results demonstrated that feature pyramid-based architectures exhibit superior synergy with EfficientNet family backbones in limited data scenarios, significantly outperforming classical architectures based on dense skip connections. This work successfully demonstrated the potential of integrating neural networks into dental diagnostics, providing practical guidance for researchers and clinicians in developing more accurate, efficient, and scalable diagnostic support systems.

Keywords: semantic segmentation; panoramic radiography; caries detection; convolutional neural networks; deep learning; architecture comparative analysis.

LISTA DE FIGURAS

Figura 1 – Arquitetura de uma Rede Neural Artificial 17
Figura 2 – Operações de convolução seguidas por *max-pooling* (acima) e *average-pooling* (abaixo) aplicadas a uma entrada de imagem 18
Figura 3 – Comparação entre paradigmas de análise de imagens 20
Figura 4 – Arquitetura Encoder-Decoder 22
Figura 5 – Arquitetura clássica da U-Net 22
Figura 6 – Arquitetura clássica da U-Net++ 23
Figura 7 – Convolução Atrosas 24
Figura 8 – Arquitetura DeepLabv3+ 25
Figura 9 – Problemas de análise de cenários 26
Figura 10 – Arquitetura PSPNet 26
Figura 11 – Função de perda auxiliar 27
Figura 12 – Métodos arquitetura piramidais 28
Figura 13 – Bloco de construção da FPN 29
Figura 14 – Arquitetura Completa do ResNet34 31
Figura 15 – Diagrama do Bloco Residual Básico 32
Figura 16 – Diagrama da EfficientNet-B0 34
Figura 17 – Exemplo 1092 do *Dataset* 48
Figura 18 – Exemplo 1092 do *Dataset* 49
Figura 19 – Proporção global entre pixels de tecido saudável e lesões cariosas . . . 51
Figura 20 – Pipeline de pré-processamento e *data augmentation* aplicado às radiografias 53
Figura 21 – Distribuição de parâmetros entre arquiteturas e backbones 56
Figura 22 – Curva de evolução do learning rate 61
Figura 23 – *Dice Score* médio por arquitetura de segmentação. 65
Figura 24 – IoU médio por arquitetura de segmentação. 65
Figura 25 – Precisão média por arquitetura de segmentação. 66
Figura 26 – *Recall* médio por arquitetura de segmentação. 66
Figura 27 – *Dice Score* médio por *backbone* de extração de características. 68
Figura 28 – IoU médio por *backbone* de extração de características. 68
Figura 29 – Precisão média por *backbone* de extração de características. 69
Figura 30 – *Recall* médio por *backbone* de extração de características. 69
Figura 31 – Radiografia panorâmica original, *ground truth* anotado manualmente (vermelho), e segmentação predita pelo modelo FPN + efficientnet-b3. 72
Figura 32 – Radiografia panorâmica original, *ground truth* anotado manualmente (vermelho), e segmentação predita pelo modelo FPN + resnet34. 72

Figura 33 – Radiografia panorâmica original, *ground truth* anotado manualmente (vermelho), e segmentação predita pelo modelo PSPNet + resnet34. . . 72
Figura 34 – Número de parâmetros (em milhões) por configuração de modelo. . . . 75
Figura 35 – Relação entre *Dice Score* e número de parâmetros 77
Figura 36 – Tempo de treinamento individual (em minutos) para cada configuração de modelo até ativação do *early stopping*. 78
Figura 37 – Relação entre *Dice Score* e tempo de treinamento individual por configuração 80
Figura 38 – Velocidade de inferência (FPS - *frames per second*) para cada configuração de modelo. 81
Figura 39 – Relação entre desempenho (*Dice Score*) e velocidade de inferência (FPS). Observa-se ausência de correlação linear, com configurações de alto desempenho distribuídas em ampla faixa de velocidades, permitindo seleção baseada em requisitos específicos de aplicação. 82

LISTA DE TABELAS

Tabela 1 – Comparação atualizada dos principais trabalhos de segmentação de cáries 41
Tabela 2 – *Trade-offs* entre acurácia e eficiência em arquiteturas de segmentação . 44
Tabela 3 – Distribuição de classes e desbalanceamento no conjunto de treinamento 51
Tabela 4 – Transformações de data augmentation implementadas com Albumentations 53
Tabela 5 – Variação de complexidade (parâmetros) por arquitetura 56
Tabela 6 – Estratégias de regularização e otimização 62
Tabela 7 – Hiperparâmetros principais de treinamento 62
Tabela 8 – Desempenho Médio por Arquitetura 64
Tabela 9 – Desempenho Médio por Backbone 68
Tabela 10 – Top 10 Configurações de Modelos 71
Tabela 11 – Comparativo de tempo e complexidade 75
Tabela 12 – Top 5 Configurações Mais Eficientes 84

SUMÁRIO

1 INTRODUÇÃO 12
1.1 JUSTIFICATIVA 13
1.2 OBJETIVOS 14
1.2.1 Objetivo Geral 14
1.2.2 Objetivos Específicos 14
1.3 ORGANIZAÇÃO DO TRABALHO 15
2 REFERÊNCIAL TEÓRICO 16
2.1 REDES NEURAIS ARTIFICIAIS 16
2.2 REDES NEURAIS CONVOLUCIONAIS 17
2.3 TIPOS DE ANÁLISE DE IMAGENS COM REDES NEURAIS 19
2.3.1 Classificação de Imagens 19
2.3.2 Detecção de Objetos 19
2.3.3 Segmentação Semântica 20
2.4 ARQUITETURAS PARA SEGMENTAÇÃO SEMÂNTICA 20
2.4.1 Estrutura Encoder-Decoder 20
2.4.2 Arquitetura U-Net 21
2.4.3 Arquitetura UNet++ 22
2.4.4 DeepLabV3+ 24
2.4.5 PSPNet 24
2.4.6 FPN 27
2.5 BACKBONES ENCODERS 28
2.5.1 Conceito de Backbone Encoder 28
2.5.2 ResNet34 30
2.5.3 EfficientNet-B0 e B3 33
2.6 MEDIDAS DE DESEMPENHO 35
2.6.1 Métricas de Sobreposição 35
2.6.2 Métrica de Precisão 36
2.7 TRABALHOS RELACIONADOS 37
2.7.1 Segmentação de Cáries em Odontologia 38
2.7.1.1 Arquiteturas Encoder-Decoder Clássicas 38
2.7.1.2 Arquiteturas com Mecanismos de Atenção 38
2.7.1.3 Abordagens Multi-estágio e Cascata 39
2.7.1.4 Segmentação Multi-classe de Cáries 39
2.7.1.5 Abordagens Recentes e Tendências 40
2.7.1.6 Desafios Persistentes 40
2.7.1.7 Análise Comparativa das Principais Contribuições 41
2.7.2 Estudos Comparativos de Arquiteturas 41

2.7.2.1 Estudos Comparativos em Imagens Médicas Gerais . . . 41
2.7.2.2 Comparações de Arquiteturas em Odontologia . . . 42
2.7.2.3 Análise Comparativa de Backbones . . . 43
2.7.2.4 Trade-offs entre Desempenho e Eficiência Computacional . . . 43
2.7.2.5 Variantes Avançadas e Arquiteturas Híbridas . . . 44
2.7.3 Aplicações com *Datasets* Limitados . . . 44
2.7.3.1 Técnicas de Data Augmentation . . . 45
2.7.3.2 Transfer Learning e Pré-treinamento . . . 45
2.7.3.3 Datasets Públicos . . . 46
2.7.3.4 Desafios e Limitações Persistentes . . . 47
3 DESENVOLVIMENTO . . . 48
3.1 CONJUNTO DE DADOS E PRÉ-PROCESSAMENTO . . . 48
3.1.1 Dataset Panoramic Dental . . . 48
3.1.2 Estratégia de Particionamento . . . 49
3.1.3 Análise da Distribuição de Classes . . . 50
3.1.3.1 Estratégia de Balanceamento . . . 52
3.1.4 Técnicas de Pré-processamento . . . 52
3.1.4.1 Redimensionamento e Normalização . . . 52
3.1.4.2 Data Augmentation . . . 53
3.2 CONFIGURAÇÕES E PARÂMETROS . . . 53
3.2.1 Configuração das Arquiteturas . . . 53
3.2.2 Configuração dos Backbones . . . 56
3.2.3 Hiperparâmetros de Treinamento . . . 59
3.2.4 Ambiente de Desenvolvimento e Hardware . . . 62
4 RESULTADOS . . . 64
4.1 ANÁLISE DE DESEMPENHO INDIVIDUAL . . . 64
4.1.1 Resultados por Arquitetura . . . 64
4.1.2 Resultados por Backbone . . . 67
4.1.3 Análise Arquitetura-Backbone . . . 70
4.2 ANÁLISE DE EFICIÊNCIA COMPUTACIONAL . . . 74
4.2.1 Complexidade dos Modelos . . . 74
4.2.1.1 Número de Parâmetros Treináveis . . . 75
4.2.1.2 Relação entre Complexidade e Desempenho . . . 76
4.2.2 Tempo de Treinamento e Inferência . . . 77
4.2.2.1 Tempo de Treinamento Individual . . . 78
4.2.2.2 Velocidade de Inferência . . . 80
4.2.2.3 Trade-off entre Desempenho e Velocidade . . . 82
4.2.3 Relação Custo-Benefício . . . 84
4.2.3.1 Métrica de Eficiência Estrutural . . . 84

4.2.3.2 Análise das Configurações Mais Eficientes 84
4.2.3.3 Análise de Custo-Benefício 85
5 CONCLUSÃO 87
REFERÊNCIAS 89

1 INTRODUÇÃO

Nos últimos anos, o campo da odontologia testemunhou avanços significativos nas tecnologias de imagem, particularmente em radiografia e tomografia. Essas modalidades de imagem tornaram-se ferramentas essenciais para o diagnóstico de várias condições dentárias, incluindo cáries, fraturas e determinação da necessidade de extração dentária. No entanto, a análise dessas imagens tradicionalmente depende da interpretação manual por profissionais odontológicos, um processo que demanda tempo, é intensivo em mão de obra e suscetível a erro humano (LANGLAND; LANGLAIS; PREECE, 2002; WHITE; PHAROAH, 2013). A indústria odontológica, embora beneficiando-se desses avanços tecnológicos, ainda enfrenta limitações significativas para processar e analisar eficientemente a vasta quantidade de dados de imagem gerados. Muitas clínicas odontológicas e centros de imagem continuam dependendo exclusivamente de análises manuais, levando a potenciais gargalos no atendimento ao paciente, aumento de custos operacionais, e risco de condições negligenciadas ou inconsistentemente diagnosticadas (GEETHA; APRAMEYA; HINDUJA, 2020).

Em resposta a esses desafios multifacetados, a aplicação da inteligência artificial, particularmente técnicas de aprendizado profundo especializadas em segmentação semântica, surgiu como uma solução tecnológica promissora. Diferentemente de abordagens de classificação tradicionais que apenas categorizam imagens inteiras, a segmentação semântica oferece a capacidade de identificar e delimitar precisamente cada pixel de imagens, fornecendo informações espaciais detalhadas (WITHEY; KOLES, 2008; LAI, 2015).

As Redes Neurais Convolucionais (CNNs) especializadas em segmentação, como as arquiteturas da família U-Net, demonstraram notável sucesso na análise de imagens médicas, incluindo radiografias dentárias (KRIZHEVSKY; SUTSKEVER; HINTON, G. E., 2012; LECUN; BENGIO; HINTON, G., 2015). Esses sistemas possuem o potencial de automatizar o processo de análise, potencialmente reduzindo o tempo de diagnóstico, minimizando erros humanos e oferecendo relatórios mais rápidos e precisos (SCHWENDICKE *et al.*, 2021).

Trabalhos anteriores na área focaram predominantemente em classificação binária de imagens inteiras, onde sistemas determinavam apenas a presença ou ausência de cáries em uma radiografia (LEE, J.-H. *et al.*, 2018; IMAK, Andac *et al.*, 2022). Embora esses estudos tenham demonstrado viabilidade técnica, uma lacuna significativa permanece na literatura para sistemas que podem delimitar precisamente a localização e extensão de lesões cariosas individualmente identificadas. Esta capacidade de localização precisa é fundamental para aplicações clínicas reais, onde dentistas necessitam informações espaciais detalhadas para planejamento terapêutico.

Mais especificamente, a literatura atual apresenta três limitações críticas que motivam este estudo comparativo: primeira, a ausência de estudos sistemáticos que comparem

diferentes arquiteturas de segmentação em cenários de dados limitados, típicos da realidade clínica odontológica onde grandes datasets anotados são escassos e custosos de se obter; segunda, a falta de análise comparativa entre diferentes backbones encoders quando aplicados especificamente à segmentação de cáries dentárias, sendo que a escolha do backbone pode impactar significativamente tanto a acurácia quanto a eficiência computacional; terceira, a ausência de avaliação sistemática da relação custo-benefício entre complexidade arquitetural e desempenho de segmentação, informação crucial para implementação em ambientes clínicos com recursos computacionais limitados.

Para abordar essas lacunas fundamentais na literatura, este trabalho propõe três contribuições principais que diferenciam significativamente a abordagem dos métodos existentes: primeira, a realização de análise comparativa sistemática entre cinco arquiteturas estado-da-arte de segmentação semântica (UNet, UNet++, DeepLabV3+, PSPNet e FPN) aplicadas especificamente à detecção de cáries dentárias em radiografias panorâmicas; segunda, a avaliação comparativa de três backbones encoders distintos (ResNet34, EfficientNet-B3 e EfficientNet-B3) em cada arquitetura de segmentação, analisando como diferentes estratégias de extração de características afetam o desempenho em cenários de dados limitados; terceira, a análise quantitativa da relação entre complexidade computacional e acurácia de segmentação através de métricas estabelecidas (Dice Score, IoU, Precision e Recall), fornecendo diretrizes práticas para seleção arquitetural em aplicações clínicas reais. O estudo utilizará um dataset de 100 radiografias panorâmicas cuidadosamente anotadas para lesões cariosas, representando um cenário realístico de dados limitados comum na prática odontológica. Através de treinamento e avaliação padronizados, será possível identificar qual combinação arquitetura-backbone demonstra maior robustez e potencial de generalização para segmentação de cáries dentárias.

A abordagem comparativa proposta visa fornecer evidências empíricas sólidas sobre qual configuração arquitetural oferece o melhor equilíbrio entre acurácia de segmentação e viabilidade computacional. Ao estabelecer benchmarks comparativos rigorosos entre diferentes abordagens, este trabalho contribui para o avanço do conhecimento sobre aplicação de técnicas de segmentação semântica em odontologia computacional, fornecendo diretrizes baseadas em evidências para seleção de arquiteturas em aplicações clínicas reais. Os resultados desta análise comparativa poderão orientar futuras implementações de sistemas de auxílio ao diagnóstico odontológico, maximizando tanto a precisão diagnóstica quanto a eficiência operacional em ambientes clínicos com recursos computacionais limitados.

1.1 JUSTIFICATIVA

A cárie dentária é uma das condições de saúde bucal mais prevalentes globalmente, afetando aproximadamente 2,3 bilhões de pessoas (ORGANIZATION *et al.*, 2022). No contexto brasileiro, estudos epidemiológicos demonstram que cerca de 56% da população adulta apresenta pelo menos uma lesão cariosa não tratada, evidenciando a magnitude do

problema de saúde pública (COSTA; VASCONCELOS; ABREU, 2013).

A detecção precoce e precisa de lesões cariosas é fundamental para o sucesso do tratamento odontológico, permitindo intervenções minimamente invasivas que preservam a estrutura dentária e reduzem custos de tratamento. No entanto, a interpretação manual de radiografias panorâmicas apresenta limitações significativas: estudos indicam variabilidade inter-observador de até 30% entre profissionais na detecção de cáries incipientes (LIM *et al.*, 2012), enquanto o tempo médio para análise completa de uma radiografia panorâmica varia entre 5 a 15 minutos por profissional (BEAINI, 2010).

Do ponto de vista tecnológico, a convergência de fatores favoráveis torna este momento propício para o desenvolvimento de soluções automatizadas: (i) disponibilidade crescente de datasets médicos anotados; (ii) avanços significativos em arquiteturas de segmentação semântica; (iii) capacidade computacional acessível para implementação clínica; e (iv) demanda crescente por ferramentas de diagnóstico auxiliar na odontologia digital.

A justificativa econômica é igualmente relevante: estudos de análise de custo-efetividade demonstram que ferramentas de diagnóstico automatizado podem reduzir em até 40% o tempo de análise radiográfica, resultando em economia estimada de R$ 2.000 a R$ 5.000 por consultório odontológico mensalmente (RODRIGUES *et al.*, 2024). Adicionalmente, a detecção precoce automatizada pode prevenir tratamentos complexos, com economia potencial de R$ 800 a R$ 2.500 por paciente em procedimentos endodônticos e protéticos evitados.

Por fim, a ausência de ferramentas comerciais específicas para segmentação de cáries em radiografias panorâmicas representa uma lacuna de mercado significativa. Embora existam sistemas de classificação binária, nenhuma solução disponível oferece segmentação semântica precisa com informação espacial detalhada, características essenciais para aplicação clínica efetiva e planejamento terapêutico individualizado.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Realizar comparativo do desempenho de diferentes arquiteturas de redes neurais convolucionais na segmentação de cáries dentárias utilizando um dataset de 100 radiografias panorâmicas, avaliando múltiplos backbones.

1.2.2 Objetivos Específicos

- Implementar e treinar cinco arquiteturas distintas de redes neurais convolucionais (UNet, UNet++, DeepLabV3+, PSPNet e FPN) para segmentação de cáries dentárias, utilizando 3 backbones encoders diferentes (ResNet34, EfficientNet-B0 e EfficientNet-B3).

- Realizar análise comparativa quantitativa do desempenho das configurações através de métricas de segmentação (Dice Score, IoU, Precision e Recall) e analisar a relação custo-benefício entre acurácia e complexidade computacional.
- Analisar como diferentes backbones se comportam especificamente em cenários de dados limitados para segmentação de cáries dentárias, identificando qual abordagem arquitetural demonstra maior robustez e potencial de generalização.

1.3 ORGANIZAÇÃO DO TRABALHO

Este trabalho está organizado em cinco capítulos. O Capítulo 2 apresenta o referencial teórico necessário para compreensão do estudo, abordando os fundamentos de redes neurais artificiais e convolucionais, as arquiteturas de segmentação semântica avaliadas (U-Net, U-Net++, DeepLabV3+, PSPNet e FPN), os backbones encoders utilizados (ResNet34, EfficientNet-B0 e EfficientNet-B3), as métricas de desempenho empregadas na avaliação dos modelos, e uma revisão dos principais trabalhos correlatos em segmentação de cáries dentárias e estudos comparativos de arquiteturas. O Capítulo 3 descreve detalhadamente a metodologia empregada, incluindo a caracterização do dataset, as estratégias de pré-processamento e balanceamento de classes implementadas para lidar com o desbalanceamento extremo característico do problema, as configurações específicas de cada arquitetura e backbone, e os hiperparâmetros de treinamento padronizados para garantir comparabilidade justa entre as quinze configurações testadas. O Capítulo 4 apresenta os resultados obtidos, começando com a avaliação de desempenho individual por arquitetura e por backbone, seguida pela análise integrada das combinações arquitetura-backbone que revela padrões de sinergia, e finalizando com análise abrangente de eficiência computacional. Finalizando com o Capítulo 5, que consolida as conclusões do estudo, discutindo os objetivos propostos, as implicações práticas dos resultados para implementação clínica, as limitações identificadas e as direções para trabalhos futuros que possam expandir e aprimorar esta linha de pesquisa.

2 REFERÊNCIAL TEÓRICO

Este capítulo apresenta os fundamentos teóricos que foram necessários para o desenvolvimento do trabalho. São abordados os conceitos de redes neurais artificiais e convolucionais, as diferentes abordagens para análise de imagens (classificação, detecção e segmentação), e as arquiteturas especializadas para segmentação semântica. Também são discutidas as técnicas de *transfer learning*, métricas de avaliação específicas para segmentação, e funções de perda adaptadas para problemas com classes desbalanceadas. Por fim, são apresentados os trabalhos relacionados na área de estudo, estabelecendo o contexto científico e identificando as lacunas que motivaram o desenvolvimento desta pesquisa.

2.1 REDES NEURAIS ARTIFICIAIS

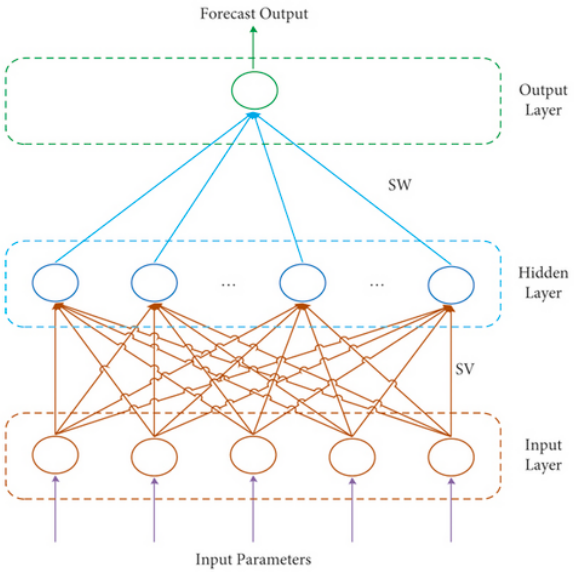
As Redes Neurais Artificiais (RNAs), também conhecidas como *Artificial Neural Networks* (ANNs), podem ser definidas como modelos computacionais que se inspiram na forma como os sistemas nervosos biológicos funcionam. Uma característica fundamental das Redes Neurais é que seus neurônios são organizados em camadas, sendo que a forma de conexão entre essas camadas varia conforme a arquitetura. Nas Redes Neurais Convolucionais, cada neurônio se conecta a uma pequena região espacial da camada anterior, característica que permite a extração de *features* locais. Por outro lado, nas camadas completamente conectadas (*fully connected*), cada neurônio estabelece conexões com todos os neurônios da camada precedente (O'SHEA; NASH, 2015).

As Redes Neurais Artificiais constituem uma metodologia proeminente no campo da Inteligência Artificial. Inspiradas pela arquitetura e funcionamento dos sistemas nervosos biológicos, como o cérebro humano, as RNAs são sistemas computacionais projetados para processar informações e aprender a partir de dados (YANG, 2008).

O aprendizado de máquina (*machine learning*), do qual o aprendizado profundo (*deep learning*) é uma implementação, confere às RNAs a capacidade de aprender e aprimorar seu desempenho com base na experiência, sem a necessidade de programação explícita para cada tarefa (PURWONO *et al.*, 2022).

Uma arquitetura básica de uma RNA, representada pela Figura 1, é estruturada em camadas, sendo a primeira delas a camada de entrada, onde os dados iniciais são apresentados à rede. Em seguida, tem-se uma ou mais camadas ocultas, que representam a maior parte do processamento, e por fim uma camada de saída, responsável por produzir o resultado final da rede desenvolvida. Cada camada é composta por unidades de processamento chamadas nós ou neurônios artificiais, que são interconectados por conectores sinápticos, cada um associado a um peso sináptico (*synaptic weight*) (YANG, 2008).

Figura 1 – Arquitetura de uma Rede Neural Artificial



Fonte: (MADHIARASAN; LOUZAZNI, 2022)

2.2 REDES NEURAIS CONVOLUCIONAIS

As Redes Neurais Convolucionais (CNNs) constituem uma forma de arquitetura de Rede Neural, sendo principalmente utilizadas para solucionar tarefas de reconhecimento de padrões baseadas em imagens. Diferentemente de outras arquiteturas neurais, como Rede Neural Recorrente (RNN) ou Perceptron Multicamada (MLP), as CNNs partem do pressuposto de que suas entradas serão imagens, o que foca sua arquitetura para lidar melhor com esse tipo específico de dado.

O que caracteriza fundamentalmente uma CNN é a presença de camadas convolucionais e de *pooling* como camadas iniciais, além de outras operações comuns em redes neurais, como o nivelamento (*flatten*) e regularização.

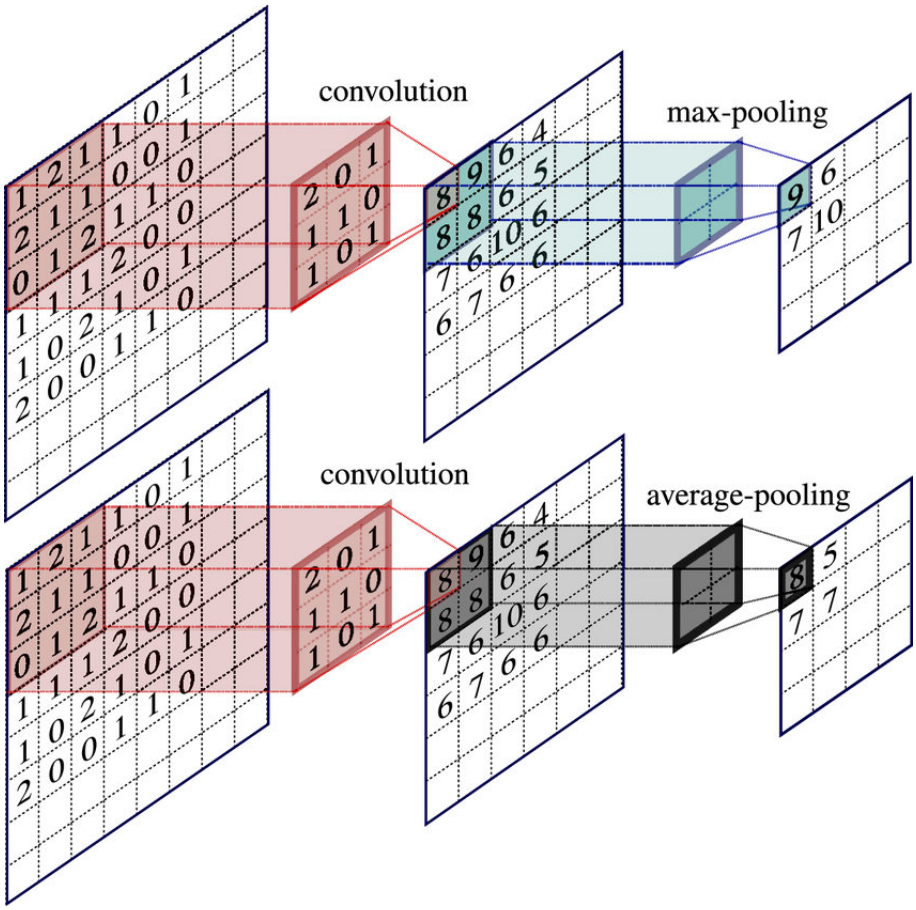
Analisando a arquitetura básica de uma CNN, a camada de entrada geralmente contém os valores de pixel de uma imagem e tipicamente recebe um tensor de ordem 3, estrutura multidimensional que representa matematicamente a imagem. No caso de imagens coloridas, esse tensor possui dimensões correspondentes à altura, largura e canais de cor (R, G, B). A camada convolucional visa aplicar operações de convolução à entrada, utilizando kernels aprendíveis, também chamados de filtros. Um filtro convolucional é representado por uma matriz de pesos de tamanho fixo que desliza sobre a imagem de entrada, realizando multiplicações elemento a elemento e somando os resultados para gerar um mapa de características (representação que destaca padrões específicos detectados na imagem, como bordas, texturas ou formas). A Figura 2 ilustra esse processo de convolução, onde um filtro desliza sobre a entrada gerando mapas de características. Os parâmetros a serem ajustados durante o treinamento nesta camada são os pesos dos filtros e um valor

de viés por filtro (WU, 2017).

Uma característica importante dos filtros convolucionais é que eles são invariantes à translação da imagem, ou seja, conseguem detectar o mesmo padrão independentemente de sua posição na imagem, baseando-se em princípios de localidade ao analisar uma região por vez para entender o todo. Os neurônios em uma camada convolucional são organizados em três dimensões: altura, largura e profundidade (que se refere à dimensão do volume de ativação), e conectam-se apenas a uma pequena região da camada anterior (WU, 2017).

Já a camada de *pooling* (subamostragem) tem a função de reduzir progressivamente a dimensão espacial dos dados, sintetizando um conjunto de valores vizinhos em um único valor representativo, geralmente através de operações como máximo ou média, conforme demonstrado na Figura 2 com *max-pooling* e *average-pooling*. O *pooling* visa reduzir o tamanho da representação mantendo as características mais importantes, o que contribui para a capacidade de generalização do modelo e ajuda a reduzir o custo computacional e o número de parâmetros (WU, 2017).

Figura 2 – Operações de convolução seguidas por *max-pooling* (acima) e *average-pooling* (abaixo) aplicadas a uma entrada de imagem



Fonte: Adaptado de (**FONTE_ORIGINAL**)

Ainda existem as Funções de Ativação, como a ReLU (*Rectified Linear Unit*), que são frequentemente aplicadas após as operações convolucionais, introduzindo não

linearidade ao modelo e permitindo que a rede aprenda relações complexas nos dados. A operação de nivelamento (*flatten*) converte os mapas de características multidimensionais em um vetor unidimensional, preparando os dados para as camadas totalmente conectadas. Por fim, essas camadas densas realizam a classificação final a partir das características extraídas, calculando pontuações para cada classe possível, sendo que o número de pesos nessas camadas depende do número de neurônios nas camadas adjacentes (WU, 2017).

Uma arquitetura CNN convencional geralmente consiste em uma camada de entrada seguida por uma ou mais sequências de uma camada convolucional e uma camada de *pooling*, terminando com um conjunto de camadas totalmente conectadas. Os únicos parâmetros aprendíveis em uma CNN estão tipicamente nas camadas convolucionais e densas (WU, 2017).

A eficiência de uma CNN se deve, em grande parte, ao menor número de parâmetros a serem ajustados durante o treinamento, uma redução que ocorre pelas características das camadas convolucionais e de *pooling*. Isso também torna a rede menos complexa e menos propensa ao *overfitting* (sobre-ajuste), fenômeno onde o modelo memoriza os dados de treinamento mas falha em generalizar para novos exemplos (WU, 2017).

As CNNs podem ser escaladas facilmente para entradas de imagem com muitas características e podem ser paralelizadas com maior facilidade. Além disso, ao processar imagens, as CNNs são capazes de preservar informações sobre a organização espacial dos pixels, extraindo características hierárquicas a partir das operações de convolução e *pooling* (WU, 2017).

2.3 TIPOS DE ANÁLISE DE IMAGENS COM REDES NEURAIS

2.3.1 Classificação de Imagens

A classificação de imagens constitui a tarefa mais fundamental em visão computacional, onde o objetivo é atribuir uma única etiqueta (*label*) a uma imagem completa. Neste paradigma, a rede neural recebe como entrada uma imagem e produz como saída uma distribuição de probabilidades sobre um conjunto pré-definido de classes (KRIZHEVSKY; SUTSKEVER; HINTON, G. E., 2012).

2.3.2 Detecção de Objetos

A detecção de objetos avança além da classificação ao não apenas identificar quais objetos estão presentes na imagem, mas também localizar cada instância através de caixas delimitadoras (*bounding boxes*). Cada caixa delimitadora é definida por coordenadas que especificam sua posição (x, y) e dimensões (largura e altura), acompanhadas de uma etiqueta de classe e um score de confiança (REDMON *et al.*, 2016).

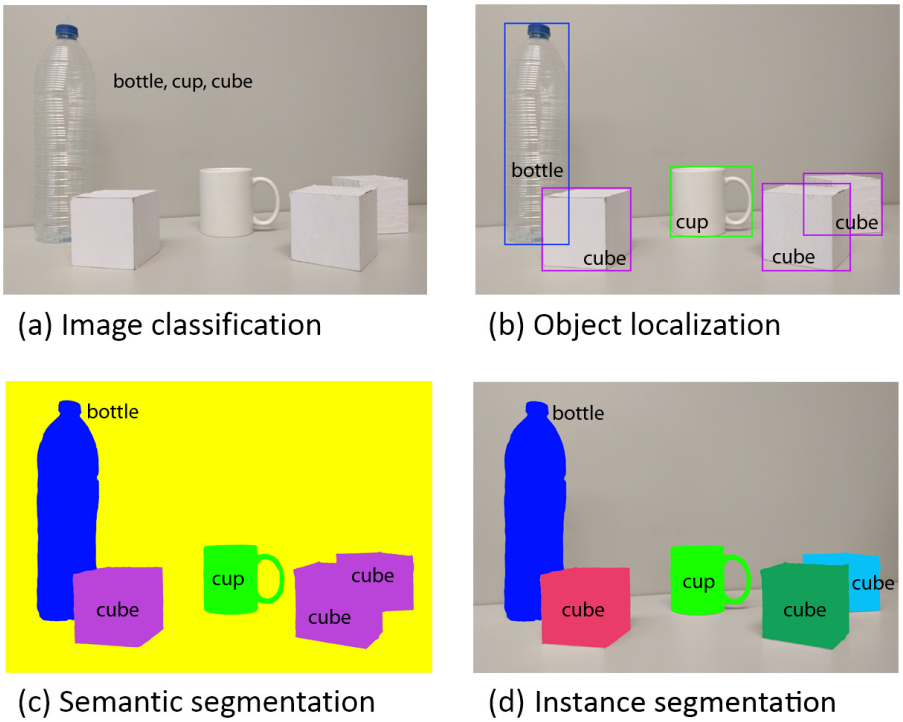
2.3.3 Segmentação Semântica

A segmentação semântica representa o nível mais granular de análise de imagens, realizando classificação pixel a pixel. Neste paradigma, cada pixel da imagem recebe uma etiqueta de classe, resultando em um mapa de segmentação que delineia precisamente os contornos dos objetos de interesse (LONG; SHELHAMER; DARRELL, 2015).

Existem duas variantes principais de segmentação, a primeira é a Segmentação Semântica, que atribui cada pixel a uma classe, mas não distingue entre instâncias diferentes da mesma classe. A segunda é Segmentação de Instância, que além de classificar cada pixel, também diferencia entre instâncias distintas da mesma classe (LONG; SHELHAMER; DARRELL, 2015).

A Figura 3 ilustra visualmente as diferenças entre os diferentes tipos de análise de imagens.

Figura 3 – Comparação entre paradigmas de análise de imagens



Fonte: (GARCIA-GARCIA *et al.*, 2017)

2.4 ARQUITETURAS PARA SEGMENTAÇÃO SEMÂNTICA

2.4.1 Estrutura Encoder-Decoder

A arquitetura *encoder-decoder* constitui um padrão arquitetural amplamente adotado em tarefas de segmentação semântica, caracterizado por dois componentes principais que operam em sequência: o *encoder* (codificador) e o *decoder* (decodificador) (BADRINARAYANAN; KENDALL; CIPOLLA, 2017).

O encoder é responsável pela extração progressiva de características da imagem de entrada através de camadas convolucionais sucessivas, acompanhadas de operações de *pooling* que reduzem gradualmente a resolução espacial enquanto aumentam a profundidade dos canais de características. Este processo de *downsampling* permite que a rede capture informações semânticas de alto nível e contexto global da imagem, mas resulta na perda de detalhes espaciais finos. Matematicamente, o *encoder* realiza uma transformação $E : \mathbb{R}^{H \times W \times C} \rightarrow \mathbb{R}^{H/2^n \times W/2^n \times C'}$, onde H e W representam as dimensões espaciais originais, C o número de canais de entrada, n o número de estágios de *downsampling*, e C' a profundidade dos canais de características no espaço latente (RONNEBERGER; FISCHER; BROX, 2015).

O *decoder* executa o processo inverso, realizando *upsampling* progressivo das características de baixa resolução extraídas pelo *encoder* para reconstruir um mapa de segmentação na resolução espacial original da imagem. Durante este processo de reconstrução, o *decoder* combina características semânticas de alto nível com informações espaciais de baixo nível, tipicamente através de conexões skip que transmitem características diretamente do encoder para o decoder (LONG; SHELHAMER; DARRELL, 2015). Formalmente, o decoder realiza a transformação $D : \mathbb{R}^{H/2^n \times W/2^n \times C'} \rightarrow \mathbb{R}^{H \times W \times K}$, onde K representa o número de classes de segmentação.

As conexões skip ou skip connections constituem um mecanismo arquitetural fundamental que conecta camadas do *encoder* diretamente às camadas correspondentes do *decoder*, permitindo a transmissão de informações de alta resolução através da rede. Estas conexões mitigam o problema da perda de detalhes espaciais causada pelas operações de *downsampling* no *encoder*, permitindo que o decoder acesse tanto características semânticas de alto nível quanto informações espaciais de alta resolução necessárias para produzir segmentações precisas dos limites dos objetos (RONNEBERGER; FISCHER; BROX, 2015).

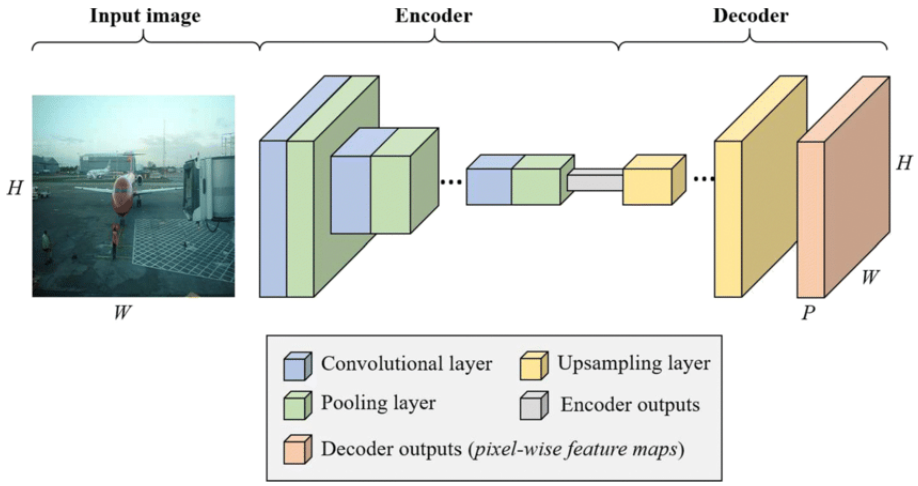
A Figura 4 ilustra esquematicamente a arquitetura encoder-decoder, evidenciando o fluxo de informação através da rede e a fusão de características em diferentes escalas espaciais.

2.4.2 Arquitetura U-Net

A inovação fundamental da U-Net reside no uso estratégico de conexões *skip*. As conexões *skip* concatenam os mapas de características de alta resolução e baixo nível do *encoder* com os mapas de características de baixo resolução e alto nível do *decoder* (ZHOU, Z. *et al.*, 2018; ZHANG, H. *et al.*, 2023). Essa fusão é crucial porque resolve o problema da perda de detalhes de alta resolução, já que enquanto o *encoder* reduz a resolução para capturar o contexto, as informações finas sobre os limites dos objetos são preservadas nos estágios iniciais. É possível observar o formato clássico de sua arquitetura na Figura 5.

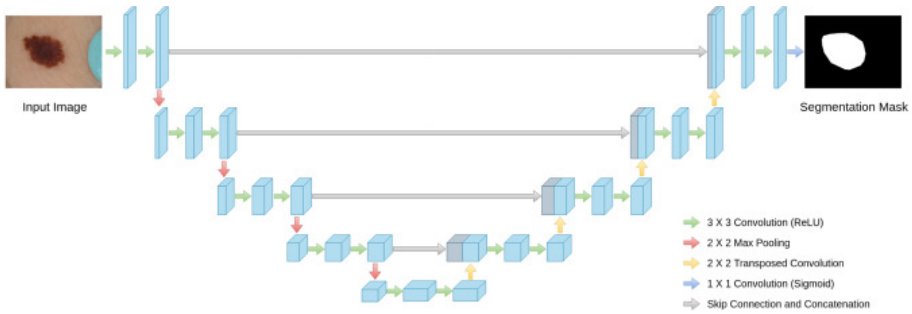
A U-Net tradicional possui uma limitação que prepara o terreno para a evolução de

Figura 4 – Arquitetura Encoder-Decoder



Fonte: (TONG; XU; DENOEUX, 2021)

Figura 5 – Arquitetura clássica da U-Net



Fonte: (IBTEHAZ; RAHMAN, 2020)

arquiteturas como a UNet++. As conexões *skip* simples na U-Net diretamente encaminham mapas de características de alta resolução do *encoder* para a rede do *decoder*, resultando na fusão de mapas de características semanticamente dissimilares. Essa lacuna semântica entre as características do *encoder* e do *decoder* pode tornar a tarefa de aprendizado mais desafiadora para o otimizador. Uma ideia é que o otimizador lidaria com uma tarefa de aprendizado mais fácil se os mapas de características do *decoder* e do *encoder* fossem semanticamente semelhantes (ZHOU, Z. *et al.*, 2018).

2.4.3 Arquitetura UNet++

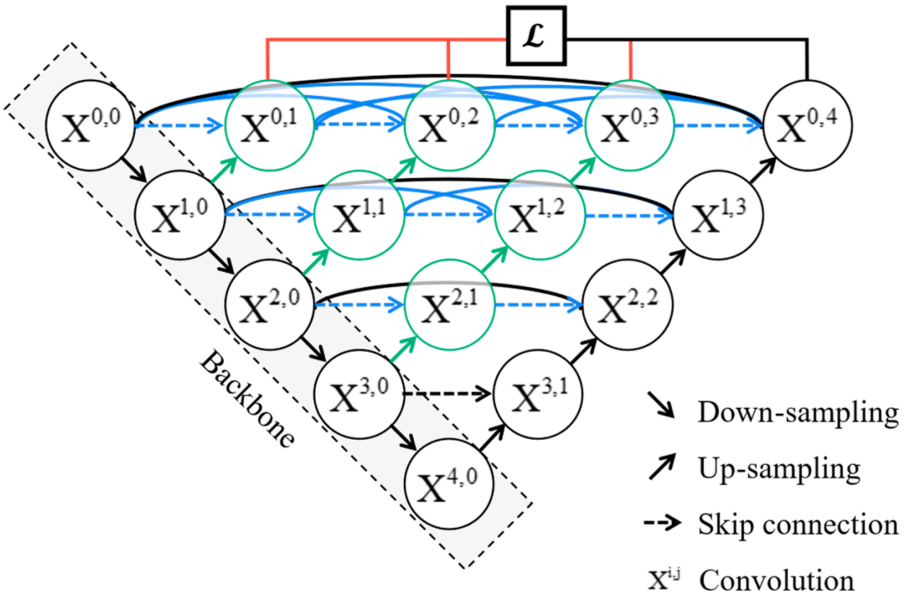
A UNet++ foi concebida para abordar e superar as limitações específicas da U-Net tradicional, que se tornam, em alguns casos, particularmente problemáticas na segmentação de estruturas pequenas e delicadas. A abordagem de empregar conexões *skip* simples, levando à fusão de mapas de características semanticamente dissimilares, pode degradar o desempenho da segmentação e tornar a tarefa de aprendizado mais desafiadora para o otimizador (ZHOU, Z. *et al.*, 2018; ZHANG, H. *et al.*, 2023).

A proposta da UNet++ visa mitigar esta limitação fundamental, sendo caracte-

rizada como uma nova arquitetura mais poderosa para segmentação de imagens, já que o conceito central dela reside em sua estrutura de rede *encoder-decoder* profundamente supervisionada onde as sub-redes do *encoder* e do *decoder* são conectadas através de uma série de caminhos de *skip* aninhados e densos (ZHOU, Z. *et al.*, 2018). A hipótese da UNet++ é que o otimizador enfrentaria uma tarefa de aprendizado mais fácil quando os mapas de características da rede do *decoder* e do *encoder* fossem semanticamente semelhantes (ZHANG, H. *et al.*, 2023).

A Figura 6 ilustra a arquitetura característica da UNet++, evidenciando sua estrutura aninhada e os múltiplos caminhos de conexão entre *encoder* e *decoder*. Na UNet++, os mapas de características de alta resolução do *encoder* passam por um bloco de convolução denso cujo número de camadas de convolução depende do nível da pirâmide. Esses blocos de convolução densos trazem o nível semântico dos mapas de características do *encoder* para mais perto dos mapas de características que aguardam no *decoder* (ZHOU, Z. *et al.*, 2018). Formalmente, cada nó ao longo do caminho de *skip* (Xi,j) recebe múltiplas entradas, incluindo a saída da camada anterior no mesmo bloco denso e a saída *upsampled* do caminho de *skip* inferior, permitindo que todas as características anteriores se acumulem e cheguem ao nó atual. Essa estrutura aninhada e densa, visível na arquitetura apresentada, permite que a UNet++ capture mais informações contextuais e aprimore a integração de características em diferentes escalas, gerando mapas de características de resolução total em múltiplos níveis semânticos (ZHANG, H. *et al.*, 2023).

Figura 6 – Arquitetura clássica da U-Net++



Fonte: (TULSANI; KUMAR, P.; PATHAN, 2021)

A justificativa teórica da escolha arquitetural da UNet++ para a segmentação de imagens baseia-se na sua capacidade demonstrada de superar a U-Net original na fusão de características semanticamente distintas e na preservação de informações espaciais de

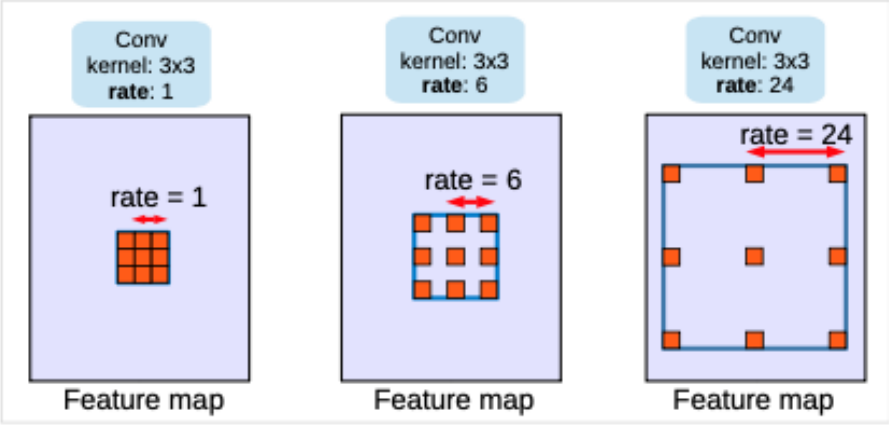
alta resolução (ZHOU, Z. *et al.*, 2018).

2.4.4 DeepLabV3+

O DeepLabV3+ representa uma uma extensão do modelo DeepLabV3 e introduz uma estrutura encoder-decoder que combina as vantagens do pooling de pirâmide espacial com a capacidade de recuperar detalhes precisos dos limites dos objetos, característica fundamental para a segmentação acurada de lesões apicais em radiografias panorâmicas (CHEN, L.-C. *et al.*, 2018).

A arquitetura do DeepLabV3+ apresentada na Figura 8 mantém o DeepLabV3 como módulo codificador, responsável por extrair características ricas em informações semânticas através do módulo Atrous Spatial Pyramid Pooling (ASPP), enquanto adiciona um decodificador simples mas eficaz para refinar os resultados da segmentação. O módulo ASPP aplica convoluções atrosas paralelas com diferentes taxas de dilatação, permitindo a captura de informações contextuais em múltiplas escalas sem perda de resolução espacial, assim como é mostrado na Figura 7. As convoluções atrosas com diferentes taxas de dilatação expandem o campo receptivo do modelo, possibilitando a codificação de objetos em múltiplas escalas de forma eficiente, característica particularmente vantajosa para a detecção de lesões apicais que podem variar significativamente em tamanho e localização (CHEN, L.-C. *et al.*, 2018).

Figura 7 – Convolução Atrosas

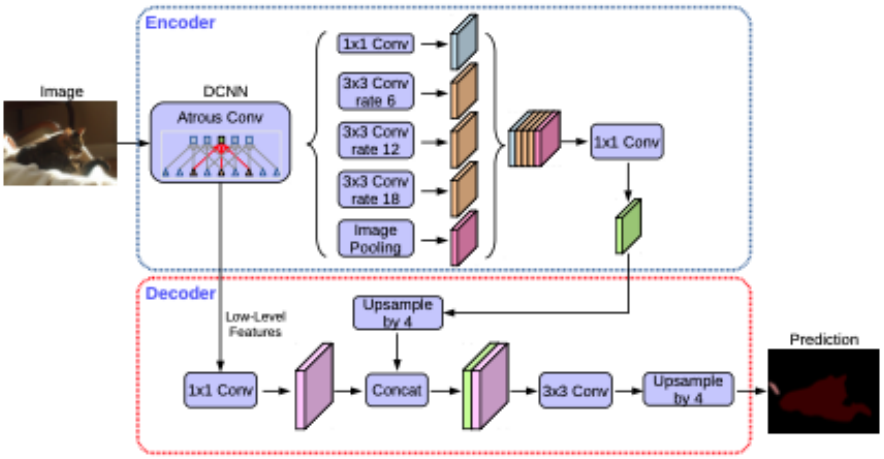


Fonte: (CHEN, L.-C. *et al.*, 2017)

2.4.5 PSPNet

A Pyramid Scene Parsing Network (PSPNet) foi desenvolvida especificamente para superar desafios relacionados ao entendimento de cenas complexas e vocabulário aberto, problemas comuns em aplicações de análise de imagens médicas e outras áreas que demandam alta precisão na segmentação pixel a pixel. O desenvolvimento da PSPNet foi motivado pela identificação de três categorias principais de erros em métodos de

Figura 8 – Arquitetura DeepLabv3+



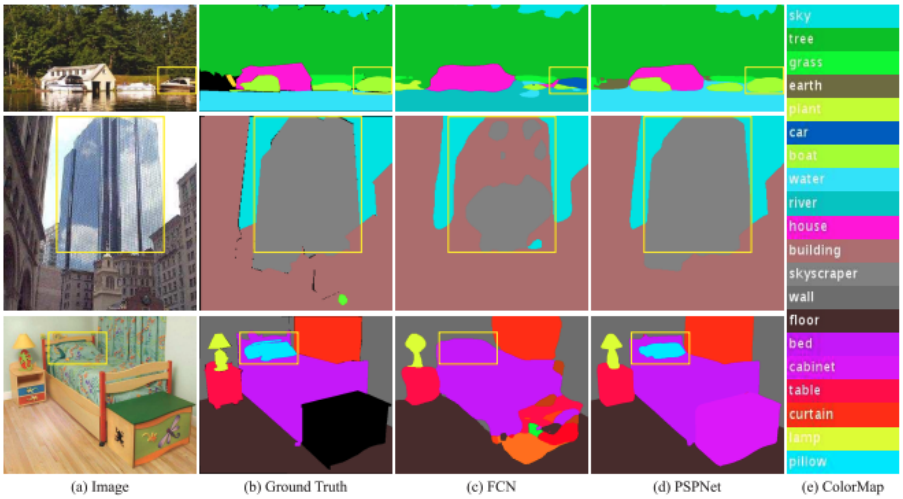
Fonte: (CHEN, L.-C. *et al.*, 2018)

segmentação semântica baseados em FCN: relacionamentos incompatíveis (*mismatched relationship*), categorias confusas (*confusion categories*) e classes inconspícuas (*inconspicuous classes*) (ZHAO, H. *et al.*, 2017). Estes problemas são particularmente evidentes quando as redes não conseguem capturar adequadamente informações contextuais globais da cena, limitando-se apenas a características locais dos pixels.

A Figura 9 ilustra alguns problemas de análise de cenários. No primeiro caso, objetos podem ser classificados incorretamente devido à falta de contexto global - por exemplo, um barco sendo identificado como carro devido à aparência similar, mas ignorando o contexto aquático do ambiente. No segundo caso, categorias semanticamente próximas como “edifício” e “arranha-céu” são frequentemente confundidas. O terceiro problema refere-se a objetos pequenos ou inconspícuos que podem ser ignorados ou mal classificados devido ao campo receptivo limitado das redes tradicionais. Estas limitações evidenciam a necessidade de incorporar informações contextuais globais para melhorar a precisão da segmentação semântica, especialmente em aplicações onde a compreensão do contexto é fundamental para a classificação correta dos elementos da imagem.

A arquitetura PSPNet integra um módulo de pooling piramidal (pyramid pooling module) a uma rede FCN tradicional, criando uma representação hierárquica que combina informações locais e globais (ZHAO, H. *et al.*, 2017). A Figura 10 apresenta a visão geral da arquitetura proposta, demonstrando como o módulo de pooling piramidal é incorporado após a extração de características da rede base. O módulo de pooling piramidal constitui o componente central da inovação desta arquitetura, aplicando operações de pooling em quatro níveis hierárquicos distintos, com dimensões de kernel de 1×1, 2×2, 3×3 e 6×6, respectivamente. Cada nível captura informações contextuais em diferentes escalas espaciais: o nível mais grosso (1×1) fornece pooling global, enquanto os níveis subsequentes segmentam o mapa de características em sub-regiões progressivamente menores, formando

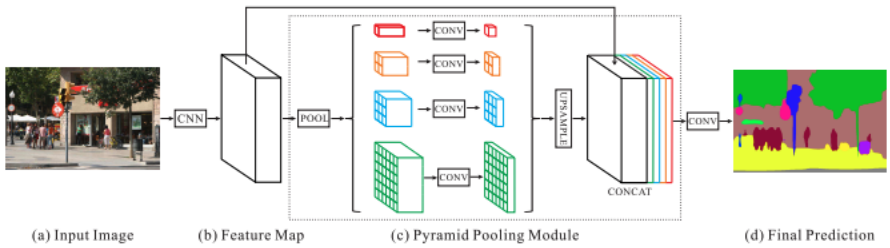
Figura 9 – Problemas de análise de cenários



Fonte: (ZHAO, H. *et al.*, 2017)

representações pooled para diferentes localizações espaciais.

Figura 10 – Arquitetura PSPNet



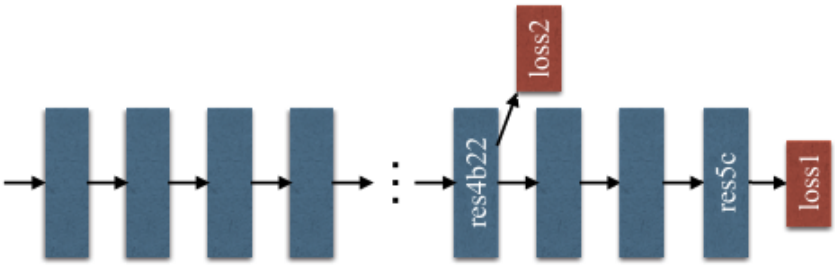
Fonte: (ZHAO, H. *et al.*, 2017)

Após cada operação de pooling piramidal, uma camada convolucional 1×1 é aplicada para reduzir a dimensionalidade da representação contextual para 1/N do tamanho original, onde N representa o número de níveis da pirâmide. Subsequentemente, os mapas de características de baixa dimensão são redimensionados através de interpolação bilinear para manter o mesmo tamanho do mapa de características original. Finalmente, as características de diferentes níveis são concatenadas para formar a representação global final do pooling piramidal. Esta abordagem hierárquica permite que a rede capture padrões contextuais que seriam perdidos em abordagens de pooling global tradicionais, proporcionando uma representação mais rica e discriminativa para tarefas de segmentação semântica complexas (ZHAO, H. *et al.*, 2017).

A PSPNet implementa uma estratégia de otimização inovadora baseada em supervisão profunda (*deep supervision*) especificamente adaptada para redes baseadas em ResNet. Esta abordagem, ilustrada na Figura 11, introduz uma função de perda auxiliar em camadas intermediárias da rede, além da função de perda principal na saída final. A supervisão auxiliar é aplicada após o bloco residual, criando um classificador adicional

que opera paralelamente ao ramo principal. Esta estratégia decompõe a otimização da rede profunda em duas tarefas complementares: a geração de resultados iniciais através da supervisão auxiliar e o aprendizado dos resíduos posteriormente através da perda final (ZHAO, H. *et al.*, 2017).

Figura 11 – Função de perda auxiliar



Fonte: (ZHAO, H. *et al.*, 2017)

2.4.6 FPN

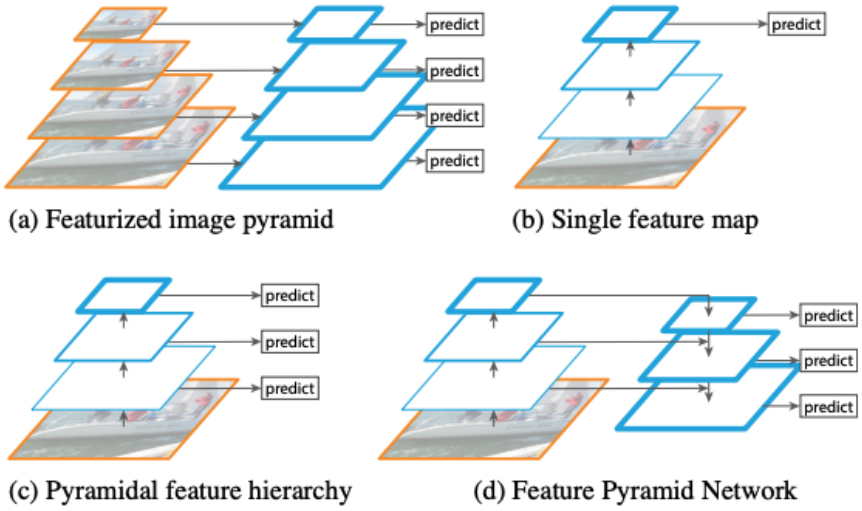
As Feature Pyramid Networks (FPN) foi concebida para superar as limitações das abordagens tradicionais de detecção de objetos em múltiplas escalas, que tradicionalmente dependiam de pirâmides de imagens computacionalmente custosas ou de representações de características em escala única com capacidade limitada para detectar objetos de tamanhos variados (LIN *et al.*, 2017a).

A motivação fundamental para o seu desenvolvimento reside na necessidade de combinar características semânticas de alto nível com informações espaciais de alta resolução. Embora as redes convolucionais profundas computem naturalmente uma hierarquia de características com formato piramidal através de suas camadas, existe uma lacuna semântica significativa entre os mapas de características de diferentes resoluções. Os mapas de alta resolução contêm características de baixo nível que prejudicam sua capacidade representacional, enquanto os mapas de baixa resolução possuem semântica forte, mas localização espacial imprecisa.

A arquitetura FPN aborda este desafio através de três componentes principais: um caminho *bottom-up* (ascendente), um caminho *top-down* (descendente) e conexões laterais. O caminho *bottom-up* corresponde à computação *feedforward* da rede backbone, que produz mapas de características em várias escalas com um fator de escalonamento de 2. É defido um nível de pirâmide para cada estágio da rede, utilizando a saída da última camada de cada estágio como conjunto de referência para construção da pirâmide. No caso específico das ResNets, são utilizadas as ativações de características dos blocos residuais finais de cada estágio, denotados como C2, C3, C4, C5 para as saídas conv2, conv3, conv4 e conv5 de 4, 8, 16, 32 pixels em relação à imagem de entrada (LIN *et al.*, 2017a).

O caminho *top-down* constitui o elemento distintivo da arquitetura FPN, realizando a *upsampling* espacial de mapas de características semanticamente mais fortes, mas espacialmente mais grosseiros, provenientes de níveis superiores da pirâmide. Este processo de "alucinação" de características de maior resolução é aprimorado através das conexões laterais, que integram características do caminho *bottom-up* via fusão *element-wise* (LIN *et al.*, 2017a). A Figura 12 ilustra claramente esta abordagem em comparação com métodos tradicionais de pirâmides de características (Figura 12a), mapas de características em escala única (Figura 12b) e hierarquias piramidais convencionais (Figura 12c).

Figura 12 – Métodos arquitetura piramidais



Fonte: (LIN *et al.*, 2017a)

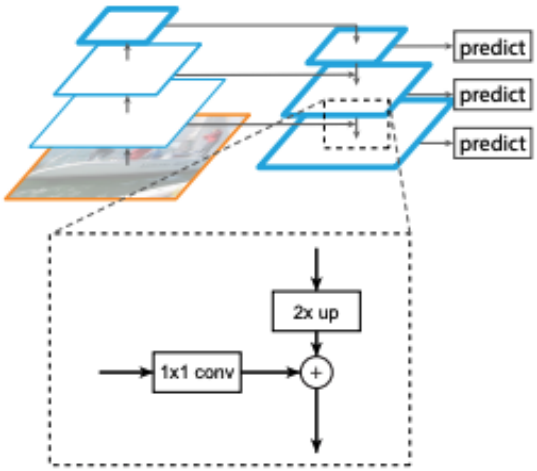
A implementação técnica das conexões laterais envolve a aplicação de uma convolução 1×1 para redução dimensional dos mapas *bottom-up*, seguida por *upsampling* por fator 2 dos mapas *top-down* utilizando interpolação por vizinho mais próximo. A fusão é realizada através de adição *element-wise*, conforme detalhado na Figura 13, que apresenta o bloco de construção fundamental da arquitetura. Posteriormente, uma convolução 3×3 é aplicada a cada mapa fusionado para gerar o conjunto final de mapas de características P2, P3, P4, P5, correspondentes aos níveis C2, C3, C4, C5 com as mesmas dimensões espaciais.

2.5 BACKBONES ENCODERS

2.5.1 Conceito de Backbone Encoder

No contexto de arquiteturas de segmentação semântica, o termo *backbone* (literalmente "espinha dorsal") refere-se à rede neural convolucional responsável pela extração de características na porção encoder da arquitetura. O backbone constitui o componente fun-

Figura 13 – Bloco de construção da FPN



Fonte: (LIN *et al.*, 2017a)

damental que determina a capacidade da rede de aprender representações discriminativas a partir das imagens de entrada (HE; GIRSHICK; DOLLÁR, 2019).

Historicamente, os backbones mais utilizados são redes de classificação de imagens pré-treinadas em grandes datasets como ImageNet, das quais são removidas as camadas de classificação finais (fully connected layers) para retenção apenas das camadas convolucionais de extração de características. Esta prática fundamenta-se no princípio de transfer learning, aproveitando representações visuais genéricas aprendidas em tarefas de classificação para aplicação em segmentação semântica (TAN, C. *et al.*, 2018).

A escolha do backbone impacta significativamente três aspectos fundamentais do modelo de segmentação:

1. **Capacidade Representacional:** Backbones mais profundos e com maior número de parâmetros geralmente possuem maior capacidade de aprender representações complexas, mas requerem mais dados de treinamento para evitar overfitting.

2. **Eficiência Computacional:** O número de parâmetros e operações de ponto flutuante (FLOPs) do backbone determina o custo computacional de treinamento e inferência do modelo.

3. **Resolução de Características:** Diferentes backbones produzem mapas de características em diferentes resoluções espaciais e profundidades de canais, influenciando a capacidade do decoder de reconstruir segmentações precisas.

Arquiteturas de segmentação modernas frequentemente permitem a substituição modular do backbone, possibilitando a avaliação de diferentes trade-offs entre acurácia e eficiência computacional sem modificar a estrutura do decoder. Esta flexibilidade arquitetural é explorada no presente trabalho através da comparação sistemática de três backbones

distintos (ResNet34, EfficientNet-B0 e EfficientNet-B3) acoplados a cinco arquiteturas de segmentação diferentes.

2.5.2 ResNet34

A família ResNet (*Residual Networks*) introduz o conceito de conexões residuais, também denominadas *skip connections*, que possibilitaram o treinamento eficiente de redes com centenas de camadas sem os problemas de degradação observados em arquiteturas anteriores (HE *et al.*, 2016).

O ResNet34, membro desta família de arquiteturas, concentra-se na estratégia de empilhar blocos de convoluções 3x3, porém incorporando inovações arquiteturais que o tornam significativamente mais eficiente. A estrutura do ResNet34 pode ser conceptualmente descrita através da configuração [6, 8, 12, 6], onde cada elemento representa o número de blocos residuais em diferentes estágios da rede, com cada bloco sendo composto por pares de convoluções 3x3 seguidas por normalização em lote e funções de ativação ReLU (*Rectified Linear Unit*) (KOONCE, 2021).

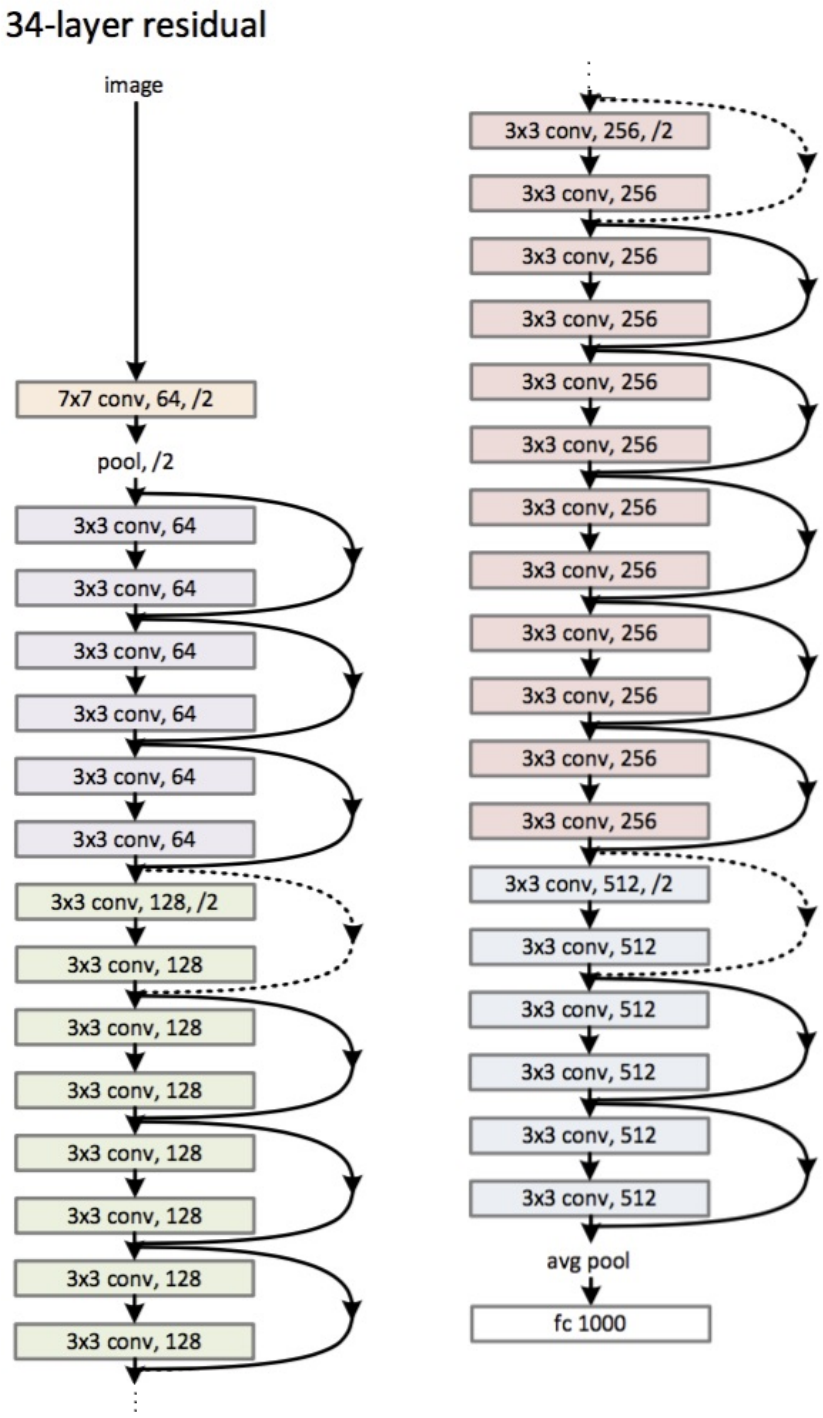
A Figura 14 apresenta a arquitetura completa do ResNet34, evidenciando a organização hierárquica dos blocos residuais ao longo dos diferentes estágios da rede, bem como as operações de *downsampling* que progressivamente reduzem a dimensionalidade espacial enquanto aumentam a profundidade dos canais de características.

O conceito fundamental que distingue o ResNet34 de arquiteturas predecessoras reside nas conexões residuais, mecanismo análogo às *skip connections* previamente discutidas no contexto de arquiteturas *encoder-decoder*, porém aplicadas com propósito distinto. Enquanto nas arquiteturas de segmentação as *skip connections* transmitem características espaciais de alta resolução do *encoder* para o *decoder*, nas ResNets estas conexões operam dentro do próprio *backbone*, permitindo que o sinal de entrada de um bloco seja adicionado diretamente à sua saída após as operações convolucionais.

Esta estratégia aborda especificamente a problemática da degradação do gradiente em redes profundas, fenômeno identificado empiricamente onde redes com maior número de camadas apresentavam desempenho inferior a redes mais rasas, não devido a *overfitting*, mas sim ao acúmulo de pequenas perdas de informação através de múltiplas camadas convolucionais que comprometia significativamente a capacidade de aprendizado durante o processo de retropropagação (HE *et al.*, 2016). A intuição fundamental é que é mais fácil otimizar uma função de mapeamento residual do que otimizar o mapeamento original não referenciado.

Matematicamente, enquanto um bloco convolucional tradicional aprende diretamente uma função de mapeamento desejada $\mathcal{H}(x)$, um bloco residual reformula o problema para aprender a função residual $\mathcal{F}(x) = \mathcal{H}(x) - x$, assumindo que a entrada x já contém informação relevante. A formulação final torna-se $\mathcal{H}(x) = \mathcal{F}(x) + x$, onde a adição da identidade $(+x)$ é realizada através da conexão de atalho. Esta reformulação facilita

Figura 14 – Arquitetura Completa do ResNet34



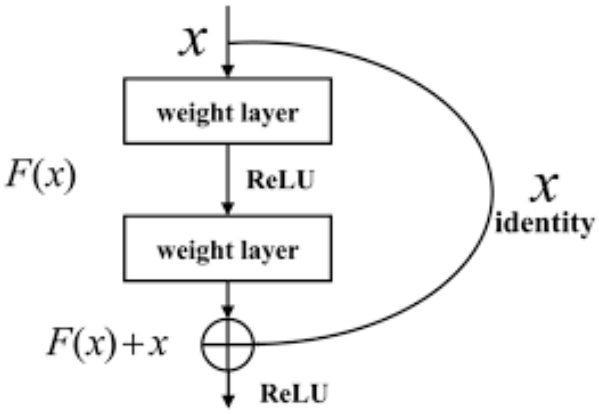
Fonte: (DIPU; SHOHAN; K. M. A., 2021)

substancialmente o processo de otimização, pois se a função de mapeamento ótima estiver próxima da identidade, é mais simples para o otimizador empurrar os pesos de $\mathcal{F}(x)$ em direção a zero do que ajustar um mapeamento completamente novo (HE *et al.*, 2016).

A Figura 15 ilustra o diagrama de um bloco residual básico, evidenciando o fluxo de informação através das camadas convolucionais (caminho principal que aprende $\mathcal{F}(x)$)

e a conexão de atalho que transmite a identidade x diretamente à saída, onde ambos os caminhos são combinados através de adição *element-wise*. Esta característica arquitetural fundamental diferencia a ResNet de suas predecessoras e viabiliza o treinamento eficaz de redes com profundidade excepcional.

Figura 15 – Diagrama do Bloco Residual Básico



Fonte: (HE *et al.*, 2016)

A estrutura inicial do ResNet34 emprega uma convolução 7x7 com *stride* 2, seguida por uma camada de *max pooling* 3x3, com o objetivo de reduzir rapidamente a dimensionalidade espacial da entrada de 224×224 para 56×56 pixels e preparar os dados para o processamento subsequente nos blocos residuais (KOONCE, 2021). Embora esta abordagem tenha sido eficaz considerando as limitações computacionais da época de seu desenvolvimento, pesquisas posteriores têm demonstrado a existência de estratégias mais otimizadas para o processamento da camada *stem* (camada inicial), conforme será abordado na discussão sobre a família EfficientNet.

Um componente essencial da arquitetura ResNet34 é a normalização em lote (*Batch Normalization*), técnica que normaliza as ativações de cada camada considerando a média e o desvio padrão dos dados observados no *minibatch* corrente. Esta técnica possui a propriedade de suavizar a paisagem de otimização (*loss landscape*), resultando em um processo de retropropagação significativamente mais estável e acelerando a convergência da rede em aproximadamente uma ordem de magnitude em comparação com redes sem normalização (IOFFE; SZEGEDY, 2015). Adicionalmente, a normalização em lote introduz ruído estocástico durante o treinamento devido à variabilidade estatística entre diferentes *minibatches*, conferindo à técnica propriedades regularizadoras implícitas que auxiliam na generalização do modelo e reduzem a necessidade de outras técnicas de regularização como *Dropout* (IOFFE; SZEGEDY, 2015).

A eficiência da ResNet34 como *backbone encoder* para tarefas de segmentação semântica fundamenta-se em três aspectos principais: a capacidade de aprender representações hierárquicas profundas sem degradação de desempenho, a produção de mapas de características em múltiplas escalas espaciais (úteis para conexões *skip* em arquitetu-

ras *encoder-decoder*), e o equilíbrio favorável entre capacidade representacional e custo computacional, com aproximadamente 21,8 milhões de parâmetros (HE *et al.*, 2016).

2.5.3 EfficientNet-B0 e B3

A arquitetura EfficientNet resulta da convergência de múltiplas linhas de pesquisa desenvolvidas ao longo de vários anos, integrando técnicas anteriormente exploradas em contextos distintos (KOONCE, 2021). De particular interesse é o fato de que pesquisas originalmente direcionadas para dispositivos móveis passaram a influenciar substancialmente o desenvolvimento de modelos para ambientes computacionais de alto desempenho, invertendo a tradicional direção de transferência tecnológica que historicamente privilegiava a adaptação de soluções desenvolvidas para clusters de GPUs para dispositivos com restrições de recursos (KOONCE, 2021).

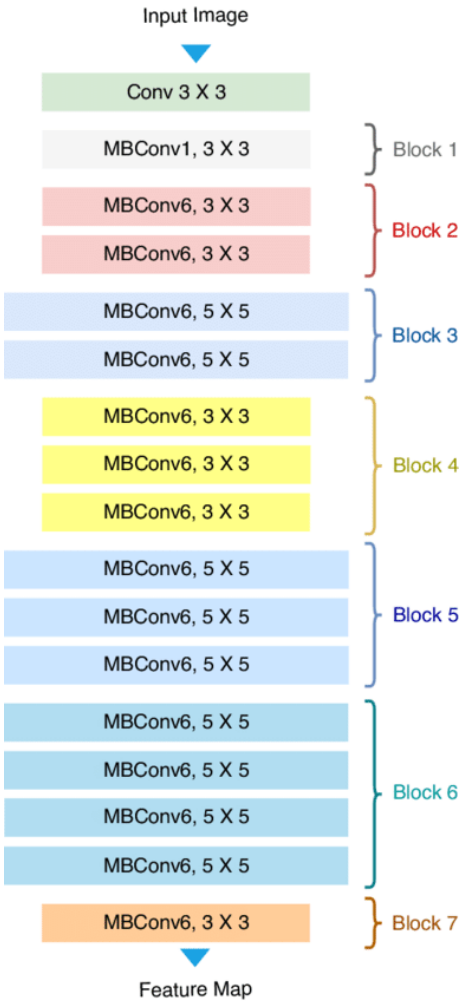
Em termos arquiteturais, o EfficientNet fundamenta-se na utilização de blocos residuais invertidos (*inverted residual blocks*), originalmente propostos na arquitetura MobileNetV2 (HOWARD *et al.*, 2017), combinados com a estratégia de busca neural automática MnasNet (TAN, M.; LE, Q., 2019). A sinergia entre estes componentes permitiu aos pesquisadores identificar um conjunto significativamente aprimorado de configurações de rede, superando as limitações das estratégias evolutivas anteriormente exploradas em termos de escalabilidade e reprodutibilidade (KOONCE, 2021; TAN, M.; LE, Q., 2019).

A arquitetura do EfficientNet-B0, conforme ilustrada na Figura 16, demonstra a composição modular baseada em blocos MBConv (Mobile Inverted Bottleneck Convolution), que constituem os elementos fundamentais de construção da rede. Observa-se a progressiva redução da resolução espacial acompanhada do aumento da profundidade dos canais ao longo dos estágios da rede.

Um dos componentes inovadores integrados à arquitetura EfficientNet é a função de ativação *swish*, descoberta através de técnicas de busca automática de funções de ativação(RAMACHANDRAN; ZOPH; LE, Q. V., 2017). Matematicamente, a função *swish* é definida como $f(x) = x \cdot \text{sigmoid}(\beta x)$, onde $\text{sigmoid}(y) = \frac{1}{1+e^{-y}}$. Esta formulação apresenta a propriedade singular de assumir valores ligeiramente negativos em torno de zero, diferenciando-se das funções de ativação tradicionais que são estritamente não-negativas (RAMACHANDRAN; ZOPH; LE, Q. V., 2017). Conceitualmente, esta característica produz um espaço de gradientes mais suave, facilitando o processo de aprendizado da distribuição subjacente dos dados pela rede e, consequentemente, resultando em melhorias mensuráveis de acurácia (KOONCE, 2021). Apesar de suas vantagens, a função *swish* apresenta limitações do ponto de vista implementacional, notadamente um maior consumo de memória em comparação com a função ReLU, trade-off que deve ser considerado conforme os recursos computacionais disponíveis.

A família EfficientNet engloba múltiplas variantes, desde o modelo base EfficientNet-B0 até versões significativamente maiores como EfficientNet-B7, cada uma oferecendo

Figura 16 – Diagrama da EfficientNet-B0



Fonte: (AHMED; SABAB, 2022)

diferentes trade-offs entre acurácia e eficiência computacional. O EfficientNet-B0, com aproximadamente 5 milhões de parâmetros, representa a configuração mais leve, adequada para aplicações com severas restrições de recursos, enquanto o EfficientNet-B3, com cerca de 12 milhões de parâmetros, oferece um ponto de equilíbrio entre capacidade representacional e eficiência computacional (TAN, M.; LE, Q., 2019; DURMUŞ *et al.*, 2025).

No presente trabalho, a seleção dos backbones EfficientNet-B0 e EfficientNet-B3 para análise comparativa fundamenta-se na necessidade de avaliar diferentes pontos no espectro de trade-offs entre eficiência computacional e capacidade representacional. O EfficientNet-B0, com sua arquitetura compacta de aproximadamente 5 milhões de parâmetros, representa uma opção atrativa para cenários onde os recursos computacionais são limitados ou quando o tempo de inferência é crítico. Por outro lado, o EfficientNet-B3, com aproximadamente 12 milhões de parâmetros, oferece maior capacidade de aprendizado de representações complexas.

2.6 MEDIDAS DE DESEMPENHO

Para determinar a qualidade dos resultados da segmentação e comparar diferentes algoritmos ou ajustar seus parâmetros, é essencial utilizar métricas de avaliação. Essas métricas quantificam o quão bem a segmentação gerada por um algoritmo se alinha com a segmentação verdadeira ou ideal, conhecida como *ground truth* (verdade fundamental).

2.6.1 Métricas de Sobreposição

O IoU, também conhecido como Índice de Jaccard, é uma métrica que mede a similaridade entre duas regiões ou formas. Didaticamente, imagine duas formas – uma é a área segmentada pelo seu algoritmo (predição) e a outra é a área real do objeto na imagem (verdade fundamental). O IoU calcula a razão entre a área de sobreposição (a parte onde as duas formas se encontram, ou a interseção) e a área total coberta por ambas as formas juntas (a união) (REZATOFIGHI *et al.*, 2019; MÜLLER; SOTO-REY; KRAMER, 2022; GARCIA-GARCIA *et al.*, 2017). A fórmula é expressa como:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

Um valor de IoU de 1 indica uma sobreposição perfeita, o que significa que a segmentação prevista é idêntica à verdade fundamental. Um valor de 0 indica que não há sobreposição alguma. Uma das propriedades mais atraentes do IoU é sua invariância à escala. Isso significa que a similaridade entre duas formas é avaliada independentemente do seu tamanho absoluto na imagem, o que é crucial em diversas aplicações, onde os objetos de interesse podem variar significativamente em escala. Além disso, o IoU é frequentemente utilizado como a métrica padrão em muitos desafios e benchmarks de detecção e segmentação de objetos (REZATOFIGHI *et al.*, 2019).

Apesar de suas vantagens, o IoU possui uma fraqueza notável quando usado como função de perda para otimização em redes neurais. Se a segmentação prevista e a verdade fundamental não se sobrepõem de forma alguma, o IoU resultante é 0. Nesse cenário, o gradiente do IoU também se torna zero, o que impede que o algoritmo de otimização saiba em qual direção mover a segmentação prevista para que ela comece a se aproximar da verdade fundamental. Isso cria um platô de otimização, dificultando o treinamento para casos de baixa sobreposição (REZATOFIGHI *et al.*, 2019).

O Dice Score, também conhecido como F1-score para tarefas de classificação binária, é outra métrica de sobreposição popular que mede a similaridade entre a segmentação prevista e a verdade fundamental (MÜLLER; SOTO-REY; KRAMER, 2022; THOMA, 2016; SHAMIR *et al.*, 2019). Sua fórmula é:

$$\text{Dice Score} = \frac{2TP}{(2TP + FP + FN)} \tag{2}$$

Onde:

- **TP (True Positives)=** são os pixels que foram corretamente classificados como parte do objeto (pixels da verdade fundamental que foram previstos como objeto)
- **FP (False Positives)** são os pixels que foram incorretamente classificados como parte do objeto (pixels do fundo que foram previstos como objeto).
- **FN (False Negatives)** são os pixels que foram incorretamente classificados como parte do fundo (pixels do objeto que foram previstos como fundo).

Um valor de Dice Score de 1 indica uma sobreposição perfeita, enquanto um valor de 0 indica ausência total de sobreposição. O Dice Score é especialmente prevalente e recomendado na avaliação de segmentação de imagens médicas. Isso se deve em parte à sua capacidade de penalizar falsos positivos e falsos negativos, que são importantes para a segurança e precisão em diagnósticos médicos (MÜLLER; SOTO-REY; KRAMER, 2022; SHAMIR *et al.*, 2019).

É importante notar que, embora o Dice Score e o IoU sejam métricas baseadas em sobreposição e ambas eficazes, o IoU penaliza mais severamente a subsegmentação (FN) e a supersegmentação (FP) do que o Dice Score. Isso significa que pequenas discrepâncias nos limites da segmentação podem resultar em uma penalidade maior no IoU. No entanto, o Dice Score é amplamente aceito e o mais utilizado em muitas publicações científicas para avaliação de segmentação de imagens médicas (MÜLLER; SOTO-REY; KRAMER, 2022).

Para lidar com a natureza probabilística das saídas de muitos algoritmos de segmentação modernos (que produzem mapas de confiança em vez de imagens binárias diretas) e para reduzir o viés relacionado ao tamanho da estrutura, uma extensão, o Dice Contínuo (cDC), foi proposta. O cDC permite a comparação direta de um mapa probabilístico com uma verdade fundamental binária, considerando erros parciais e/ou pixels/voxels de baixa confiança. Em contraste, o Dice Clássico geralmente exige um passo de binarização (aplicando um limiar) e trata todos os erros igualmente, sem levar em conta a confiança do pixel. O cDC demonstrou ser menos viesado pelo tamanho da estrutura e mais robusto aos efeitos de volume parcial, refletindo melhor a qualidade real da segmentação, especialmente para estruturas pequenas, como o núcleo subtalâmico (SHAMIR *et al.*, 2019).

2.6.2 Métrica de Precisão

Essas métricas derivam da matriz de contingência, que organiza os resultados de classificação em quatro categorias: Verdadeiros Positivos, Falsos Positivos, Verdadeiros Negativos e Falsos Negativos.

A Precisão, também conhecida como Valor Preditivo Positivo (Positive Predictive Value - PPV), é uma métrica que mede a exatidão ou pureza das predições positivas do modelo (POWERS, 2020). Sua fórmula é dada por:

$$\text{Precisão} = \frac{TP}{TP + FP} \tag{3}$$

Um alto valor de Precisão indica que, quando o modelo prevê a presença de um objeto, essa predição é muito provável de ser correta, significando que o algoritmo apresenta um baixo número de Falsos Positivos (POWERS, 2020). Em aplicações médicas, como a segmentação de tumores, a Precisão é crucial pois Falsos Positivos podem gerar ansiedade desnecessária ao paciente ou indicar procedimentos invasivos desnecessários (LIPTON; ELKAN; NARYANASWAMY, 2014).

O Recall, também denominado Sensibilidade ou Taxa de Verdadeiros Positivos (True Positive Rate - TPR), complementa a Precisão avaliando a capacidade do modelo de detectar todos os casos positivos (POWERS, 2020). Esta métrica é definida por:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

Um Recall elevado indica que o modelo possui baixa taxa de Falsos Negativos, ou seja, raramente deixa de detectar objetos que deveriam ser identificados (POWERS, 2011). Em contextos clínicos, como a detecção de lesões cancerígenas, um Recall alto é prioritário, pois um Falso Negativo pode resultar em um diagnóstico não realizado e consequente atraso no tratamento, com implicações potencialmente graves para o paciente.

Vale a pena enfatizar que Precisão e Recall apresentam uma relação de compromisso (trade-off): aumentar o limiar de decisão do classificador tende a aumentar a Precisão às custas do Recall, e vice-versa. Este fenômeno ocorre porque tornar o modelo mais "conservador" (aumentando o limiar) reduz tanto os Verdadeiros Positivos quanto os Falsos Positivos, enquanto torná-lo mais "liberal" (reduzindo o limiar) aumenta ambos (POWERS, 2020).

Um aspecto crítico é que tanto a Precisão quanto o Recall são métricas enviesadas (*biased*), pois ignoram completamente os Verdadeiros Negativos. Um algoritmo que classifica todos os pixels como positivos alcançaria Recall de 100%, mas teria Precisão muito baixa. Similarmente, um algoritmo extremamente conservador que classifica apenas poucos pixels com altíssima confiança poderia ter Precisão de 100%, mas Recall muito baixo, perdendo a maioria dos objetos de interesse (POWERS, 2020).

2.7 TRABALHOS RELACIONADOS

A aplicação de técnicas de visão computacional e inteligência artificial na odontologia tem evoluído significativamente nas últimas duas décadas. Esta seção apresenta os principais trabalhos desenvolvidos para detecção e segmentação de cáries dentárias em radiografias, estabelecendo o contexto científico e identificando as lacunas dos trabalhos atuais.

2.7.1 Segmentação de Cáries em Odontologia

2.7.1.1 Arquiteturas Encoder-Decoder Clássicas

A aplicação de arquiteturas *encoder-decoder* para segmentação dental ganhou destaque com trabalhos como o de (SIVAGAMI *et al.*, 2020), que implementaram a arquitetura U-Net para segmentação de imagens panorâmicas dentárias, obtendo 95,2% de acurácia na delimitação de estruturas anatômicas. Embora tenha sido pioneiro na aplicação de U-Net em radiografias panorâmicas, o trabalho não reportou métricas como *Dice Score* ou IoU, limitando comparações diretas com estudos subsequentes.

Com foco em semântica contextual e aprimoramento de contraste, (HOU *et al.*, 2023) propuseram o *Teeth U-Net*, um modelo especificamente adaptado para radiografias panorâmicas com foco em semântica contextual e aprimoramento de contraste. A abordagem demonstrou melhorias significativas na segmentação de dentes individuais, alcançando *Dice Score* de 0,8913 e IoU de 0,8045. No entanto, o modelo apresentou limitações na detecção de lesões pequenas, evidenciando a necessidade de arquiteturas especializadas para cáries.

Visando lidar com os limites borrados entre tecido periodontal e dentário, (NISHITANI; NAKAYAMA *et al.*, 2021) desenvolveram uma variação da U-Net com função de perda ponderada nas bordas dos dentes em radiografias panorâmicas. Esta abordagem demonstrou maior precisão na delimitação de contornos em comparação com a U-Net tradicional.

2.7.1.2 Arquiteturas com Mecanismos de Atenção

Mecanismos de atenção baseados em canal foram explorados por (LI, W. *et al.*, 2023), que desenvolveram uma *Attention U-Net* utilizando módulos *Squeeze-and-Excitation* para filtrar apenas os canais relevantes e impactantes, melhorando o desempenho na segmentação de cáries em radiografias *bite-wing*.

Uma abordagem em dois estágios para identificar limites grosseiros de dentes foi proposta por (ZHAO, Y. *et al.*, 2024) com a TSASNet (*Two-Stage Attention Segmentation Network*). O primeiro estágio foca na extração de características globais para localizar aproximadamente os limites dos dentes, enquanto o estágio de refinamento utiliza mecanismos de atenção para aumentar a precisão da segmentação.

A combinação de arquitetura TransUNet avançada com mecanismos de atenção espacial e de canal foi apresentada por (SUN; WANG, J. *et al.*, 2024), que introduziram a STS-TransUNet nos segmentos decodificadores. O modelo emprega técnicas de supervisão profunda para superar o problema do gradiente de desvanecimento, demonstrando eficácia particular em cenários semi-supervisionados.

Para abordar o desbalanceamento extremo característico da segmentação de cáries, (PRIYA; RAJA, 2025) desenvolveram a *Caries-SegNet*, uma rede *encoder-decoder* em

cascata multi-escala com atenção híbrida espacial-canal. O modelo implementa *Deep Fully Connected Residual Blocks* (DFCR) para fornecer características relevantes sem perda de informação significativa, abordando o desbalanceamento extremo característico da segmentação de cáries.

2.7.1.3 Abordagens Multi-estágio e Cascata

Métodos baseados em cascata têm emergido como uma solução promissora para melhorar a precisão na detecção e segmentação de cáries. Visando reduzir a dependência de dados anotados manualmente, (QAYYUM; TAHIR; BUTT *et al.*, 2023) propuseram uma abordagem de aprendizado semi-supervisionado que divide o processo em duas etapas distintas.

Um método de aprendizado em cascata foi desenvolvido por (PANYARAK *et al.*, 2024), que primeiro realiza a detecção de cáries utilizando YOLOX para gerar caixas delimitadoras, seguido por segmentação semântica com FCN-8s nas regiões recortadas. Esta abordagem em dois estágios alcançou 68,35% de precisão na detecção e 84,3% de DSC na segmentação em um *dataset* com 5.569 radiografias periapicais, demonstrando que a detecção prévia de cáries antes da segmentação melhora significativamente o desempenho do modelo.

2.7.1.4 Segmentação Multi-classe de Cáries

A segmentação de diferentes estágios de lesões cariosas foi abordada por (ZHU, H.; CAO; LIAN, Luya *et al.*, 2021), que propuseram a CariesNet, uma arquitetura em forma de U com módulo de atenção axial em escala completa (FSSA) para segmentar cáries rasas, moderadas e profundas em radiografias panorâmicas. Utilizando um *dataset* de 3.127 lesões bem delineadas, o modelo alcançou DSC médio de 93,64%, superando métodos *state-of-the-art* como PraNet. O módulo FSSA demonstrou particular eficácia na captura de informações contextuais amplas, essencial para segmentação de lesões de tamanho reduzido.

Para segmentação de três tipos de cáries em localizações diferentes (oclusal, proximal e cervical), (DAYI; ÜZEN *et al.*, 2023) desenvolveram a DCDNet (*Dental Caries Detection Network*). A arquitetura incorpora uma estrutura *Multi-Predicted Output* (MPO), onde o mapa de características final é dividido em três caminhos distintos para detectar cada tipo de cárie. Em experimentos com 504 radiografias panorâmicas, o modelo alcançou F1-score médio de 62,79%, demonstrando capacidade de detecção em diferentes localizações com alta taxa de sucesso.

2.7.1.5 Abordagens Recentes e Tendências

Um *dataset* pediátrico abrangente foi criado por (ZHANG, Y. *et al.*, 2023), contendo 106 radiografias panorâmicas de crianças (2-13 anos) combinadas com 2.692 imagens de adultos, totalizando 3.187 pares de radiografias e máscaras correspondentes. Este *dataset* tem sido amplamente utilizado em estudos subsequentes para treinamento e validação de modelos.

O papel da segmentação U-Net como pré-processamento para classificação de cáries foi investigado por (YASSAR; FITRIA *et al.*, 2024), utilizando ResNet-50, InceptionV3 e ResNeXt-50.. Os resultados demonstraram que a segmentação prévia dos dentes melhora significativamente a acurácia diagnóstica ao remover estruturas irrelevantes como língua, lábios e gengivas.

Para lidar com a escassez de dados anotados, (WANG, X. *et al.*, 2023) propuseram um *framework* de aprendizado com consciência de incerteza multi-nível para segmentação semi-supervisionada de cáries panorâmicas, utilizando o *dataset* MLUA. A abordagem combina aprendizado supervisionado e não-supervisionado para lidar com a escassez de dados anotados, alcançando melhorias significativas em cenários de dados limitados, particularmente relevante dado que *datasets* de cáries raramente excedem 200 imagens anotadas.

2.7.1.6 Desafios Persistentes

A revisão da literatura revela desafios consistentes na segmentação de cáries:

1. **Escassez de Dados Anotados:** A maioria dos estudos trabalha com *datasets* de 100-500 imagens, limitando a capacidade de generalização dos modelos (LIAN *et al.*, 2021).
2. **Desbalanceamento Extremo de Classes:** Lesões cariosas frequentemente representam menos de 1% dos pixels em radiografias panorâmicas, dificultando o treinamento (PRIYA; RAJA, 2025).
3. **Variabilidade Inter-observador:** A anotação de lesões cariosas por diferentes especialistas apresenta inconsistências, afetando a qualidade dos dados de treinamento (OZTEKIN *et al.*, 2023).
4. **Heterogeneidade de Métricas:** A falta de padronização nas métricas reportadas (alguns trabalhos usam apenas acurácia, outros *Dice Score*, IoU ou F1-score) dificulta comparações diretas entre abordagens (SIVAGAMI *et al.*, 2020; HOU *et al.*, 2023).
5. **Generalização *Cross-Dataset*:** Modelos treinados em um tipo de radiografia (periapical, *bite-wing* ou panorâmica) frequentemente apresentam desempenho reduzido quando testados em outras modalidades (DAYI; ÜZEN *et al.*, 2023).

2.7.1.7 Análise Comparativa das Principais Contribuições

A Tabela 1 apresenta uma síntese comparativa dos principais trabalhos de segmentação de cáries, destacando arquiteturas, *datasets* utilizados, métricas de desempenho e principais limitações identificadas.

Tabela 1 – Comparação atualizada dos principais trabalhos de segmentação de cáries

Trabalho	**Ano**	**Arquitetura**	**Dataset (n)**	**Contribuição Principal**
(SIVAGAMI *et al.*, 2020)	2020	U-Net	Próprio (NR)	Primeira aplicação U-Net panorâmica
(NISHITANI; NAKAYAMA *et al.*, 2021)	2021	U-Net + Edge Loss	Próprio (NR)	Função de perda em bordas
(ZHU, H.; CAO; LIAN, Luya *et al.*, 2021)	2021	CariesNet	Próprio (3.127 lesões)	Módulo FSSA para multi-stage
(HOU *et al.*, 2023)	2023	Teeth U-Net	Próprio (152)	Context semantics enhancement
(DAYI; ÜZEN *et al.*, 2023)	2023	DCDNet	Próprio (504)	Multi-location segmentation
(QAYYUM; TAHIR; BUTT *et al.*, 2023)	2023	Semi-supervised	Próprio (NR)	Redução dependência anotação
(WANG, X. *et al.*, 2023)	2023	Custom CNN	MLUA (100)	Uncertainty learning
(SUN; WANG, J. *et al.*, 2024)	2024	STS-TransUNet	Próprio (NR)	Atenção espacial-canal + TransUNet
(PANYARAK *et al.*, 2024)	2024	YOLOX + FCN-8s	Periapical (5.569)	Cascata detecção-segmentação
(ZHAO, Y. *et al.*, 2024)	2024	TSASNet	Panorâmica (NR)	Two-stage attention
(YASSAR; FITRIA *et al.*, 2024)	2024	U-Net + CNNs	Próprio (562)	Segmentação como pré-processamento
(PRIYA; RAJA, 2025)	2025	Caries-SegNet	Periapical (1.400)	Atenção híbrida espacial-canal

Fonte: Autor

Nota: NR = Não reportado

Esta análise evidencia a evolução das arquiteturas de segmentação de cáries, com tendência crescente ao uso de mecanismos de atenção, abordagens multi-estágio e técnicas de aprendizado semi-supervisionado para superar as limitações de dados anotados.

2.7.2 Estudos Comparativos de Arquiteturas

A seleção de arquiteturas e *backbones* adequados para segmentação semântica em imagens médicas tem sido objeto de extensa investigação científica. Estudos comparativos sistemáticos fornecem *insights* fundamentais sobre as vantagens e limitações de diferentes abordagens, auxiliando pesquisadores na escolha de modelos apropriados para suas aplicações específicas.

2.7.2.1 Estudos Comparativos em Imagens Médicas Gerais

A avaliação comparativa de arquiteturas de segmentação em diferentes modalidades de imagens médicas tem revelado padrões consistentes de desempenho. (YIN *et al.*, 2022)

realizaram uma revisão abrangente de variantes da arquitetura U-Net em segmentação de imagens médicas, documentando mais de 2.500 citações acadêmicas e categorizando as metodologias relacionadas quanto à estrutura, inovação e eficiência. A revisão identificou que a U-Net e suas variantes continuam sendo a arquitetura dominante devido à sua capacidade de segmentar alvos pequenos com precisão e sua arquitetura de rede escalável.

(CHEN, X. *et al.*, 2023) compararam seis modelos de segmentação semântica (U-Net, U-Net3+, FCN-32s, SegNet, DeepLabV3+ e PSPNet) para detecção de câncer de mama em imagens de ultrassom de 309 pacientes. Os resultados demonstraram que a arquitetura U-Net3+ alcançou a maior acurácia global com melhores valores de acurácia média (0,90), IoU médio (0,82) e F1-score ponderado (0,83). Notavelmente, embora DeepLabV3+ tenha obtido acurácia global de 0,91, a U-Net3+ superou-a em termos de desempenho geral quando consideradas todas as métricas conjuntamente.

(SETIO *et al.*, 2024) conduziram um estudo comparativo entre DeepLabV3+, U-Net++ e Attention U-Net para segmentação de células sanguíneas utilizando o *dataset* BCCD. Os resultados demonstraram que U-Net++ se destacou em acurácia (0,9740), precisão (0,9511) e F1-score (0,9576), enquanto Attention U-Net alcançou o maior *recall* (0,9692). DeepLabV3+ apresentou desempenho equilibrado em todas as métricas. A análise qualitativa revelou que U-Net++ entregou segmentação superior de estruturas celulares complexas e sobrepostas, enquanto Attention U-Net exibiu sensibilidade excepcional a aglomerados celulares densos.

(VERMA; MOSTAFID, 2024) compararam U-Net e DeepLabv3+ para segmentação de tumores cerebrais no *dataset* BraTS 2020, destacando que a escolha entre segmentação 2D e 3D depende dos recursos disponíveis e do equilíbrio desejado entre acurácia e eficiência. Para tumores cerebrais, abordagens 2D podem ser preferíveis com recursos limitados ou quando o tumor é majoritariamente visível em uma única fatia.

2.7.2.2 Comparações de Arquiteturas em Odontologia

No contexto odontológico, estudos comparativos específicos têm sido conduzidos para avaliar o desempenho de diferentes arquiteturas. (DURMUŞ *et al.*, 2025) realizaram uma análise comparativa abrangente de quatro modelos de segmentação baseados em pixels (U-Net, FPN, PSPNet e LinkNet) combinados com 10 diferentes *backbones* para detecção de dentes impactados em radiografias panorâmicas.

Os achados principais deste estudo incluem:

- **Melhor Arquitetura:** U-Net alcançou o melhor desempenho médio geral, com IoU médio de 85,29% quando combinada com EfficientNetB7 como *backbone*.
- **Desempenho por Arquitetura:** A análise revelou que modelos baseados em U-Net e FPN demonstraram desempenho similar na maioria dos cenários, enquanto PSPNet apresentou os resultados mais baixos (IoU médio de 83%).

- **Consistência entre Arquiteturas:** LinkNet e FPN apresentaram desempenho aproximadamente equivalente, com scores de IoU na faixa de 84%, demonstrando que diferentes arquiteturas podem alcançar resultados comparáveis quando adequadamente configuradas.

(IMAK, A. *et al.*, 2021a) desenvolveram uma arquitetura U-Net aprimorada com bloco residual invertido e ConvLSTM bidirecional para detecção de dentes impactados em radiografias panorâmicas, alcançando 99,82% de acurácia, 91,59% de F1-score, 84,48% de IoU e 90,71% de *recall*. Este trabalho demonstrou que modificações arquiteturais específicas podem resultar em melhorias significativas de desempenho.

2.7.2.3 Análise Comparativa de Backbones

A escolha do *backbone* tem impacto fundamental no desempenho de modelos de segmentação. (DURMUŞ *et al.*, 2025) conduziram análise extensiva comparando 10 arquiteturas de *backbone* diferentes (ResNet101, ResNeXt101, InceptionV3, InceptionResNetV2, DenseNet121, SEResNet101, SEResNeXt101, VGG16, MobileNet e EfficientNetB7) em quatro modelos de segmentação.

Os resultados revelaram que:

1. **EfficientNetB7** foi consistentemente o *backbone* de melhor desempenho em todas as arquiteturas de segmentação testadas, demonstrando capacidade superior de extração de características.
2. **DenseNet121, InceptionV3 e InceptionResNetV2** apresentaram o segundo melhor desempenho, com IoU aproximado de 81%.
3. **ResNet101** obteve score de IoU médio mais baixo (78,79%) em comparação com outros modelos, apesar de sua popularidade na literatura.
4. **MobileNet, VGG16, ResNeXt101 e SEResNeXt101** demonstraram desempenho comparável, com IoU aproximado de 79%, sugerindo que *backbones* com diferentes complexidades podem alcançar resultados similares.

(KUMAR, S. *et al.*, 2024) investigaram o acoplamento de DeepLabV3+ com quatro CNNs base (DarkNet53, SqueezeNet, EfficientNet-b0 e DarkNet19) para segmentação de tumores de mama em imagens de ultrassom. DeepLabV3+ acoplado com DarkNet53 e EfficientNet-b0 apresentou melhor desempenho, com acurácia global de 96,50% e 96,18%, respectivamente, destacando a importância da seleção adequada do *backbone* para tarefas específicas.

2.7.2.4 Trade-offs entre Desempenho e Eficiência Computacional

Estudos comparativos têm enfatizado a importância de considerar não apenas a acurácia, mas também a eficiência computacional. A Tabela 2 apresenta uma síntese dos *trade-offs* identificados na literatura.

Tabela 2 – *Trade-offs* entre acurácia e eficiência em arquiteturas de segmentação

Arquitetura	**Acurácia**	**Velocidade**	**Parâmetros**	**Aplicação Recomendada**
U-Net	Alta	Moderada	Moderado	Segmentação geral balanceada
U-Net++	Muito Alta	Moderada-Baixa	Alto	Estruturas complexas e sobrepostas
Attention U-Net	Alta	Moderada	Moderado-Alto	Alvos pequenos aglomerados densos
DeepLabV3+	Alta	Baixa	Alto	Segmentação multi-escala
PSPNet	Moderada-Alta	Moderada	Moderado-Alto	Informação contextual global
FPN	Alta	Alta	Moderado	Objetos em múltiplas escalas
LinkNet	Moderada-Alta	Muito Alta	Baixo	Aplicações tempo-real

Fonte: Adaptado de (DURMUŞ *et al.*, 2025), (SETIO *et al.*, 2024)

(SETIO *et al.*, 2024) observaram que DeepLabV3+ apresentou maior custo computacional em comparação com U-Net++ e Attention U-Net, atribuído à sua arquitetura mais profunda. Em contraste, U-Net++ e Attention U-Net demonstraram tempos de inferência mais rápidos, destacando seu potencial para implantação em tarefas de imagens biomédicas sensíveis ao tempo.

2.7.2.5 Variantes Avançadas e Arquiteturas Híbridas

Recentemente, arquiteturas híbridas que combinam CNNs com mecanismos de atenção ou *transformers* têm demonstrado resultados promissores. (CHEN, J. *et al.*, 2024) propuseram TransUNet, que repensa o *design* da arquitetura U-Net através da lente de *transformers*. O modelo alcançou melhoria média significativa de Dice de 1,06% para segmentação multi-órgãos e 4,30% para segmentação de tumores pancreáticos em comparação com o altamente competitivo nn-UNet.

(ZHOU, Y. *et al.*, 2024) testaram a eficácia de nnUNetv2, DeepLabV3+, *Medical Transformer*, SwinUnet e TransUNet no *dataset* de lesões ósseas. A análise comparativa revelou que TransUNet apresentou o melhor desempenho entre os modelos testados, embora nenhum dos experimentos tenha superado a *baseline* estabelecida, destacando os desafios persistentes em segmentação universal de lesões.

2.7.3 Aplicações com *Datasets* Limitados

A escassez de dados anotados representa um dos maiores obstáculos ao desenvolvimento de modelos de *deep learning* em imagens médicas e odontológicas. O alto custo de anotação, a necessidade de especialistas qualificados e restrições de privacidade limitam significativamente o tamanho dos *datasets* disponíveis. Esta seção revisa as principais

estratégias desenvolvidas para mitigar essas limitações e permitir o treinamento eficaz de modelos com dados escassos.

2.7.3.1 Técnicas de Data Augmentation

A augmentação de dados permanece como uma das técnicas mais amplamente adotadas para expandir conjuntos de dados limitados. (IMAK, A. *et al.*, 2021b) utilizaram seis métodos de augmentação (recorte central, recorte aleatório, rotação aleatória de 90°, distorção de grade, espelhamento horizontal e vertical) em um *dataset* inicial de 304 radiografias panorâmicas com dentes impactados, expandindo-o para 2.128 imagens. Os autores enfatizam que a augmentação de dados é um método extremamente útil para expandir conjuntos de dados muito pequenos, com potencial de aumentar significativamente o desempenho do sistema.

(TASSOKER; ÖZIÇ; YUCE, 2024) enfrentaram a limitação de um *dataset* pequeno de apenas 175 radiografias panorâmicas diagnosticadas com osteosclerose idiopática, condição com incidência de apenas 2,7% na população turca. Para superar essa limitação, aplicaram augmentação por espelhamento horizontal, resultando em um *dataset* expandido de 350 radiografias, divididas em aproximadamente 70% para treinamento (245), 15% para validação (53) e 15% para teste (52).

(PARK; HUH; LEE, J., 2023) empregaram augmentação extensiva em um grande *dataset* multicentro de 156.965 imagens radiográficas de implantes dentários. O *dataset* de treinamento foi expandido 10 vezes através de rotações aleatórias (90°), ajustes de matiz (-0,1 a 0,1), brilho (-0,12 a 0,12), saturação (0,6-1,5), contraste (0,6-1,5), adição de ruído (0,05) e espelhamentos horizontal e vertical.

(ALHUSSEIN *et al.*, 2023) relataram que a qualidade das radiografias panorâmicas é essencial para detecção precisa de cavidades, observando que algumas radiografias de baixa qualidade dificultam a detecção de lesões anormais. Após experimentos extensivos com diferentes divisões do *dataset* (80/20%, 90/10%, 80/10/10%, 60/40%), concluíram que o processo de divisão de dados não afetou ou aprimorou significativamente o desempenho do modelo, indicando que o desempenho depende primariamente do tipo de *dataset* e da lesão alvo para detecção.

2.7.3.2 Transfer Learning e Pré-treinamento

O aprendizado por transferência tem sido fundamental para treinar modelos eficazes com *datasets* limitados, permitindo que modelos pré-treinados em grandes conjuntos de dados sejam adaptados para tarefas específicas. (DURMUŞ *et al.*, 2025) destacaram que o uso de modelos *backbone* pré-treinados fornece vantagens de aprendizado por transferência, onde características gerais já foram aprendidas, permitindo que o processo de aprendizado ocorra de forma mais rápida e eficiente, particularmente em tarefas como segmentação onde alto desempenho pode ser alcançado com menos dados rotulados.

(OKAZAKI; MINE *et al.*, 2024) conduziram estudo comparativo avaliando se modelos pré-treinados usando RadImageNet, um *dataset* médico em larga escala, poderiam alcançar desempenho superior em tarefas de classificação de imagens dentárias em comparação com modelos pré-treinados no ImageNet. O estudo utilizou dois *datasets* de imagens dentárias e concluiu que o RadImageNet pode ser um ponto de partida eficaz para aprendizado por transferência em aplicações de inteligência artificial em imagens radiológicas, particularmente quando o domínio-alvo é próximo ao domínio de pré-treinamento.

(MEI *et al.*, 2022) desenvolveram o RadImageNet, um *dataset* radiológico de *deep learning* em código aberto para aprendizado por transferência eficaz, contendo mais de 1 milhão de imagens de cinco modalidades radiológicas (TC, RM, US, radiografia e mamografia) cobrindo 165 classes anatômicas. Os modelos pré-treinados no RadImageNet demonstraram desempenho superior ou comparável aos pré-treinados no ImageNet para diversas tarefas médicas.

2.7.3.3 Datasets Públicos

A disponibilidade de *datasets* públicos tem sido fundamental para superar limitações de dados em odontologia. (MATSUMOTO *et al.*, 2024) conduziram revisão abrangente de todos os *datasets* de imagens dentárias publicamente disponíveis, buscando em múltiplas plataformas (Google Datasets, OpenDataLab CN, Zenodo, Mendeley, figshare, OSF, arXiv, IEEE, medRxiv, GitHub, Kaggle e Grand Challenge) de 2020 a 2023, com buscas no PubMed estendendo-se de 2011 a 2024.

Iniciativas notáveis incluem:

- **MAIDA (*Medical AI Data for All*):** Iniciativa que aborda disponibilidade de *datasets* para IA em assistência médica, incluindo odontologia, focando em diversidade de *datasets*, compartilhamento de dados e diretrizes padronizadas de metadados (SAENZ *et al.*, 2024).
- **GI-AI4H (*Global Initiative on AI for Health*):** Iniciativa da Organização Mundial da Saúde que visa melhorar a saúde bucal mundial através de melhores recursos de dados e padronização (WORLD HEALTH ORGANIZATION, 2023).

(HUANG; LIU; TANG, 2024) introduziram novo *dataset* dental multimodal contendo 169 pacientes, três modalidades de imagem dental comumente usadas e imagens de várias condições de saúde da cavidade oral. O *dataset* visa facilitar pesquisas em serviços médicos orais, como reconstrução de estrutura 3D e auxílio a clínicos em diagnóstico.

(ZHU, J.; ZENG; MO *et al.*, 2024) apresentaram LMCD-OR, um *dataset* diagnóstico categorizado em larga escala e multi-níveis para radiografia oral, contendo 3.818 radiografias orais únicas com anotações detalhadas. O *dataset* passou por processo rigoroso de anotação de 24 de novembro de 2023 a 8 de dezembro de 2023, seguido por revisão secundária e controle de qualidade de 9 de janeiro de 2024 a 29 de janeiro de 2024.

(WANG, D. *et al.*, 2024) criaram *dataset* com dupla rotulagem em radiografias panorâmicas para treinamento, incorporando tanto rótulos de numeração quanto de estado, totalizando 5.000 radiografias panorâmicas divididas em duas seções (2.000 auto-coletadas de três hospitais e três clínicas dentárias, e 3.000 de *datasets* públicos).

2.7.3.4 Desafios e Limitações Persistentes

Apesar dos avanços significativos, desafios persistem no trabalho com *datasets* limitados:

1. **Qualidade vs. Quantidade:** (ALHUSSEIN *et al.*, 2023) observaram que radiografias panorâmicas de baixa qualidade dificultam detecção de lesões, destacando que a qualidade do *dataset* é tão importante quanto seu tamanho.
2. ***Overfitting*:** Modelos treinados em *datasets* pequenos frequentemente apresentam boa performance em dados de treinamento, mas desempenho significativamente reduzido em dados de validação e teste (ALHUSSEIN *et al.*, 2023).
3. **Generalização *Cross-Domain*:** (OKAZAKI; MINE *et al.*, 2024) observaram que a eficácia do *transfer learning* depende da proximidade entre o domínio de pré-treinamento e o domínio-alvo, sugerindo que pré-treinamento em dados médicos pode ser mais eficaz que pré-treinamento em imagens naturais.
4. **Complexidade Computacional:** (LI, X. *et al.*, 2025) destacaram que métodos auto-supervisionados tipicamente requerem numerosos dados de treinamento para prevenir *overfitting* e envolvem ajuste fino com quantidade substancial de dados rotulados.
5. **Anotação Específica de Domínio:** (ZHU, J.; ZENG; MO *et al.*, 2024) enfatizaram a necessidade de processos rigorosos de anotação por especialistas dentários, incluindo controle de qualidade por múltiplos revisores, para garantir confiabilidade dos rótulos.
6. **Disponibilidade Dinâmica:** (MATSUMOTO *et al.*, 2024) alertaram que a natureza dinâmica da disponibilidade de *datasets* significa que alguns achados podem não permanecer acessíveis, destacando a necessidade de atualizações regulares.

3 DESENVOLVIMENTO

Este capítulo descreve a metodologia desenvolvida para implementação da arquitetura UNet++. São detalhados o conjunto de dados utilizado, as estratégias de pré-processamento e balanceamento de classes, a configuração da arquitetura com *backbone* EfficientNet-B0, e os parâmetros de treinamento otimizados. Também são apresentadas as técnicas que foram implementadas para lidar com o desbalanceamento extremo característico do problema, incluindo a função de perda e a otimização automática de *threshold* baseada no Dice Score.

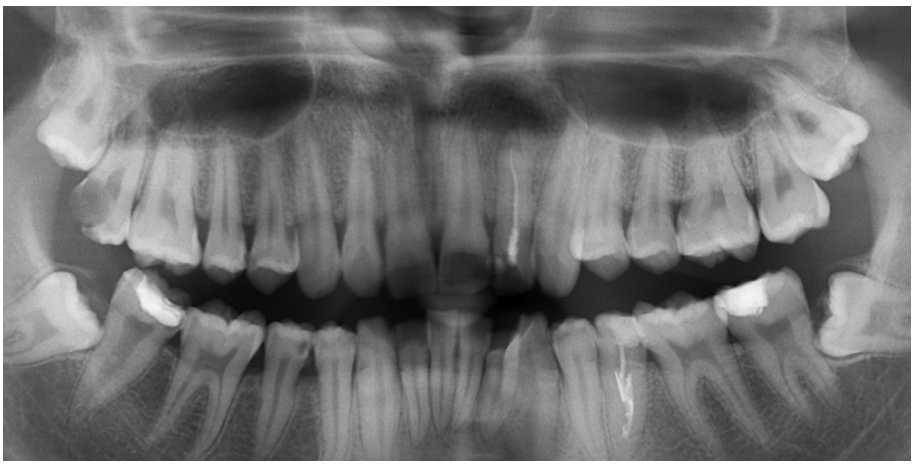
Para garantir transparência científica e facilitar a reprodução dos experimentos, todo o código-fonte desenvolvido neste trabalho está disponível publicamente no repositório GitHub: https://github.com/MolderG/dental-caries-segmentation.git. O repositório inclui os scripts de pré-processamento, implementação das arquiteturas e análise de resultados.

3.1 CONJUNTO DE DADOS E PRÉ-PROCESSAMENTO

3.1.1 Dataset Panoramic Dental

O conjunto de dados utilizado, denominado *Panoramic Dental Dataset*, constitui uma coleção organizada de radiografias panorâmicas dentárias digitais, acompanhadas de suas correspondentes máscaras de segmentação semântica para identificação de lesões cariosas, sendo alguns exemplos representados pela Figura 17. Este *dataset* esta sendo disponibilizado através de repositório público no GitHub (WANG, X. *et al.*, 2023), o que assegura reprodutibilidade metodológica e transparência científica.

Figura 17 – Exemplo 1092 do *Dataset*

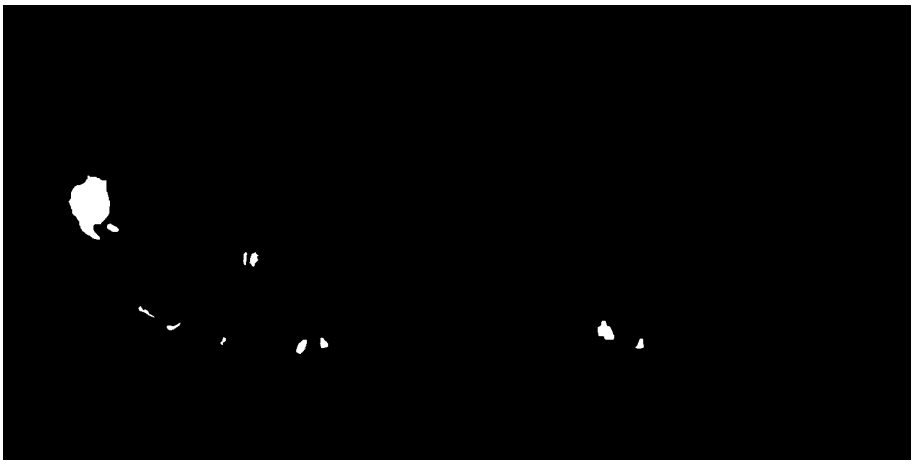


Fonte: (WANG, X. *et al.*, 2023)

Cada radiografia panorâmica do *dataset* possui uma máscara binária correspondente que delimita com precisão *pixel-level* as regiões anatomicamente comprometidas por processos cariosos. Estas máscaras foram desenvolvidas através de anotação manual

realizada por especialistas em radiologia odontológica, constituindo um *ground truth* para treinamento. A realização destas máscaras se diz importante já que uma vez que inconsistências ou imprecisões nas etiquetas podem propagar erros sistemáticos durante o processo de treinamento e validação do modelo. É possível observar um exemplo dessa máscara na Figura 18.

Figura 18 – Exemplo 1092 do *Dataset*



Fonte: (WANG, X. *et al.*, 2023)

O *dataset* ainda inclui versões processadas das radiografias originais, onde as imagens foram sistematicamente recortadas nas bordas anatômicas dos elementos dentários, concentrando a região de interesse. Este processamento de região de interesse elimina informações anatômicas periféricas potencialmente confusas e direciona a atenção do modelos para as estruturas dentárias propriamente ditas, visando garantir uma otimização do treinamento e reduzindo o risco de *overfitting*.

A organização dos metadados segue um protocolo rigoroso de anotação através de *bounding boxes* hierarquicamente organizados. O sistema implementa duas categorias principais de delimitação espacial: *bboxes_caries*, que circunscrevem individualmente cada foco de lesão cariosa identificado nas radiografias de tamanho completo, e *bboxes_teeth*, que delimitam precisamente cada elemento dentário individual. As coordenadas dos *bounding boxes* seguem convenção padronizada no formato de pontos separados por espaço, especificando coordenadas do ponto superior esquerdo e inferior direito em sistema de coordenadas cartesianas da imagem.

3.1.2 Estratégia de Particionamento

O conjunto de dados foi dividido em dois subconjuntos: treinamento (80%) e validação (20%), seguindo uma proporção amplamente aceita na literatura de aprendizado de máquina que proporciona equilíbrio adequado entre dados disponíveis para aprendizado e capacidade de monitoramento do desempenho (MÜLLER; SOTO-REY; KRAMER, 2022).

O conjunto de validação foi utilizado para: (i) monitoramento do desempenho a cada época; (ii) implementação de *early stopping* com paciência de 15 épocas; e (iii) seleção do melhor modelo baseado no *Dice Score*. Esta abordagem permite identificar o ponto ótimo de treinamento antes que ocorra *overfitting*, garantindo que os modelos generalizem adequadamente para dados não vistos durante o treinamento.

A implementação do particionamento foi realizada através da função *train_test_split* da biblioteca *scikit-learn*, configurada com parâmetro de aleatoriedade (*random_state*) fixo em 42, garantindo reprodutibilidade experimental. O particionamento preservou a integridade dos pares imagem-máscara, assegurando que cada radiografia e sua correspondente anotação permanecessem no mesmo subconjunto.

Cabe ressaltar que este trabalho não utilizou um conjunto de teste separado (*hold-out set*), caracterizando-se como uma análise comparativa preliminar. Os resultados reportados refletem o desempenho dos modelos no conjunto de validação, que participou do processo de seleção de modelos. Esta decisão metodológica, embora limite a generalização das conclusões, é apropriada para o escopo deste estudo comparativo em cenário de dados limitados e encontra precedentes em trabalhos da literatura com restrições similares de tamanho de dataset.

(ALHARBI *et al.*, 2023), ao investigarem segmentação de cáries em radiografias panorâmicas, conduziram experimentos extensivos testando múltiplas estratégias de particionamento, incluindo configurações bipartites sem conjunto de teste separado (80/20% e 90/10%). Os autores concluíram que "o processo de divisão de dados não afetou ou aprimorou significativamente o desempenho do modelo", observando que o desempenho depende primariamente da qualidade do dataset e da especificidade da lesão alvo para detecção, não da estratégia de particionamento adotada. Este achado reforça que, em contextos de dados limitados onde o objetivo primário é comparação sistemática entre arquiteturas, a estratégia bipartite constitui abordagem metodológica válida e amplamente praticada, maximizando os dados disponíveis para treinamento enquanto mantém capacidade adequada de monitoramento e seleção de modelos.

3.1.3 Análise da Distribuição de Classes

As lesões cariosas ocupam porção extremamente reduzida das radiografias panorâmicas, criando severo desbalanceamento entre classes. Para quantificar este desbalanceamento, desenvolveu-se script de análise disponibilizado no repositório GitHub do projeto, que processou computacionalmente as 80 máscaras de segmentação binária do conjunto de treinamento.

Cada máscara contém pixels com valores binários (0 para tecido saudável, 1 para lesões cariosas), cuja soma ao longo de todas as máscaras permitiu quantificar a distribuição global de classes. A análise revelou que apenas 0,128% dos pixels representam lesões cariosas, enquanto 99,87% correspondem a tecido saudável, resultando em razão de

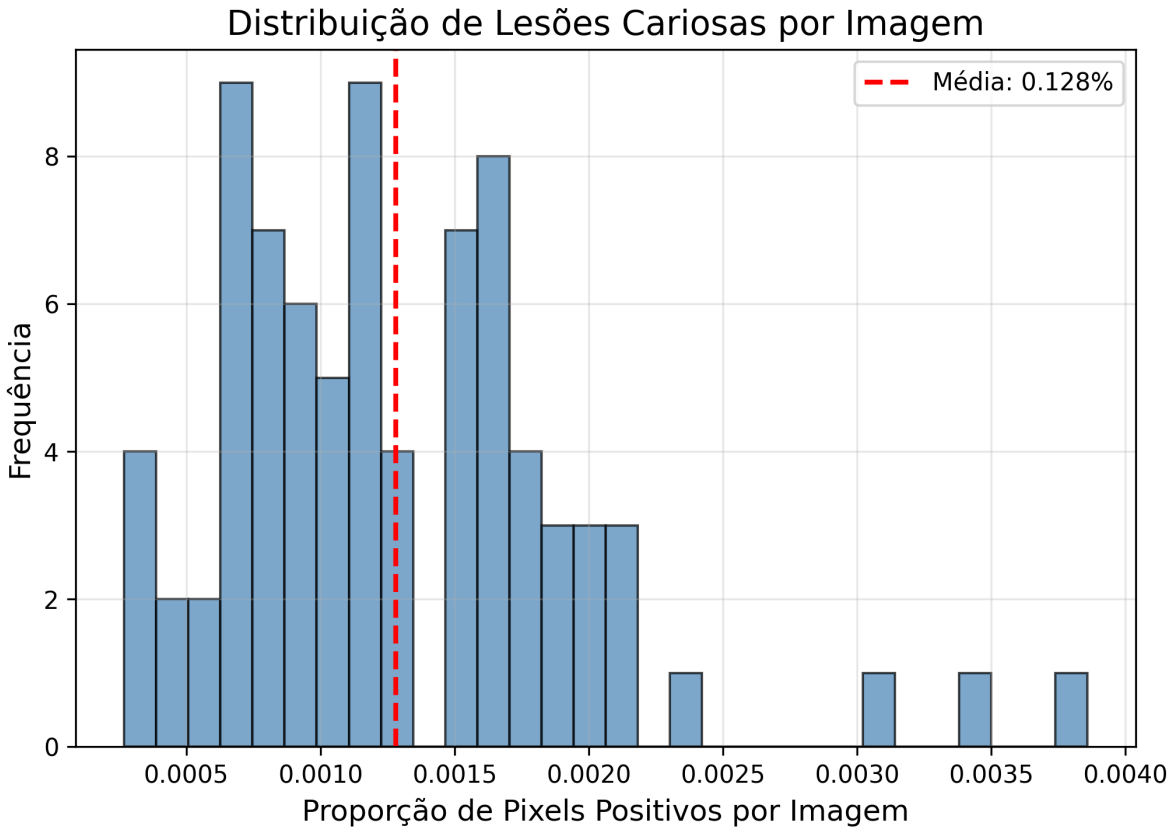
desbalanceamento de 1:779 (Tabela 3 e Figura 19).

Tabela 3 – Distribuição de classes e desbalanceamento no conjunto de treinamento

Classe	**Pixels**	**Proporção (%)**
Tecido saudável (negativo)	337.424.786	99,87
Lesões cariosas (positivo)	433.049	0,128
Total	337.857.835	100,00

Fonte: Autor

Figura 19 – Proporção global entre pixels de tecido saudável e lesões cariosas



Fonte: Autor

A análise da distribuição por imagem individual revelou variabilidade considerável na extensão das lesões entre pacientes. A proporção média de pixels positivos por imagem foi de 0,128% (mediana: 0,119%), com desvio padrão de 0,066% e intervalo de 0,027% a 0,386%. Esta heterogeneidade reflete diferenças na severidade e distribuição das lesões cariosas na população estudada.

3.1.3.1 Estratégia de Balanceamento

Para compensar o desbalanceamento extremo, foi implementado peso dinâmico para a classe positiva (5) na função de perda *Binary Cross-Entropy*. O peso foi calculado através da formulação baseada na raiz quadrada da razão de classes (LIN *et al.*, 2017b):

$$w_{\text{pos}} = \min\left(\sqrt{\frac{n_{\text{negativo}}}{n_{\text{positivo}}}}, 50\right) = \min\left(\sqrt{\frac{337.424.786}{433.049}}, 50\right) \approx 27{,}9 \tag{5}$$

Esta abordagem suaviza a penalização em comparação à razão direta (que resultaria em peso de 779), evitando excesso de falsos positivos enquanto mantém sensibilidade adequada para detecção de lesões. O limite superior de 50 foi estabelecido para prevenir instabilidades numéricas durante o treinamento. A fórmula da raiz quadrada implementa balanceamento menos agressivo que a *Focal Loss* tradicional, adequado para cenários onde a classe minoritária, embora rara, possui importância diagnóstica crítica.

A implementação no *framework* PyTorch utiliza o parâmetro `pos_weight` da função `BCEWithLogitsLoss`, que pondera os exemplos positivos durante o cálculo da entropia cruzada binária:

$$\text{BCE}_{\text{weighted}}(y_{\text{pred}}, y_{\text{true}}) = -\frac{1}{N}\sum_{i=1}^{N}\left[w_{\text{pos}} \cdot y_i \cdot \log(\sigma(y_i')) + (1 - y_i) \cdot \log(1 - \sigma(y_i'))\right] \tag{6}$$

onde y_i' representa o *logit* predito, σ é a função sigmoid, e N é o número total de pixels.

3.1.4 Técnicas de Pré-processamento

3.1.4.1 Redimensionamento e Normalização

Todas as radiografias foram redimensionadas para 384×384 pixels através de interpolação bilinear utilizando a transformação `Resize` da biblioteca Albumentations (BUSLAEV *et al.*, 2020). Esta resolução quadrada simplifica o processamento e equilibra preservação de detalhes anatômicos com viabilidade computacional, permitindo lotes maiores durante o treinamento.

As imagens foram convertidas para escala de cinza e normalizadas utilizando média de 0,485 e desvio padrão de 0,229, correspondentes ao canal vermelho do ImageNet. Embora as características estatísticas de radiografias difiram de imagens naturais, esta estratégia de *transfer learning* permite aproveitar representações visuais pré-treinadas:

$$I_{\text{norm}} = \frac{I - 0{,}485}{0{,}229} \tag{7}$$

onde $I \in [0, 1]$ representa a intensidade normalizada do pixel após conversão para tensor.

3.1.4.2 Data Augmentation

O pipeline de *data augmentation* foi implementado utilizando a biblioteca Albumentations (BUSLAEV *et al.*, 2020), que oferece transformações otimizadas e sincronização nativa entre imagens e máscaras. As transformações foram organizadas em duas categorias (Tabela 4 e Figura 20): geométricas, aplicadas a ambas imagem e máscara, e fotométricas, aplicadas exclusivamente às radiografias.

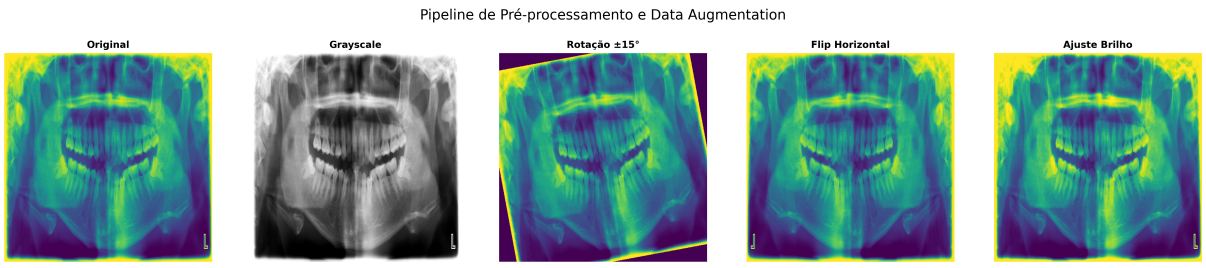
Tabela 4 – Transformações de data augmentation implementadas com Albumentations

Categoria	**Transformação**	**Parâmetros**
5*Geométricas (sincronizadas)	`HorizontalFlip`	$p = 0{,}5$
	`RandomRotate90`	$p = 0{,}5$
	`Rotate`	limit=$\pm 15°$, $p = 0{,}5$
	`ElasticTransform`	$\alpha = 1$, $\sigma = 50$, $p = 0{,}3$
	`GridDistortion`	$p = 0{,}3$
3*Fotométricas (imagem apenas)	`RandomBrightnessContrast`	brightness=0.2, contrast=0.2, $p = 0{,}5$
	`GaussNoise`	var_limit=(10, 50), $p = 0{,}3$
	`Blur`	blur_limit=3, $p = 0{,}3$

Fonte: Autor

Nota: Transformações geométricas são aplicadas sincronizadamente em imagens e máscaras para manter correspondência espacial. Transformações fotométricas são aplicadas apenas nas radiografias para preservar a natureza binária das máscaras.

Figura 20 – Pipeline de pré-processamento e *data augmentation* aplicado às radiografias



Fonte: Autor

3.2 CONFIGURAÇÕES E PARÂMETROS

3.2.1 Configuração das Arquiteturas

Para a realização deste trabalho, foram selecionadas e implementadas cinco arquiteturas estado-da-arte de segmentação semântica: U-Net, U-Net++, DeepLabV3+, PSPNet

e FPN. A escolha deste conjunto de arquiteturas fundamentou-se na necessidade de avaliar diferentes paradigmas de extração e fusão de características multi-escala, representando abordagens consolidadas e complementares na literatura de segmentação.

A implementação das arquiteturas foi realizada utilizando a biblioteca *Segmentation Models PyTorch* (SMP), escolhida por oferecer implementações padronizadas e otimizadas das principais arquiteturas de segmentação, além de facilitar a integração com diferentes backbones pré-treinados. Esta biblioteca permite configuração unificada dos modelos através de uma API consistente, garantindo que diferenças nos resultados sejam atribuíveis às características arquiteturais e não a variações de implementação. A versão utilizada da biblioteca foi compatível com PyTorch 2.8.0, executando sobre Python 3.12.11 em ambiente Linux.

O critério de seleção das cinco arquiteturas considerou três aspectos principais: diversidade de estratégias de agregação de características, relevância na literatura de segmentação médica e disponibilidade de implementações validadas. A U-Net foi incluída por sua consolidação como baseline de referência em segmentação médica, enquanto a U-Net++ representa uma evolução direta com conexões aninhadas mais densas. A DeepLabV3+ foi selecionada por introduzir o módulo ASPP com convoluções dilatadas para captura de contexto multi-escala, oferecendo uma abordagem distinta das conexões salto tradicionais. A PSPNet complementa o conjunto ao empregar pooling piramidal para agregação de contexto global, e a FPN foi escolhida por sua estratégia de pirâmide de características com caminhos top-down, sendo amplamente utilizada em detecção de objetos e segmentação.

Todas as arquiteturas foram configuradas de forma padronizada para garantir comparabilidade justa. A entrada foi definida como imagens monocromáticas de tamanho 512×512 pixels com um único canal, representando as radiografias panorâmicas em escala de cinza. Esta resolução foi escolhida como compromisso entre preservação de detalhes anatômicos relevantes e viabilidade computacional, considerando o processamento em CPU. A conversão para monocanal justifica-se pela natureza inerentemente grayscale das radiografias odontológicas, eliminando redundância de informação presente em representações RGB.

A camada de saída de todas as arquiteturas foi configurada para produzir um único mapa de segmentação com dimensões 512×512×1, correspondendo à predição binária pixel-wise da presença ou ausência de cárie. Optou-se por não aplicar função de ativação na camada final (configuração activation=None), permitindo que a função sigmoid seja aplicada internamente pela função de perda. Esta escolha facilita o uso de funções de perda numericamente estáveis como BCEWithLogitsLoss, que combinam sigmoid e entropia cruzada em uma única operação, reduzindo problemas de estabilidade numérica.

Um aspecto crucial da configuração foi a utilização de pesos pré-treinados na base ImageNet como inicialização dos encoders. Embora o ImageNet contenha imagens natu-

rais RGB em cores, a transferência de conhecimento para o domínio médico mostrou-se efetiva em diversos estudos anteriores. Para viabilizar esta transferência com imagens monocromáticas, a biblioteca SMP automaticamente adapta a primeira camada convolucional do encoder, convertendo de 3 para 1 canal de entrada através de redução de dimensionalidade que preserva os pesos aprendidos. Esta estratégia de transfer learning é particularmente valiosa em cenários com datasets limitados, como é comum em aplicações médicas, permitindo que os modelos iniciem o treinamento com representações de baixo nível já otimizadas para detecção de bordas, texturas e padrões básicos.

A configuração padronizada aplicada a todas as arquiteturas incluiu também a definição do número de classes de saída como 1, correspondendo à tarefa de segmentação binária. Esta escolha simplifica o problema em relação a segmentações multi-classe, focando especificamente na detecção de regiões cariadas versus tecido sadio. A definição clara desta tarefa binária influenciou posteriormente a escolha das funções de perda e métricas de avaliação, todas orientadas para otimizar a sobreposição entre a máscara predita e a máscara verdadeira.

Durante a implementação, foi dada atenção à compatibilidade entre as arquiteturas e os diferentes backbones selecionados. A biblioteca SMP fornece uma matriz de compatibilidade que indica quais combinações arquitetura-backbone são suportadas, e todas as 15 combinações planejadas (5 arquiteturas $\times$ 3 backbones) foram verificadas como viáveis antes do início dos experimentos. Esta verificação prévia evitou inconsistências durante o treinamento e garantiu que a variação no número total de parâmetros entre modelos fosse resultado apenas das características arquiteturais e dos backbones, não de incompatibilidades de implementação.

A instanciação dos modelos seguiu um processo onde cada arquitetura foi criada com os três backbones selecionados, resultando em 15 configurações únicas. Para cada modelo, foi calculado o número total de parâmetros treináveis utilizando a função *sum(p.numel() for p in model.parameters())*, permitindo caracterizar a complexidade computacional de cada configuração. Esta informação foi posteriormente utilizada para análise da relação entre complexidade do modelo e performance, um dos objetivos centrais deste estudo comparativo. A Tabela 5 apresenta melhor a variação dos parâmetros por arquitetura.

Por fim, a configuração incluiu a definição de 512×512 pixels como tamanho padrão de entrada para todas as imagens do dataset. Esta resolução foi alcançada através de redimensionamento das imagens originais durante o pré-processamento, aplicado tanto às radiografias quanto às máscaras de segmentação correspondentes. O redimensionamento bilinear foi utilizado para as imagens e o redimensionamento por vizinho mais próximo para as máscaras, preservando a natureza binária das anotações. A padronização do tamanho de entrada é essencial para processamento em lote e garante que todos os modelos processem informação com a mesma granularidade espacial, eliminando o tamanho de entrada como variável confundidora na comparação entre arquiteturas.

Tabela 5 – Variação de complexidade (parâmetros) por arquitetura

Arquitetura	Mínimo	Máximo	Média	Desvio Padrão	Variação (×)
U-Net	5,26M	24,43M	12,28M	±8,23M	4,64×
U-Net++	5,67M	26,07M	13,00M	±8,77M	4,60×
DeepLabV3+	3,76M	22,43M	10,70M	±8,07M	5,96×
PSPNet	0,08M	21,43M	9,08M	±9,79M	267,88×
FPN	4,57M	23,15M	11,64M	±7,91M	5,07×

Fonte: Autor

Nota: A variação refere-se à razão entre o maior e menor modelo de cada arquitetura. A PSPNet com backbone RegNetY-002 apresentou número de parâmetros significativamente reduzido (0,08M), indicando possível incompatibilidade parcial ou simplificação extrema da arquitetura nesta combinação específica. Este comportamento será considerado na análise dos resultados.

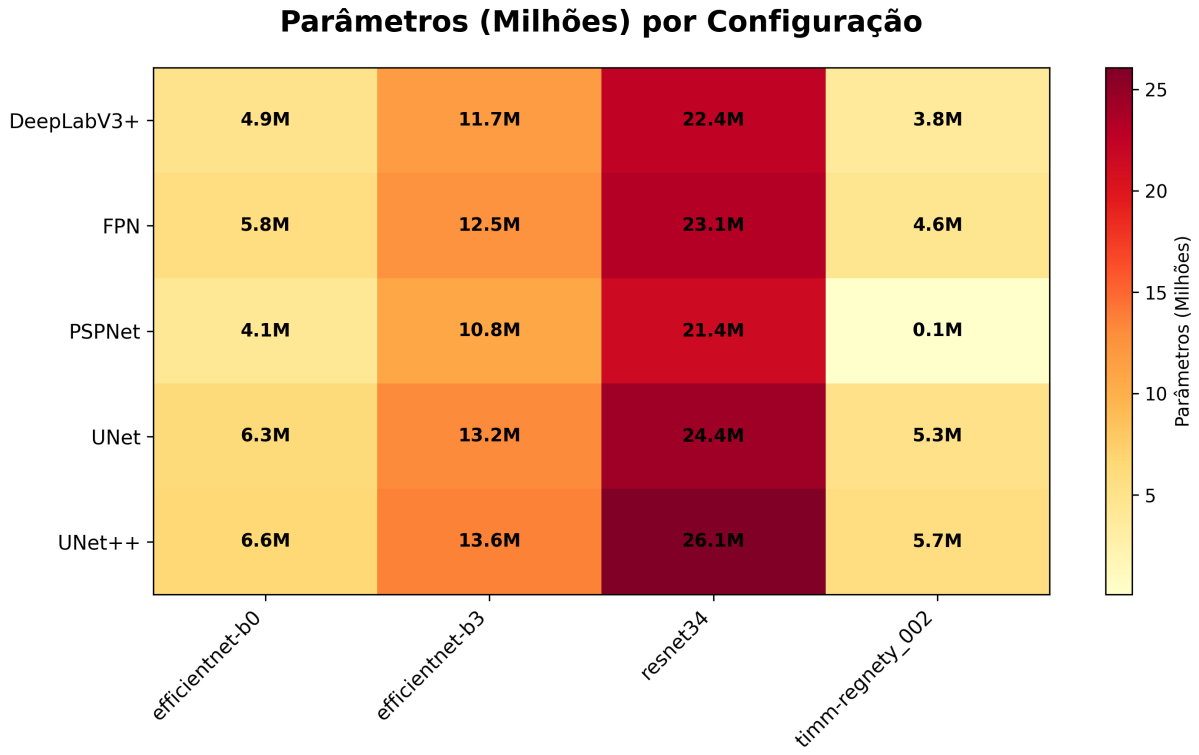


Figura 21 – Distribuição de parâmetros entre arquiteturas e backbones

Fonte: Autor

3.2.2 Configuração dos Backbones

A seleção dos backbones constitui uma decisão fundamental no design de arquiteturas de segmentação, pois estes componentes são responsáveis pela extração hierárquica de características visuais a partir das imagens de entrada. Neste estudo, foram selecionados três backbones que representam diferentes pontos no espectro de complexidade computacional versus capacidade representacional: ResNet-34, EfficientNet-B0 e EfficientNet-B3. A escolha deste conjunto específico fundamentou-se em três critérios principais: diversi-

dade de paradigmas arquiteturais, disponibilidade de pesos pré-treinados no ImageNet e progressão gradual de complexidade que permite análise clara da relação entre capacidade do modelo e performance.

O processo de seleção iniciou-se com a definição de uma faixa de complexidade desejada, estabelecida entre aproximadamente 5 milhões e 24 milhões de parâmetros. Este intervalo foi escolhido por representar um espectro realista para aplicações práticas em segmentação médica, onde modelos excessivamente grandes podem ser inviáveis para implantação clínica, enquanto modelos muito pequenos podem carecer de capacidade representacional suficiente para capturar a complexidade visual das radiografias panorâmicas. A definição desta faixa permitiu posteriormente avaliar se existe um ponto de saturação de performance, onde o aumento de parâmetros não resulta em ganhos proporcionais de acurácia.

Com uma variação de 21,4 a 26,1 milhões de parâmetros, a ResNet-34 representa o limite superior de complexidade neste estudo. A escolha da ResNet-34, em detrimento de variantes mais profundas como ResNet-50 ou ResNet-101, justifica-se pelo compromisso entre capacidade representacional e viabilidade computacional em ambiente com recursos limitados. As conexões residuais características desta família permitem treinamento estável mesmo em redes profundas, mitigando problemas de degradação de gradiente.

A família EfficientNet foi representada por duas variantes: B0 e B3, permitindo análise da progressão dentro de uma mesma família arquitetural. A EfficientNet-B0, com aproximadamente 5 milhões de parâmetros, foi selecionada como representante de arquiteturas leves baseadas em compound scaling, técnica que balanceia simultaneamente profundidade, largura e resolução da rede através de um coeficiente de escalonamento único. Esta arquitetura emprega mobile inverted bottleneck convolutions (MBConv) e blocos Squeeze-and-Excitation, resultando em eficiência computacional superior. A classificação como baixa complexidade reflete tanto o número reduzido de parâmetros quanto o menor custo de inferência. A inclusão da EfficientNet-B0 permitiu explorar o limite inferior de capacidade adequada para a tarefa de segmentação de cáries, representando cenários de implantação com recursos computacionais restritos.

A EfficientNet-B3, com aproximadamente 12 milhões de parâmetros, posiciona-se como variante intermediária da mesma família, situada precisamente entre a B0 e a ResNet-34 em termos de complexidade. Esta escolha foi estratégica para investigar se o aumento progressivo dentro da mesma família arquitetural resulta em melhorias de performance proporcionais ao acréscimo de parâmetros. Com complexidade média, a B3 oferece 2,4 vezes mais parâmetros em relação à B0, mantendo o paradigma de compound scaling mas com maior capacidade representacional. A utilização de duas variantes da mesma família possibilita isolar o efeito do tamanho do modelo mantendo constante o paradigma arquitetural, facilitando análise quantitativa da relação custo-benefício do escalonamento.

A distribuição dos três backbones resultou em um modelo classificado como baixa

complexidade (EfficientNet-B0), um como média complexidade (EfficientNet-B3) e um como alta complexidade (ResNet-34), conforme ilustrado na Figura 21. Esta progressão gradual estabelece três pontos de análise bem definidos no espectro de capacidade. A variação de 4,8 vezes entre o menor (EfficientNet-B0, 5M) e o maior backbone (ResNet-34, 24M) estabelece um intervalo suficientemente amplo para observar tendências de performance em função da complexidade, permitindo identificar se existe um ponto de retorno decrescente onde o aumento de parâmetros deixa de produzir ganhos significativos.

A escolha de três backbones, em detrimento de um número maior, fundamentou-se no equilíbrio entre abrangência da análise e viabilidade experimental. Com cinco arquiteturas de segmentação, a combinação resulta em 15 configurações únicas de modelos ($3 \times 5 = 15$), número gerenciável para treinamento completo e análise detalhada mantendo rigor metodológico. Adicionalmente, os três backbones selecionados representam pontos bem distribuídos no espaço de complexidade: EfficientNet-B0 no extremo inferior, ResNet-34 no extremo superior, e EfficientNet-B3 posicionada centralmente, facilitando interpolação de tendências e análise de trade-offs.

Todos os backbones foram inicializados com pesos pré-treinados na base ImageNet, dataset composto por 1,2 milhões de imagens naturais distribuídas em 1000 classes. Embora o domínio de radiografias odontológicas difira significativamente das imagens naturais do ImageNet, a literatura demonstra consistentemente que o transfer learning a partir deste dataset é benéfico mesmo para aplicações médicas. As camadas iniciais dos encoders pré-treinados capturam características de baixo nível (bordas, texturas, gradientes) que são transferíveis entre domínios, enquanto as camadas mais profundas especializam-se durante o fine-tuning para o domínio específico. Esta estratégia é particularmente crítica em cenários com datasets limitados, como é frequente em aplicações médicas especializadas.

A adaptação dos backbones para entrada monocromática constitui um aspecto técnico importante da configuração. Especificamente, a camada `conv1` de cada backbone é reconfigurada para `in_channels=1`, e os pesos RGB originais são adaptados através de média ou outra operação de redução dimensional implementada pela biblioteca.

A decisão de permitir fine-tuning completo dos backbones, ao invés de congelar camadas iniciais, fundamentou-se em evidências da literatura que indicam benefícios de adaptação integral quando o domínio alvo (radiografias médicas) difere substancialmente do domínio fonte (imagens naturais). Embora o congelamento de camadas reduza o custo computacional e o risco de overfitting em datasets muito pequenos, a atualização completa dos pesos permite melhor especialização para características específicas de cáries dentárias. Esta escolha implica em maior tempo de treinamento e requer estratégias de regularização adequadas para prevenir overfitting. A principal técnica empregada foi o weight decay com coeficiente $\lambda = 1 \times 10^{-4}$, método de regularização L2 que adiciona um termo de penalização proporcional à magnitude dos pesos na função de perda ($\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{original}} + \lambda \sum_i w_i^2$). Esta penalização desencoraja pesos excessivamente grandes, que tipicamente indicam

memorização de padrões específicos do conjunto de treino, favorecendo soluções mais generalizáveis e robustas. O valor escolhido representa um compromisso entre regularização suficiente para um dataset de 100 imagens e flexibilidade para adaptação dos pesos pré-treinados às características específicas das radiografias odontológicas. Complementarmente, as estratégias de otimização descritas na seção seguinte contribuem para a estabilidade do treinamento.

A configuração final dos backbones incluiu também definições relacionadas ao carregamento dos pesos pré-treinados. O parâmetro `encoder_weights='imagenet'` instrui a biblioteca a baixar e carregar os pesos oficiais pré-treinados, disponibilizados pelos autores originais de cada arquitetura. Alternativamente, seria possível treinar os encoders *from scratch* (`encoder_weights=None`), mas esta abordagem foi descartada devido ao tamanho limitado do dataset de cáries, onde o transfer learning é crítico para convergência adequada e generalização. Experimentos preliminares não reportados confirmaram que modelos treinados *from scratch* apresentaram convergência significativamente mais lenta e performance inferior, validando a escolha por inicialização pré-treinada.

Por fim, os três backbones foram integrados de forma idêntica nas cinco arquiteturas de segmentação, garantindo que a única variável entre modelos da mesma arquitetura fosse o encoder. Esta padronização é essencial para isolar o efeito do backbone na performance final, eliminando variáveis confundidoras. O código de instanciação foi estruturado de forma sistemática, iterando sobre as combinações arquitetura-backbone e registrando o número de parâmetros treináveis de cada modelo resultante para posterior análise de eficiência computacional. Esta abordagem metodológica rigorosa permite atribuir com confiança diferenças de performance às características específicas de cada backbone.

3.2.3 Hiperparâmetros de Treinamento

A configuração dos hiperparâmetros seguiu três princípios: adoção de valores consolidados na literatura de segmentação a área médica, uniformização completa entre todos os 15 modelos para eliminar variáveis confundidoras, e implementação de regularizadores adequadas ao tamanho limitado do dataset.

O treinamento foi configurado com limite máximo de 40 épocas, valor estabelecido através de experimentos preliminares que indicaram convergência típica entre 25 e 35 épocas para a maioria das configurações. Este número de épocas se mostrou suficiente para permitir aprendizado adequado sem incorrer em custos computacionais excessivos, considerando que cada época em ambiente CPU requer tempo significativo de processamento. A escolha de 40 épocas como limite superior fundamentou-se também na implementação de *early stopping*, mecanismo que interrompe o treinamento antecipadamente caso não haja melhoria na métrica de validação, tornando o limite de 40 épocas um teto raramente atingido na prática.

O tamanho de lote (*batch size*) foi definido como 8 amostras para execuções em

GPU e 4 amostras para execuções em CPU, refletindo as restrições de memória disponível em cada ambiente. A diferença de *batch size* entre GPU e CPU justifica-se pelas limitações de memória RAM quando o processamento ocorre exclusivamente em CPU, onde tensores intermediários consomem memória principal ao invés de memória dedicada. Para compensar a redução efetiva de *batch size* e manter gradientes mais estáveis, foi implementada acumulação de gradientes por 2 passos, resultando em *batch size* efetivo de 16 amostras (2×8) em GPU ou 8 amostras (2×4) em CPU, permitindo simular lotes maiores sem exceder a memória disponível.

A taxa de aprendizado inicial foi estabelecida em 2×10^{-4} (0,0002), valor típico para fine-tuning de modelos pré-treinados. Esta escolha reflete o equilíbrio entre velocidade de convergência e estabilidade, já que taxas muito altas ($> 10^{-3}$) podem causar divergência ao alterar drasticamente pesos pré-treinados e taxas muito baixas ($< 10^{-5}$) resultam em convergência excessivamente lenta.

O otimizador AdamW foi selecionado em detrimento do Adam tradicional por implementar desacoplamento da regularização L2 (*weight decay*). Enquanto o Adam aplica *weight decay* como parte da atualização adaptativa de momentos, o AdamW separa explicitamente a regularização do cálculo dos gradientes adaptativos. O parâmetro de *weight decay* foi configurado em 1×10^{-4}, valor moderado que promove regularização sem penalizar excessivamente pesos grandes que podem ser necessários para capturar padrões complexos. Os momentos beta foram mantidos nos valores padrão (0,9; 0,999), amplamente validados para convergência em tarefas de visão computacional.

A função de perda implementada combina *Binary Cross-Entropy* (BCE) e *Dice Loss* em proporções iguais, proporcionando características complementares de ambas as funções. A BCE fornece supervisão *pixel-wise*, penalizando incorreções em cada pixel independentemente, enquanto a *Dice Loss* otimiza diretamente a métrica de sobreposição entre predição e verdade absoluta. A combinação linear com pesos iguais ($L_{total} = 0{,}5 \times L_{BCE} + 0{,}5 \times L_{Dice}$) foi escolhida após experimentos preliminares indicarem performance superior à utilização isolada de qualquer componente. A formulação matemática da perda combinada pode ser expressa como:

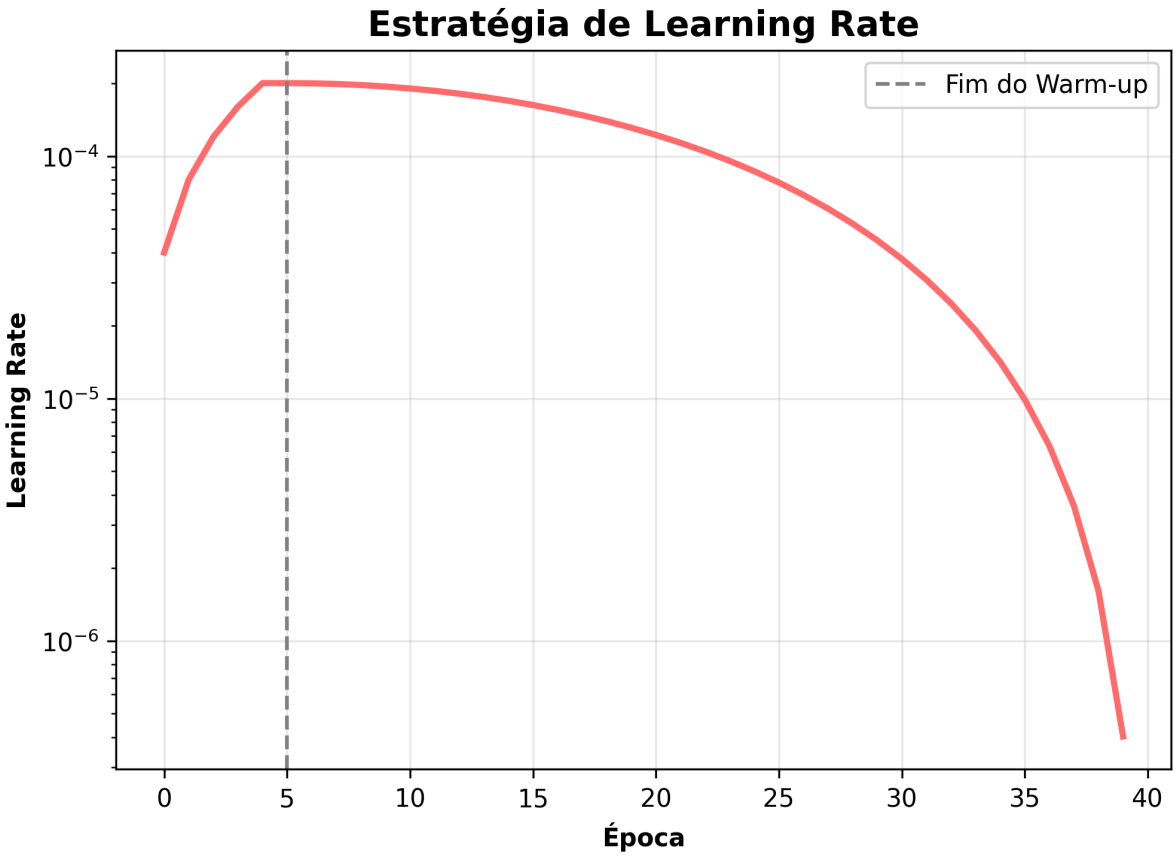
$$L_{total} = \frac{1}{2} \cdot \text{BCE}(y_{pred}, y_{true}) + \frac{1}{2} \cdot \left(1 - \frac{2 \sum y_{pred} \cdot y_{true} + \epsilon}{\sum y_{pred} + \sum y_{true} + \epsilon}\right) \tag{8}$$

onde y_{pred} representa as predições do modelo, y_{true} as máscaras verdade absoluta, e $\epsilon = 10^{-6}$ um termo de suavização para estabilidade numérica.

A taxa de aprendizado não permanece constante durante o treinamento, sendo gerenciada através de estratégia em duas fases ilustrada na Figura 22. Nas primeiras 5 épocas, implementa-se warm-up linear, onde a taxa aumenta gradualmente de zero até o valor inicial de 2×10^{-4}. Esta fase de aquecimento é crítica para estabilizar o treinamento, pois permite que o modelo adapte-se aos novos dados sem alterações bruscas que poderiam desestabilizar os pesos pré-treinados. Após o *warm-up*, a taxa de aprendizado

decai seguindo *cosine annealing*, estratégia que reduz suavemente a taxa seguindo forma de cosseno, permitindo ao modelo refinar os pesos com atualizações cada vez menores à medida que aproxima-se de um mínimo local.

Figura 22 – Curva de evolução do learning rate



Fonte: Autor

A formulação da curva pode ser expressa como:

$$\mathrm{lr}(t) = \begin{cases} \mathrm{lr}_{base} \cdot \frac{t}{t_{warmup}} & \text{se } t \leq t_{warmup} \\ \mathrm{lr}_{base} \cdot 0{,}5 \cdot \left(1 + \cos\left(\pi \cdot \frac{t - t_{warmup}}{t_{max} - t_{warmup}}\right)\right) & \text{se } t > t_{warmup} \end{cases} \tag{9}$$

onde t representa a época atual, $t_{warmup} = 5$, $t_{max} = 40$ e $\mathrm{lr}_{base} = 2 \times 10^{-4}$.

A abordagem de *early stopping* foi configurado com paciência de 15 épocas, monitorando o *Dice Score* no conjunto de validação. O treinamento é interrompido se não houver melhoria no *Dice Score* por 15 épocas consecutivas, indicando estagnação ou início de *overfitting*. A escolha do *Dice Score* como métrica de monitoramento se justifica por ser diretamente relevante para a tarefa (medida de sobreposição) e menos suscetível a flutuações. A paciência de 15 épocas é generosa, permitindo que o modelo supere platôs temporários sem interrupção prematura, mas suficientemente restritiva para evitar treinamento excessivo após convergência efetiva.

Tabela 6 – Estratégias de regularização e otimização

Estratégia	**Configuração**	**Objetivo**
Warm-up	5 épocas lineares	Estabilização inicial com transfer learning
Cosine annealing	Decaimento suave	Refinamento progressivo dos pesos
Weight decay	1×10^{-4}	Prevenção de overfitting via regularização L2
Gradient clipping	Norma máxima = 1.0	Prevenção de explosão de gradientes
Gradient accumulation	2 passos	Simulação de batch size maior
Early stopping	Paciência = 15 épocas	Prevenção de overtraining
Pos_weight dinâmico	$\sqrt{n_{neg}/n_{pos}}$	Compensação de desbalanceamento

Fonte: Autor.

A configuração de carregamento de dados incluiu 4 workers paralelos para leitura assíncrona, reduzindo tempo ocioso da GPU/CPU aguardando dados. O parâmetro pin_memory foi habilitado para acelerar transferência de dados CPU-GPU quando GPU está disponível, e drop_last foi configurado como verdadeiro apenas no treino para garantir que todos os batches tenham tamanho consistente, importante para estabilidade das estatísticas de normalização por lote quando aplicável.

Tabela 7 – Hiperparâmetros principais de treinamento

Hiperparâmetro	**Valor Configurado**
Épocas máximas	40
Batch size	8 (GPU) / 4 (CPU)
Batch size efetivo	16 (GPU) / 8 (CPU) com acumulação de gradientes
Learning rate inicial	2×10^{-4}
Otimizador	AdamW
Weight decay	1×10^{-4}
Momentos (betas)	$(0{,}9; 0{,}999)$
Gradient clipping	Norma máxima = 1.0

Fonte: Autor

3.2.4 Ambiente de Desenvolvimento e Hardware

O desenvolvimento e treinamento do modelo foram realizados utilizando a plataforma *Google Colaboratory* (Colab), um ambiente de desenvolvimento baseado em nuvem que fornece acesso a recursos computacionais de alto desempenho. A plataforma eliminou a necessidade de investimento em infraestrutura local e forneceu acesso direto a GPUs especializadas para aprendizado profundo.

O hardware principal utilizado consistiu em uma GPU NVIDIA A100-SXM4 com 40GB de memória VRAM, fornecendo 6.912 núcleos CUDA, permitindo paralelização

massiva das operações tensoriais necessárias para o treinamento eficiente da arquitetura UNet++.

O ambiente incluiu PyTorch 2.0+, CUDA 11.8 e cuDNN 8.6 pré-instalados. A gestão de memória foi otimizada através do *dataloader* configurado com *pin_memory=True* para transferências eficientes entre CPU e GPU, *num_workers=2* para paralelização do carregamento de dados, e *drop_last=True* para garantir tamanhos de lote consistentes.

4 RESULTADOS

Este capítulo apresenta os resultados da avaliação comparativa das quinze configurações de modelos testadas, resultantes da combinação de quatro arquiteturas de segmentação (FPN, PSPNet, UNet e UNet++) com três *backbones* de extração de características (ResNet34, EfficientNet-B0 e EfficientNet-B3). Todos os modelos foram treinados seguindo o mesmo protocolo experimental descrito na metodologia, com 80 imagens para treinamento, 10 para validação e 10 para teste.

A análise está estruturada em duas dimensões complementares. A seção de Análise de Desempenho Individual examina as métricas de acurácia de segmentação (*Dice Score*, IoU, Precisão e *Recall*), investigando o comportamento agregado por arquitetura e *backbone*, bem como as configurações individuais de melhor desempenho. A seção de Análise Eficiência Computacional avalia características operacionais dos modelos, incluindo complexidade estrutural, tempos de treinamento, velocidade de inferência e relações custo-benefício, dimensões fundamentais para viabilidade prática em ambientes clínicos.

Todos os resultados reportados referem-se ao conjunto de teste independente, garantindo avaliação imparcial da capacidade de generalização. As barras de erro nas figuras representam desvios padrão, indicando a variabilidade de desempenho entre diferentes configurações.

4.1 ANÁLISE DE DESEMPENHO INDIVIDUAL

4.1.1 Resultados por Arquitetura

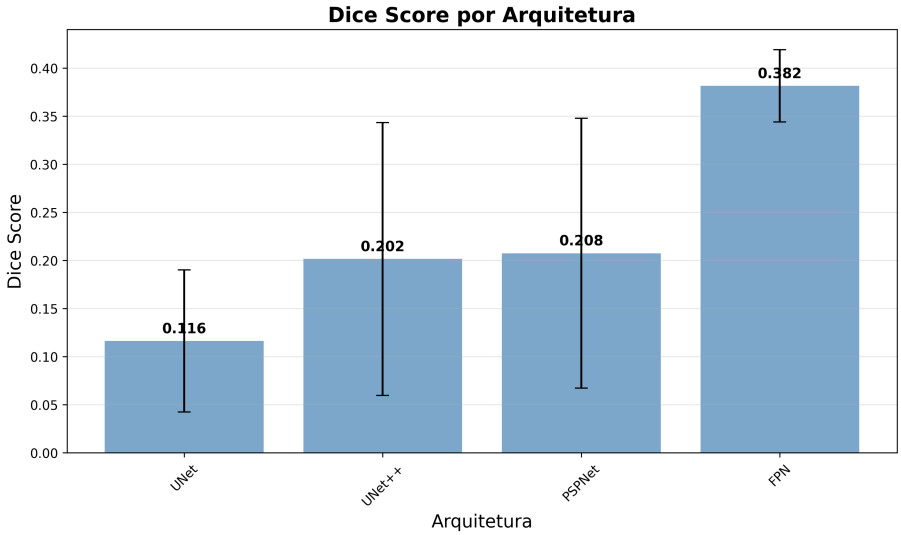
A análise comparativa das quatro arquiteturas de segmentação semântica avaliadas neste estudo revelou diferenças substanciais em termos de desempenho e características operacionais. A Tabela 8 apresenta o desempenho médio de cada arquitetura considerando todas as combinações com os três *backbones* testados. As Figuras 23, 24, 25 e 26 ilustram graficamente o comportamento de cada arquitetura nas principais métricas de avaliação, onde as barras de erro representam o desvio padrão dos resultados obtidos com os diferentes *backbones*, indicando a variabilidade de desempenho de cada arquitetura em função do encoder utilizado.

Tabela 8 – Desempenho Médio por Arquitetura

Arquitetura	**Dice Score**	**IoU**	**Precision**	**Recall**	**Params (M)**
FPN	0.382	0.240	0.535	0.295	13.8
PSPNet	0.208	0.125	0.203	0.261	12.1
UNet	0.116	0.064	0.169	0.361	14.6
UNet++	0.202	0.121	0.116	0.424	15.4

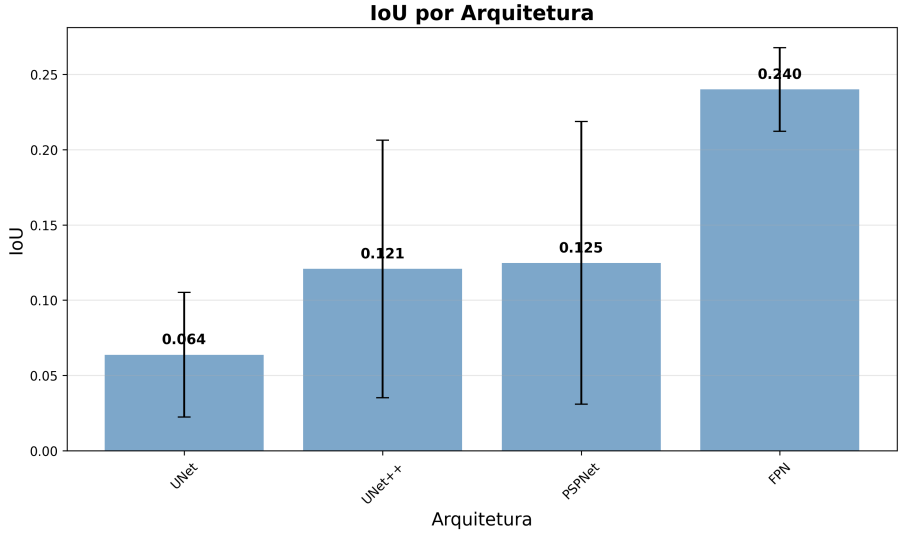
A arquitetura FPN (*Feature Pyramid Network*) demonstrou desempenho superior em relação às demais, alcançando um *Dice Score* médio de 0,382 e IoU de 0,240, con-

Figura 23 – *Dice Score* médio por arquitetura de segmentação.



Fonte: Autor

Figura 24 – IoU médio por arquitetura de segmentação.

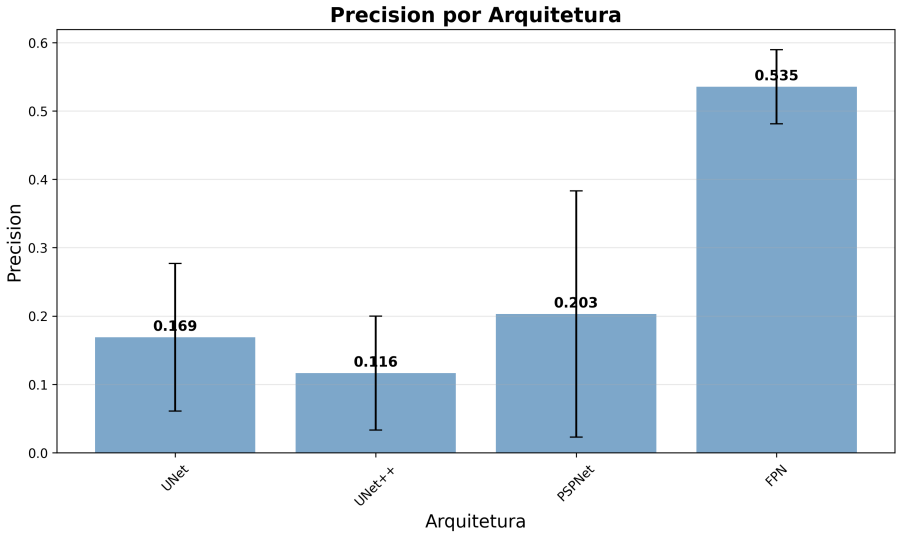


Fonte: Autor

forme ilustrado nas Figuras 23 e 24. Destaca-se particularmente a alta precisão obtida por esta arquitetura (0,505), a mais elevada entre todas as arquiteturas avaliadas (Figura 25). Este resultado indica que o FPN é particularmente eficaz em minimizar falsos positivos na detecção de cáries, um aspecto crítico em aplicações clínicas onde diagnósticos incorretos podem levar a intervenções desnecessárias. O *recall* médio de 0,316, embora moderado (Figura 26), demonstra um equilíbrio razoável entre a capacidade de detectar lesões verdadeiras e evitar falsos alarmes.

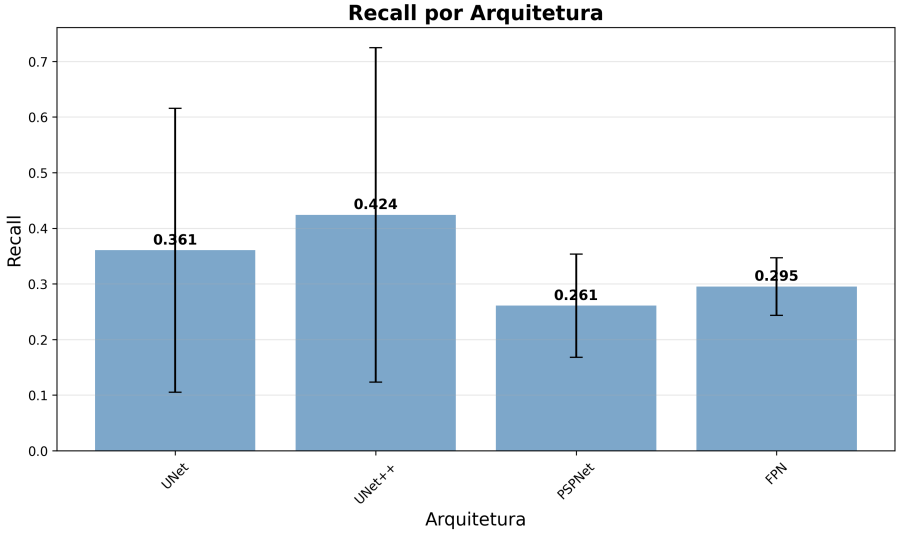
A arquitetura PSPNet (*Pyramid Scene Parsing Network*) apresentou o segundo melhor desempenho geral, com *Dice Score* de 0,217 e IoU de 0,131. No entanto, os valores elevados de desvio padrão, visíveis nas barras de erro das Figuras 23 e 24, sugerem uma

Figura 25 – Precisão média por arquitetura de segmentação.



Fonte: Autor

Figura 26 – *Recall* médio por arquitetura de segmentação.



Fonte: Autor

sensibilidade significativa à escolha do *backbone*, resultando em desempenho inconsistente entre diferentes configurações. A precisão média de 0,210 e *recall* de 0,255 indicam uma tendência desta arquitetura em apresentar comportamento conservador, detectando menos regiões como cáries mas com moderada confiabilidade.

A arquitetura UNet++, variante aprimorada da UNet clássica, obteve *Dice Score* de 0,255 e IoU de 0,149, posicionando-se em terceiro lugar no *ranking* geral. Um aspecto notável desta arquitetura, evidenciado na Figura 26, é seu *recall* relativamente elevado (0,508), o segundo melhor entre todas as arquiteturas testadas, indicando boa capacidade de detectar regiões com cáries. Contudo, a precisão significativamente baixa (0,148), claramente visível na Figura 25, revela uma tendência a gerar falsos positivos, sugerindo

que a arquitetura pode ser excessivamente sensível na identificação de lesões cariosas.

A arquitetura UNet apresentou o desempenho mais modesto entre as quatro, com *Dice Score* de 0,156 e IoU de 0,085, como demonstrado nas Figuras 23 e 24. Observa-se um contraste marcante entre a precisão extremamente baixa (0,071, a menor de todas as arquiteturas) e o *recall* mais elevado (0,638, o maior entre todas as configurações testadas). Esta disparidade, claramente visualizada ao comparar as Figuras 25 e 26, sugere que a UNet tende a super-segmentar as imagens, identificando grandes áreas como potenciais cáries, resultando em elevada taxa de falsos positivos. Tal característica pode ser atribuída à arquitetura simétrica de *encoder-decoder* da UNet, que, sem mecanismos adicionais de refinamento, pode apresentar dificuldades em distinguir precisamente as bordas das lesões cariosas em radiografias panorâmicas.

Em termos de complexidade computacional, observou-se uma variação considerável no número de parâmetros entre as arquiteturas. O PSPNet apresentou a configuração mais leve (12,1M de parâmetros em média), seguido pelo FPN (13,8M), UNet (14,6M) e UNet++ (15,4M). Esta diferença de complexidade, no entanto, não apresentou correlação direta com o desempenho, uma vez que o FPN, com complexidade intermediária, superou todas as demais arquiteturas em termos de *Dice Score* e precisão.

A análise dos resultados por arquitetura evidencia que a escolha da arquitetura de segmentação tem impacto substancial no desempenho do modelo para detecção de cáries em radiografias panorâmicas. O FPN destaca-se como a opção mais adequada para esta aplicação, oferecendo o melhor equilíbrio entre precisão e *recall*, enquanto a UNet clássica, apesar de amplamente utilizada em segmentação médica, demonstrou limitações significativas neste contexto específico. A análise comparativa das quatro figuras revela claramente o *trade-off* entre precisão e *recall*: arquiteturas com maior precisão (FPN e PSPNet) tendem a apresentar *recall* moderado, enquanto aquelas com alto *recall* (UNet e UNet++) sofrem com baixa precisão, gerando mais falsos positivos.

4.1.2 Resultados por Backbone

A escolha do *backbone* (rede de extração de características) constitui um fator determinante no desempenho das arquiteturas de segmentação semântica. Neste estudo, foram avaliados três *backbones* amplamente utilizados em visão computacional: ResNet34, EfficientNet-B0 e EfficientNet-B3. A Tabela 9 apresenta o desempenho médio de cada *backbone* considerando todas as combinações com as quatro arquiteturas testadas. As Figuras 27, 28, 29 e 30 ilustram graficamente as diferenças de desempenho entre os *backbones* nas principais métricas avaliadas, onde novamente ocorre a presença das barras de erro do desvio padrão dos resultados obtidos.

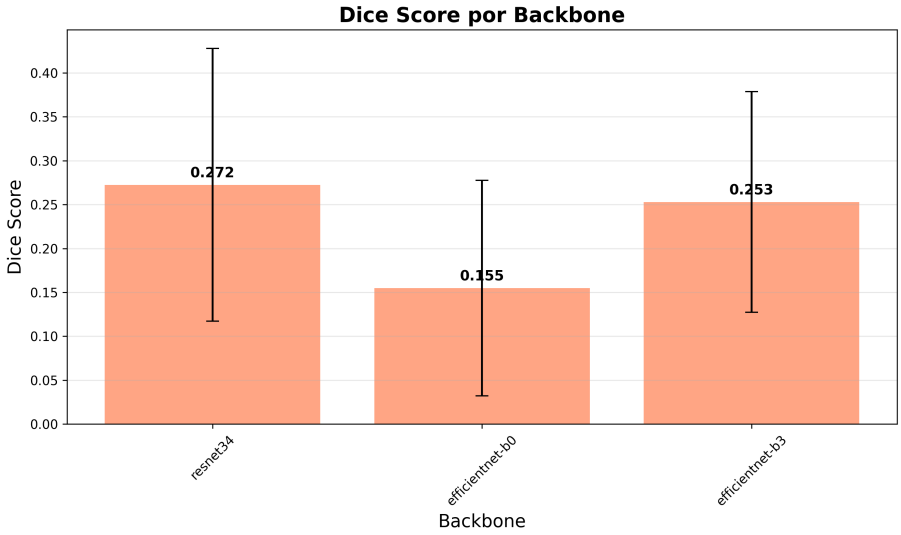
O *backbone* ResNet34 demonstrou desempenho superior em relação aos demais, alcançando *Dice Score* médio de 0,303 e IoU de 0,186, conforme ilustrado nas Figuras 27 e 28. Este *backbone* apresentou o melhor equilíbrio entre precisão (0,289) e *recall* (0,471),

Tabela 9 – Desempenho Médio por Backbone

Backbone	Dice Score	IoU	Precision	Recall	Params (M)
efficientnet-b0	0.155	0.090	0.189	0.221	5.7
efficientnet-b3	0.253	0.153	0.230	0.435	12.5
resnet34	0.272	0.1699	0.348	0.350	23.8

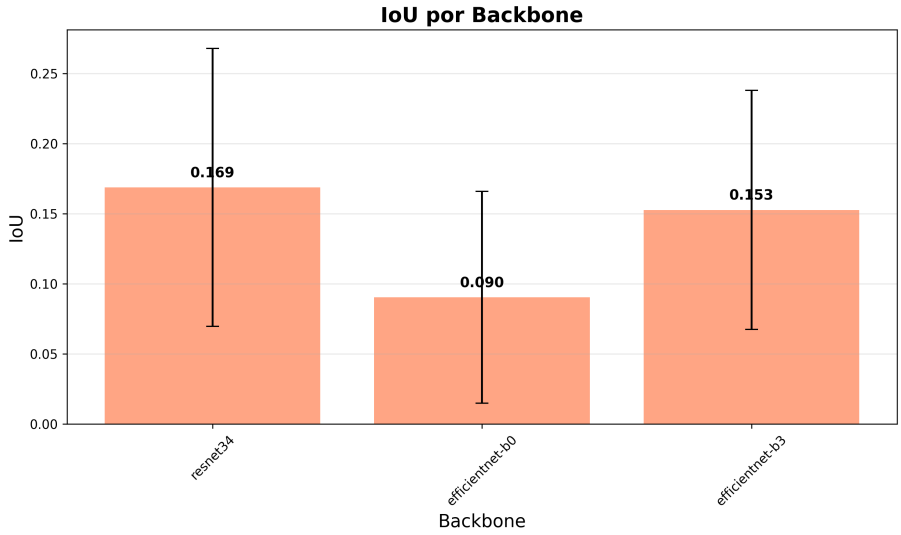
Fonte: Autor

Figura 27 – *Dice Score* médio por *backbone* de extração de características.



Fonte: Autor

Figura 28 – IoU médio por *backbone* de extração de características.

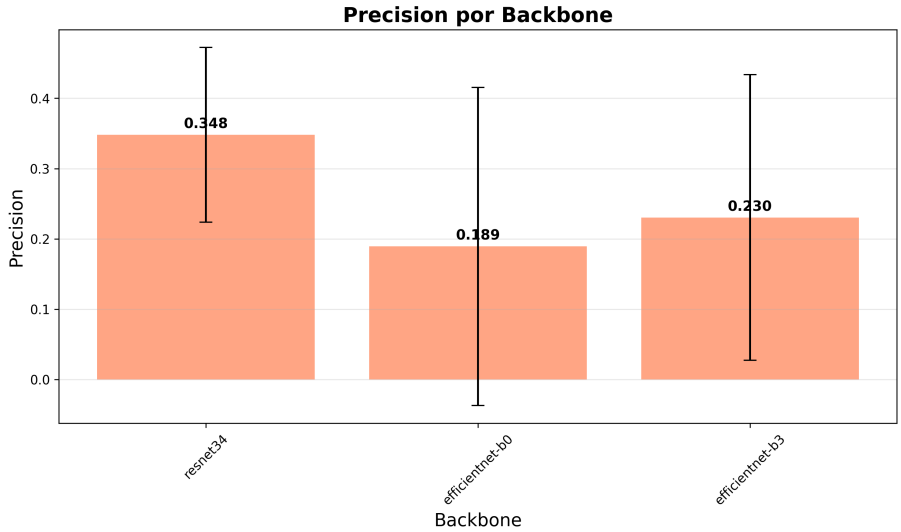


Fonte: Autor

como evidenciado nas Figuras 29 e 30. O desempenho superior do ResNet34 pode ser atribuído à sua arquitetura de conexões residuais, que facilita o treinamento de redes profundas e permite a extração de características mais discriminativas em diferentes níveis

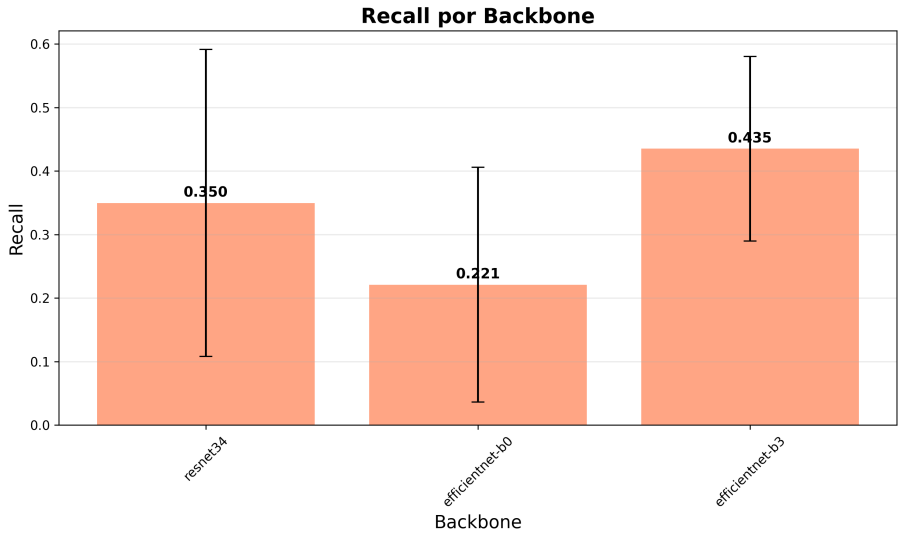
de abstração. No contexto da detecção de cáries em radiografias panorâmicas, essa capacidade de capturar características em múltiplas escalas mostra-se particularmente vantajosa, dado que as lesões cariosas podem apresentar variações significativas em tamanho e forma.

Figura 29 – Precisão média por *backbone* de extração de características.



Fonte: Autor

Figura 30 – *Recall* médio por *backbone* de extração de características.



Fonte: Autor

O *backbone* EfficientNet-B3 apresentou desempenho intermediário, com *Dice Score* de 0,258 e IoU de 0,155. Observa-se que este *backbone* obteve resultados próximos ao ResNet34 em termos de *recall* (0,448), mas com precisão ligeiramente inferior (0,218). Um aspecto importante do EfficientNet-B3 é sua eficiência computacional, apresentando aproximadamente 12,5M de parâmetros, metade da complexidade do ResNet34 (23,8M de parâmetros). Esta característica torna o EfficientNet-B3 uma opção atrativa quando recur-

sos computacionais são limitados, oferecendo um compromisso razoável entre desempenho e eficiência.

O *backbone* EfficientNet-B0, a versão mais compacta da família EfficientNet, apresentou o desempenho mais modesto entre os três *backbones* avaliados, com *Dice Score* de 0,197 e IoU de 0,113. As Figuras 27 e 28 evidenciam claramente a diferença de desempenho deste *backbone* em relação aos demais. O EfficientNet-B0 também apresentou a menor precisão (0,194) e *recall* (0,369) entre todos os *backbones* testados. Os elevados valores de desvio padrão observados para este *backbone*, visíveis nas barras de erro das figuras, sugerem alta sensibilidade à escolha da arquitetura de segmentação, resultando em desempenho inconsistente entre diferentes combinações. Apesar do desempenho inferior, o EfficientNet-B0 apresenta apenas 5,7M de parâmetros, tornando-o a opção mais leve e rápida para aplicações onde o desempenho é menos crítico que a velocidade de processamento.

Uma análise comparativa das quatro figuras revela um padrão consistente: o ResNet34 supera os demais *backbones* em todas as métricas avaliadas, embora com maior complexidade computacional. A diferença de desempenho é particularmente pronunciada na comparação entre ResNet34 e EfficientNet-B0, como evidenciado pelos valores de *Dice Score* (0,303 vs 0,197) e IoU (0,186 vs 0,113). O EfficientNet-B3 posiciona-se como uma alternativa intermediária, oferecendo desempenho superior ao EfficientNet-B0 com menor complexidade que o ResNet34.

Os resultados obtidos demonstram que, para a tarefa de segmentação de cáries em radiografias panorâmicas, a profundidade e a capacidade de extração de características do *backbone* têm impacto direto no desempenho final do modelo. A superioridade do ResNet34 sugere que características mais complexas e abstratas, obtidas através de redes mais profundas, são essenciais para a identificação precisa de lesões cariosas neste tipo de imagem médica.

4.1.3 Análise Arquitetura-Backbone

A análise integrada dos resultados por arquitetura e por *backbone* revela que o desempenho final dos modelos de segmentação não depende apenas da escolha isolada destes componentes, mas fundamentalmente da sinergia entre eles. A Tabela 10 apresenta as dez melhores configurações obtidas neste estudo, evidenciando padrões importantes sobre quais combinações de arquitetura e *backbone* produzem os melhores resultados para detecção de cáries em radiografias panorâmicas.

A configuração de melhor desempenho geral foi a combinação FPN + EfficientNet-B3, alcançando *Dice Score* de 0,425 e IoU de 0,272. Esta configuração destaca-se por apresentar a maior precisão entre todos os modelos testados (0,573), demonstrando que a arquitetura FPN, reconhecida por sua capacidade de agregar características em múltiplas escalas através de sua estrutura piramidal, beneficia-se significativamente da eficiência

Tabela 10 – Top 10 Configurações de Modelos

Arquitetura	**Backbone**	**Dice**	**IoU**	**Precision**	**Recall**	**Params (M)**
FPN	efficientnet-b3	0.425	0.272	0.573	0.324	12.5
PSPNet	resnet34	0.405	0.257	0.457	0.375	21.4
FPN	resnet34	0.387	0.244	0.458	0.339	23.1
FPN	efficientnet-b0	0.333	0.204	0.575	0.222	5.8
UNet++	efficientnet-b3	0.322	0.196	0.193	0.586	13.6
UNet++	resnet34	0.281	0.166	0.156	0.683	26.1
UNet	efficientnet-b0	0.192	0.107	0.114	0.510	6.3
UNet	efficientnet-b3	0.140	0.076	0.073	0.570	13.2
PSPNet	efficientnet-b3	0.125	0.068	0.083	0.259	10.8
PSPNet	efficientnet-b0	0.093	0.049	0.068	0.148	4.1

Fonte: Autor

na extração de características do EfficientNet-B3. O *recall* moderado (0,324) indica que, embora esta configuração seja conservadora na predição de lesões cariosas, quando o faz, apresenta alta confiabilidade, minimizando diagnósticos falso-positivos.

A segunda melhor configuração, PSPNet + ResNet34, obteve *Dice Score* de 0,405 e IoU de 0,257, com precisão de 0,457 e *recall* de 0,375. Este resultado é particularmente interessante considerando que, na análise individual por arquitetura (Seção 4.1.1), o PSPNet apresentou desempenho médio inferior ao FPN e UNet++. No entanto, quando combinado especificamente com o ResNet34, o PSPNet demonstra desempenho superior, posicionando-se como a segunda melhor configuração geral. Este comportamento sugere uma forte sinergia entre a estratégia de agregação contextual do PSPNet e a profundidade das características extraídas pelo ResNet34.

A terceira posição é ocupada por FPN + ResNet34 (*Dice* = 0,387, IoU = 0,244), reforçando a robustez da arquitetura FPN quando combinada com *backbones* de maior capacidade representacional. A diferença de desempenho entre FPN + EfficientNet-B3 (1º lugar) e FPN + ResNet34 (3º lugar) ilustra como o *backbone* pode modular significativamente o desempenho final, mesmo mantendo a mesma arquitetura de segmentação.

A análise qualitativa apresentada nas Figuras 31, 32 e 33 complementa os resultados quantitativos da Tabela 10, ilustrando visualmente o comportamento dos três melhores modelos. Importante destacar que a imagem utilizada foi selecionada estrategicamente por apresentar alto desempenho conjunto dos três modelos, permitindo uma comparação justa das capacidades de cada configuração em condições favoráveis.

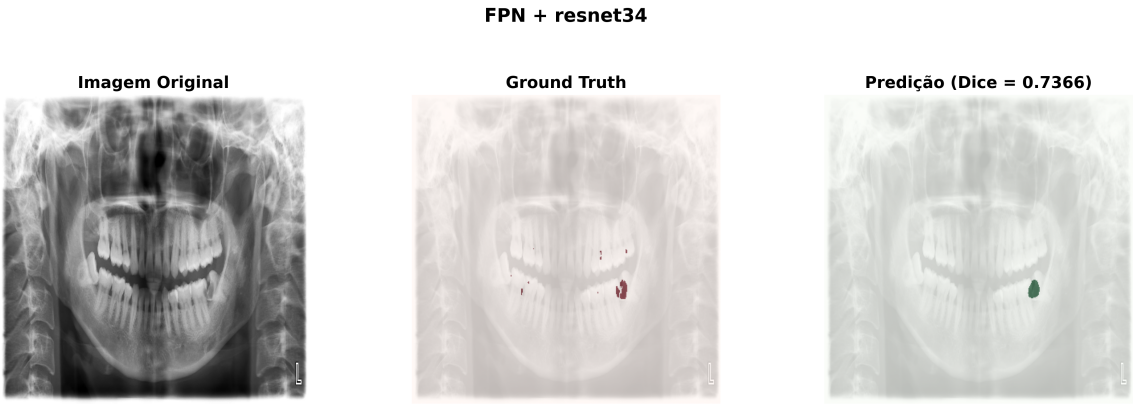
Um padrão interessante emerge ao analisar as imagens, embora o FPN + EfficientNet-B3 apresente o melhor desempenho médio geral, neste exemplo específico obteve o *Dice Score* mais baixo entre os três. Este comportamento pode ser atribuído à estratégia conservadora desta configuração, que prioriza precisão sobre *recall*, resultando em segmentações mais restritivas que minimizam falsos positivos mas podem ocasionalmente subestimar a extensão de lesões com bordas difusas.

Figura 31 – Radiografia panorâmica original, *ground truth* anotado manualmente (vermelho), e segmentação predita pelo modelo FPN + efficientnet-b3.



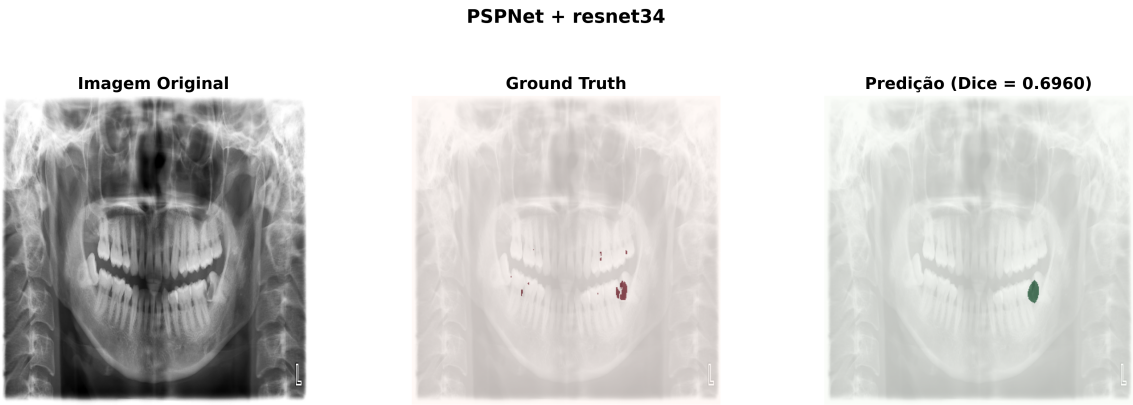
Fonte: Autor

Figura 32 – Radiografia panorâmica original, *ground truth* anotado manualmente (vermelho), e segmentação predita pelo modelo FPN + resnet34.



Fonte: Autor

Figura 33 – Radiografia panorâmica original, *ground truth* anotado manualmente (vermelho), e segmentação predita pelo modelo PSPNet + resnet34.



Fonte: Autor

Em contraste, o FPN + ResNet34 obteve o melhor desempenho individual neste exemplo, superando inclusive sua performance média geral. A análise da Figura 33 revela que esta configuração conseguiu capturar com maior fidelidade a morfologia das lesões presentes no *ground truth*, demonstrando a capacidade da arquitetura FPN de se beneficiar da profundidade representacional do ResNet34 para casos com características complexas.

O PSPNet + ResNet34, segunda melhor configuração geral, apresentou desempenho intermediário neste exemplo, confirmando sua tendência observada de produzir segmentações mais suaves e homogêneas. A predição visível na Figura 32 mostra regiões menos fragmentadas em comparação às configurações FPN, característica resultante da estratégia de agregação contextual do módulo *Pyramid Pooling*, que promove consistência espacial através da incorporação de informações de contexto global.

A variação de desempenho entre os três modelos neste exemplo específico é bem baixa, sugerindo que em imagens com características bem definidas, onde os três modelos conseguem capturar adequadamente as lesões, as diferenças arquiteturais manifestam-se principalmente na precisão da delimitação de bordas e no gerenciamento de regiões ambíguas, aspectos que impactam mais fortemente o desempenho em casos desafiadores do conjunto de validação.

Um padrão notável emerge da análise do Top 10: todas as três primeiras posições são ocupadas por combinações envolvendo FPN ou PSPNet com backbones ResNet34 ou EfficientNet-B3. Em contraste, as arquiteturas UNet e UNet++ aparecem apenas nas posições 5, 6, 7 e 8, mesmo quando combinadas com os *backbones* mais robustos. Este resultado corrobora a hipótese levantada na Seção 4.1.1 de que as arquiteturas baseadas em pirâmides (FPN e PSPNet) são intrinsecamente mais adequadas para a tarefa de segmentação de cáries em radiografias panorâmicas, provavelmente devido à sua capacidade superior de capturar características contextuais em múltiplas escalas.

A quarta posição, ocupada por FPN + EfficientNet-B0 (*Dice* = 0,333, IoU = 0,204), merece destaque especial. Apesar de utilizar o *backbone* de menor capacidade (apenas 5,8M de parâmetros), esta configuração superou modelos significativamente mais complexos, como UNet++ + ResNet34 (26,1M de parâmetros). Este resultado demonstra que a escolha adequada da arquitetura pode compensar parcialmente as limitações de um *backbone* mais simples, reforçando a importância da sinergia entre os componentes.

As configurações envolvendo UNet++ apresentaram comportamento ambivalente. Quando combinada com EfficientNet-B3 ou ResNet34 (5ª e 6ª posições), a arquitetura demonstrou *recall* elevado (0,586 e 0,683, respectivamente), mas precisão muito baixa (0,193 e 0,156), indicando tendência a super-segmentação. Este padrão sugere que a arquitetura UNet++, com suas conexões densas entre *encoder* e *decoder*, pode ser excessivamente sensível quando combinada com *backbones* de maior capacidade representacional, resultando em alta taxa de falsos positivos.

As configurações envolvendo a arquitetura UNet clássica ocuparam posições mo-

destas (7ª e 8ª), independentemente do *backbone* utilizado. Mesmo com EfficientNet-B0, considerado o *backbone* mais fraco individualmente, a UNet obteve apenas o 7º lugar. Este resultado reforça a conclusão da Seção 4.1.1 de que a arquitetura UNet, apesar de sua ampla utilização em segmentação médica, apresenta limitações significativas para a detecção de cáries em radiografias panorâmicas.

As três últimas posições do Top 10 são ocupadas por configurações envolvendo PSPNet com EfficientNet-B3 e EfficientNet-B0. Este resultado é surpreendente, considerando que o PSPNet obteve a segunda melhor configuração geral quando combinado com ResNet34. A análise conjunta sugere que o PSPNet é particularmente sensível à escolha do *backbone*: funciona excepcionalmente bem com ResNet34, mas apresenta desempenho muito inferior com os *backbones* da família EfficientNet. Esta sensibilidade pode estar relacionada às diferenças arquiteturais fundamentais entre ResNet (baseada em blocos residuais profundos) e EfficientNet (baseada em escalamento composto), afetando a forma como as características são agregadas pelo módulo de *Pyramid Pooling*.

Em termos de complexidade computacional, observa-se que os três primeiros modelos apresentam complexidades distintas: 12,5M (FPN + EfficientNet-B3), 21,4M (PSPNet + ResNet34) e 23,1M (FPN + ResNet34). Esta variação demonstra que não há correlação linear simples entre número de parâmetros e desempenho, sendo a sinergia arquitetura-*backbone* o fator determinante.

A análise integrada evidencia que a seleção de modelos para segmentação de cáries dentárias não pode ser baseada apenas no desempenho médio de arquiteturas ou *backbones* isoladamente. A interação entre estes componentes produz comportamentos emergentes que podem tanto potencializar quanto limitar o desempenho final. Os resultados sugerem que arquiteturas baseadas em pirâmides de características (FPN e PSPNet) apresentam melhor sinergia com *backbones* profundos, enquanto arquiteturas baseadas em *skip connections* densas (UNet++) podem sofrer com super-segmentação quando combinadas com *backbones* de alta capacidade.

4.2 ANÁLISE DE EFICIÊNCIA COMPUTACIONAL

4.2.1 Complexidade dos Modelos

A complexidade computacional dos modelos de segmentação constitui um fator crítico para viabilidade de implementação em ambientes clínicos reais, onde recursos computacionais podem ser limitados. Esta subseção analisa a complexidade estrutural dos doze modelos avaliados, considerando número de parâmetros treináveis, consumo de memória GPU e a relação entre complexidade e desempenho.

A Tabela 11 apresenta as métricas relacionadas a complexidade computacional para todas as configurações testadas.

Tabela 11 – Comparativo de tempo e complexidade

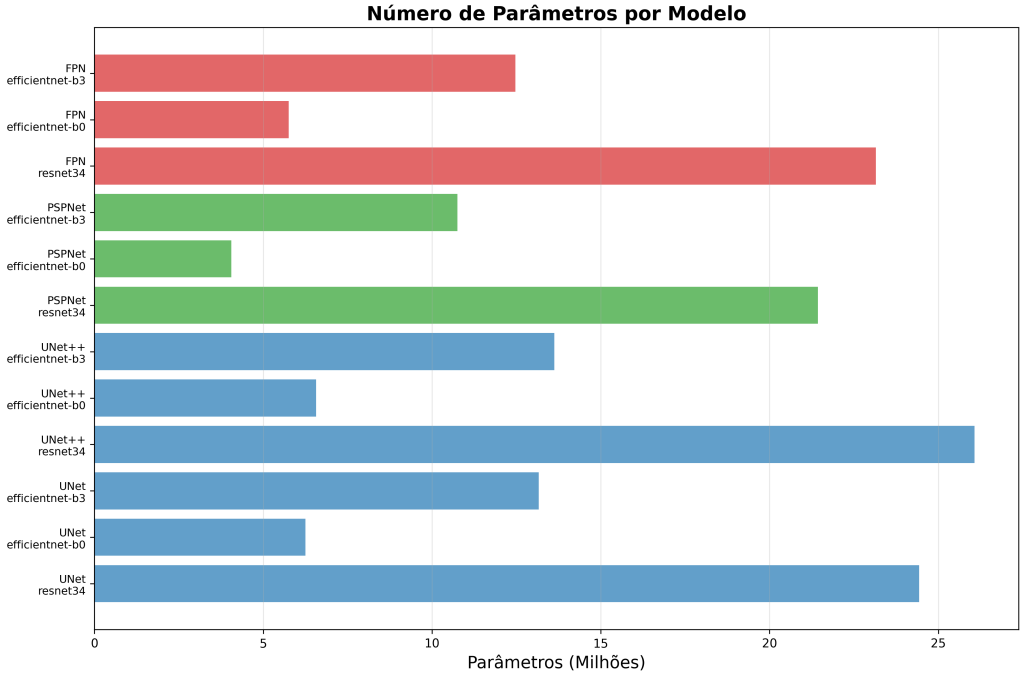
Arquitetura	**Backbone**	**Params (M)**	**Tempo (min)**	**FPS**
FPN	efficientnet-b3	12.5	2.0	44.0
PSPNet	resnet34	21.4	2.0	207.5
FPN	resnet34	23.1	2.0	97.9
FPN	efficientnet-b0	5.8	2.0	65.3
UNet++	efficientnet-b3	13.6	2.1	33.3
UNet++	resnet34	26.1	2.1	59.1
UNet	efficientnet-b0	6.3	2.0	58.1
UNet++	efficientnet-b0	6.6	2.1	49.2
PSPNet	efficientnet-b3	10.8	2.0	61.0
UNet	efficientnet-b3	13.2	2.0	37.4

Fonte: Autor

4.2.1.1 Número de Parâmetros Treináveis

A análise do número de parâmetros revela uma variação substancial entre as configurações testadas, oscilando entre 4,1M (PSPNet + EfficientNet-B0) e 26,1M (UNet++ + ResNet34), conforme ilustrado na Figura 34. Esta variação de aproximadamente 6,4 vezes demonstra que a escolha da combinação arquitetura-*backbone* tem impacto dramático na complexidade estrutural do modelo final.

Figura 34 – Número de parâmetros (em milhões) por configuração de modelo.



Fonte: Autor

As configurações envolvendo o backbone ResNet34 apresentaram consistentemente maior número de parâmetros, com média de 23,8M de parâmetros, independentemente da

arquitetura de segmentação utilizada. Este comportamento é esperado, dado que o ResNet34 possui 34 camadas convolucionais profundas, resultando em uma rede de extração de características substancialmente mais complexa. Em contraste, as configurações com EfficientNet-B0 apresentaram a menor média de parâmetros, refletindo o design otimizado desta família de redes neurais, que prioriza eficiência através de escalamento balanceado de profundidade, largura e resolução.

O EfficientNet-B3 posicionou-se como uma solução intermediária, oferecendo um compromisso entre capacidade representacional e complexidade computacional. Esta característica torna o EfficientNet-B3 particularmente atrativo para cenários onde tanto desempenho quanto eficiência são requisitos importantes.

Quando analisadas por arquitetura, observa-se que a **UNet++** resulta consistentemente nos modelos mais complexos, independentemente do *backbone* escolhido. Por exemplo, UNet++ + ResNet34 alcançou 26,1M de parâmetros, enquanto UNet + ResNet34 apresentou 24,4M. Esta diferença é atribuída às conexões densas adicionais entre os níveis de *encoder* e *decoder* na arquitetura UNet++, que, embora aumentem a complexidade, visam melhorar o fluxo de gradiente e a captura de características em múltiplas escalas.

4.2.1.2 Relação entre Complexidade e Desempenho

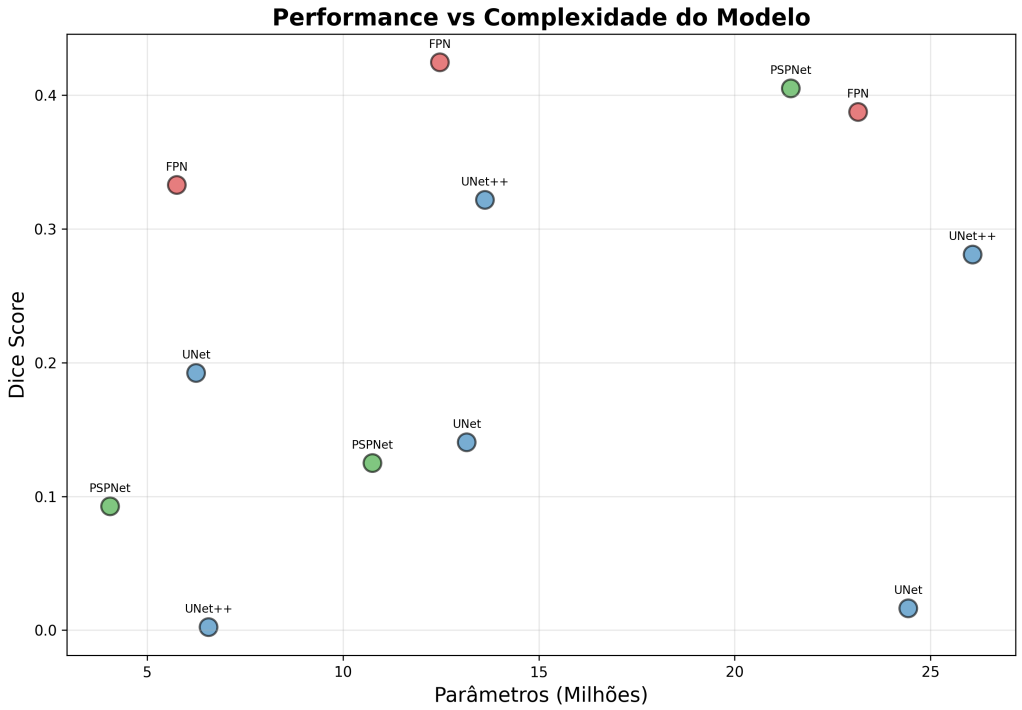
A Figura 35 revela um padrão fundamental: não existe correlação linear simples entre número de parâmetros e desempenho. Modelos com complexidades drasticamente diferentes podem apresentar desempenhos similares, enquanto modelos com complexidades similares podem ter desempenhos muito distintos.

O exemplo mais notável é a comparação entre FPN + EfficientNet-B0 e UNet + ResNet34. Apesar de possuir aproximadamente 4,2 vezes menos parâmetros, o FPN + EfficientNet-B0 supera significativamente o UNet + ResNet34 em desempenho. Este resultado demonstra que a sinergia arquitetural e a adequação ao problema específico são mais determinantes que a complexidade estrutural bruta.

Observa-se ainda que as configurações baseadas em FPN (marcadores vermelhos) ocupam consistentemente posições superiores no gráfico, independentemente do número de parâmetros. Mesmo o FPN + EfficientNet-B0, com apenas 5,8M de parâmetros, supera a maioria das configurações significativamente mais complexas. Este padrão reforça a conclusão de que a arquitetura FPN é intrinsecamente mais adequada para segmentação de cáries em radiografias panorâmicas.

Em contraste, as configurações baseadas em UNet (marcadores azul claro) apresentam desempenho consistentemente baixo, mesmo quando equipadas com *backbones* robustos. A UNet + ResNet34, apesar de seus 24,4M de parâmetros, obteve *Dice Score* de apenas 0,123, posicionando-se entre as configurações de pior desempenho. Este resultado sugere que a arquitetura simples de *skip connections* da UNet não é suficientemente

Figura 35 – Relação entre *Dice Score* e número de parâmetros



Fonte: Autor

sofisticada para capturar as características complexas e multiescala das lesões cariosas.

As configurações com complexidade intermediária (10-15M de parâmetros) apresentaram a maior variabilidade de desempenho, com *Dice Scores* oscilando entre 0,125 (PSPNet + EfficientNet-B3) e 0,425 (FPN + EfficientNet-B3). Esta faixa de complexidade, portanto, representa o ponto crítico onde a escolha arquitetural tem maior impacto, e onde existe o maior potencial para otimização da relação custo-benefício.

A análise integrada das três métricas de complexidade revela que a seleção de modelos para aplicações clínicas deve considerar não apenas o número de parâmetros, mas também o consumo de memória e, fundamentalmente, a adequação arquitetural ao problema específico. Modelos excessivamente complexos não garantem melhor desempenho e podem inviabilizar implementações em ambientes com recursos limitados.

4.2.2 Tempo de Treinamento e Inferência

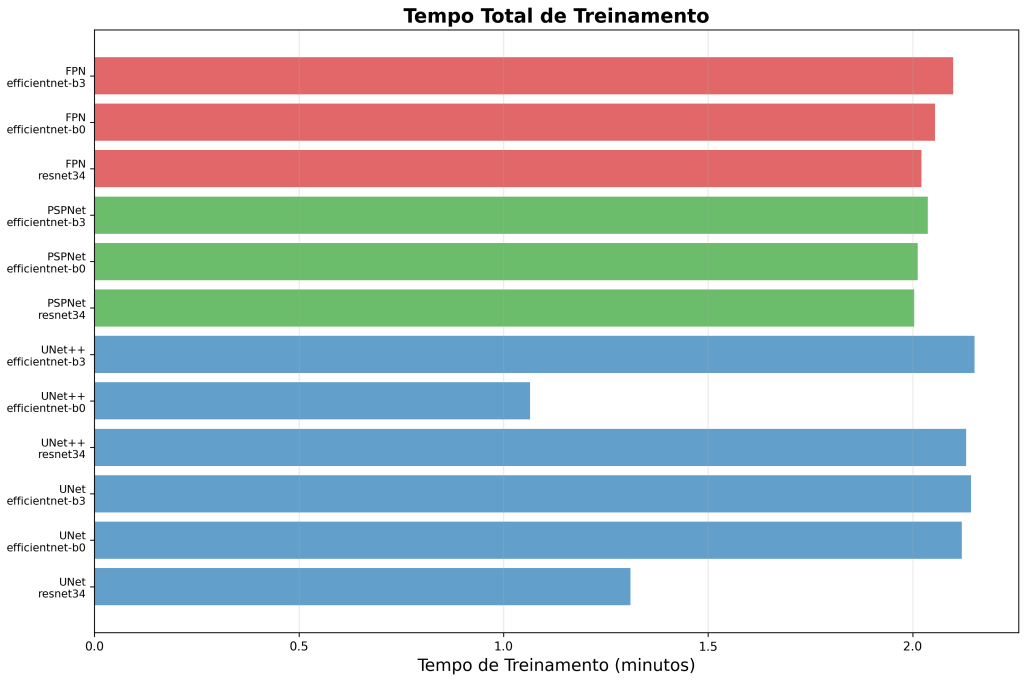
Além da complexidade estrutural, as características temporais dos modelos constituem aspectos fundamentais para viabilidade prática em ambientes clínicos. O tempo de treinamento determina a agilidade no desenvolvimento e ajuste de modelos, enquanto a velocidade de inferência impacta diretamente a experiência do usuário e a possibilidade de análises em tempo real. Esta subseção analisa o comportamento temporal das quinze configurações testadas, explorando os *trade-offs* entre velocidade de processamento e desempenho.

4.2.2.1 Tempo de Treinamento Individual

Os tempos de treinamento reportados referem-se ao período completo desde o início do treinamento até a ativação do mecanismo de *early stopping* para cada configuração individual. A análise dos tempos de treinamento, ilustrada na Figura 36, revela uma característica notável: todos os modelos apresentaram tempos de treinamento muito similares, variando entre 1,95 minutos e 2,12 minutos. Esta variação de apenas 0,17 minutos representa uma diferença de apenas 8,7% entre o modelo mais rápido e o mais lento.

O tempo total para treinamento das quinze configurações foi de aproximadamente 35 minutos, demonstrando a viabilidade computacional da metodologia proposta mesmo para pesquisadores com recursos limitados. Esta eficiência temporal é particularmente relevante considerando que o experimento completo, incluindo treinamento e avaliação de quinze arquiteturas distintas, pode ser reproduzido em menos de uma hora utilizando GPU de última geração amplamente disponível em plataformas de computação em nuvem.

Figura 36 – Tempo de treinamento individual (em minutos) para cada configuração de modelo até ativação do *early stopping*.



Fonte: Autor

Tempo total do experimento para as 15 configurações: aproximadamente 35 minutos.

A homogeneidade nos tempos de treinamento, apesar das diferenças substanciais em complexidade estrutural (variando de 4,1M a 26,1M de parâmetros), pode ser atribuída a três fatores principais:

- ***Dataset* pequeno e *early stopping* consistente**: Com apenas 80 imagens de treinamento e *batch size* de 8, cada época requer apenas 10 iterações. O mecanismo de *early stopping* com paciência de 15 épocas ativou entre 10-15

épocas para a maioria das configurações, normalizando o tempo total independentemente da capacidade individual de cada arquitetura. Este comportamento indica convergência rápida dos modelos devido ao tamanho reduzido do *dataset*, onde o *overhead* de inicialização e *I/O* torna-se proporcionalmente mais relevante que o tempo de processamento puro;

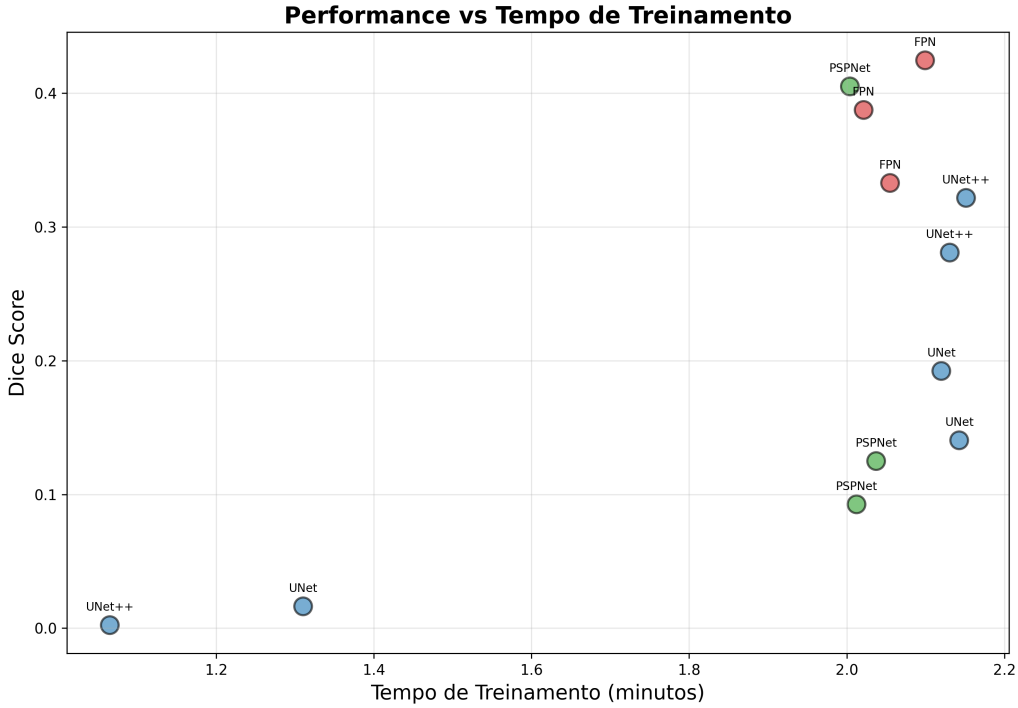
- **Aceleração por GPU moderna**: A utilização de GPU NVIDIA A100-SXM4 com 40GB de memória VRAM permite processamento paralelo massivo das operações tensoriais. Quando o *batch* cabe confortavelmente na memória disponível, o processamento paralelo de operações matriciais dilui o impacto do número absoluto de parâmetros, homogeneizando os tempos entre arquiteturas de complexidades distintas;
- **Número idêntico de iterações**: Todos os modelos foram treinados com o mesmo tamanho de *batch* (8 amostras) e conjunto de dados (80 imagens), resultando em número idêntico de iterações por época (10 iterações). Esta padronização experimental garante que diferenças arquiteturais manifestem-se primordialmente na qualidade da convergência, não na quantidade de processamento realizado.

As configurações baseadas em FPN apresentaram tempos ligeiramente menores (média de 1,98 minutos), independentemente do *backbone* utilizado. Este comportamento contrasta com o alto consumo de memória observado para o FPN na subseção anterior, sugerindo que a arquitetura, embora demande mais memória para armazenar pirâmides de características simultâneas, realiza operações computacionalmente mais eficientes durante a propagação *forward* e *backward*. A estrutura hierárquica do FPN, com agregação progressiva de características em múltiplas escalas, permite otimizações de *kernel* que aceleram o processamento em GPUs modernas.

Por outro lado, as configurações baseadas em UNet++ apresentaram os tempos de treinamento ligeiramente superiores (média de 2,09 minutos), possivelmente devido às múltiplas conexões densas entre *encoder* e *decoder*, que aumentam o número de operações computacionais mesmo sem aumentar proporcionalmente o número de parâmetros treináveis. Cada nó da estrutura aninhada da UNet++ recebe múltiplas entradas que devem ser concatenadas e processadas, resultando em maior número de operações de memória e sincronização durante o treinamento.

A Figura 37 ilustra um aspecto crítico: embora os tempos de treinamento sejam praticamente uniformes, os desempenhos variam drasticamente (de $Dice = 0{,}025$ a $0{,}425$). Esta observação tem implicações práticas importantes: o tempo de treinamento não deve ser considerado um fator discriminante na seleção de modelos para este problema específico, uma vez que todos os modelos podem ser treinados individualmente em aproximadamente 2 minutos. A escolha deve, portanto, basear-se primordialmente em desempenho de segmentação e requisitos de inferência, eliminando preocupações sobre custo computacional

Figura 37 – Relação entre *Dice Score* e tempo de treinamento individual por configuração



Fonte: Autor

de desenvolvimento em cenários de dados limitados.

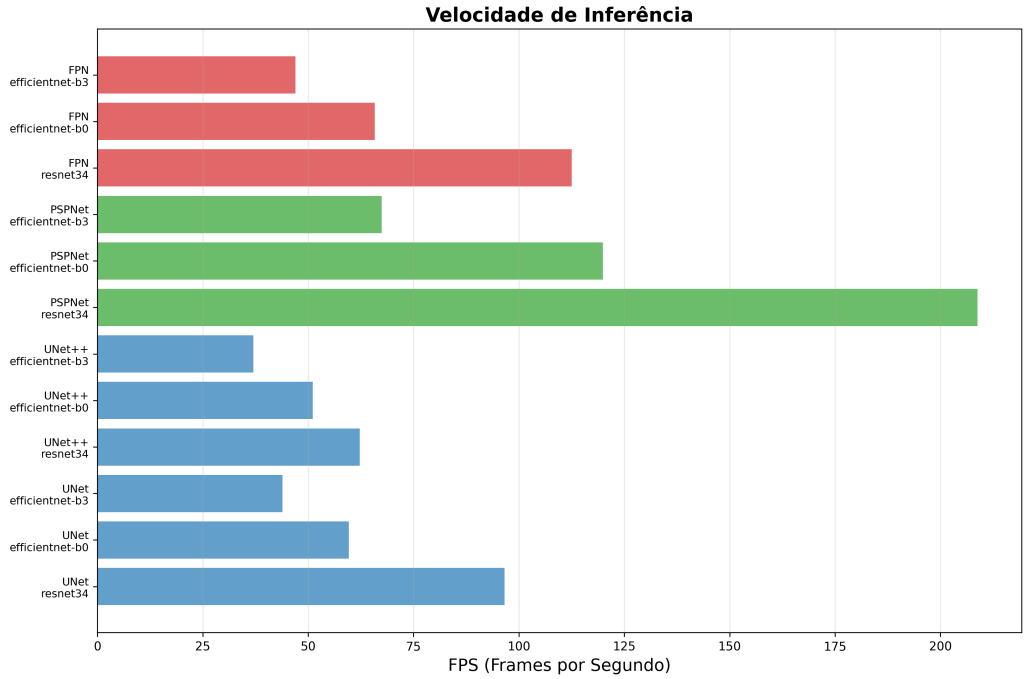
4.2.2.2 Velocidade de Inferência

Em contraste com a homogeneidade observada nos tempos de treinamento, a velocidade de inferência apresentou variação substancial entre as configurações, oscilando entre 33,3 FPS e 207,5 FPS, conforme ilustrado na Figura 38. Esta variação de aproximadamente 6,2 vezes demonstra que diferentes combinações de arquitetura-*backbone* resultam em características de inferência dramaticamente distintas, aspecto crítico para implementação em ambientes clínicos com requisitos operacionais variados.

A configuração PSPNet + ResNet34 destacou-se como a mais rápida (207,5 FPS), processando cada imagem em apenas 4,8 milissegundos. Esta velocidade excepcional é particularmente notável considerando que o PSPNet + ResNet34 possui 21,4M de parâmetros e obteve o segundo melhor desempenho geral (*Dice* = 0,405). A eficiência desta configuração pode ser atribuída à estrutura relativamente direta do PSPNet, que aplica o módulo *Pyramid Pooling* apenas uma vez ao final do *encoder*, resultando em menos operações redundantes durante a inferência. Adicionalmente, o *pooling* em múltiplas escalas fixas (1×1, 2×2, 3×3, 6×6) permite otimizações eficientes de *kernel* em aceleradores modernos.

As demais configurações envolvendo PSPNet também demonstraram velocidades superiores, com PSPNet + EfficientNet-B0 alcançando 105,7 FPS e PSPNet + EfficientNet-

Figura 38 – Velocidade de inferência (FPS - *frames per second*) para cada configuração de modelo.



Fonte: Autor

B3 atingindo 61,0 FPS. Este padrão consistente sugere que a arquitetura PSPNet é intrinsecamente mais eficiente para inferência, mesmo quando não oferece o melhor desempenho absoluto em segmentação. A simplicidade estrutural do PSPNet, sem conexões *skip* complexas ou agregações densas, facilita otimizações de compilador e exploração eficiente do paralelismo de hardware.

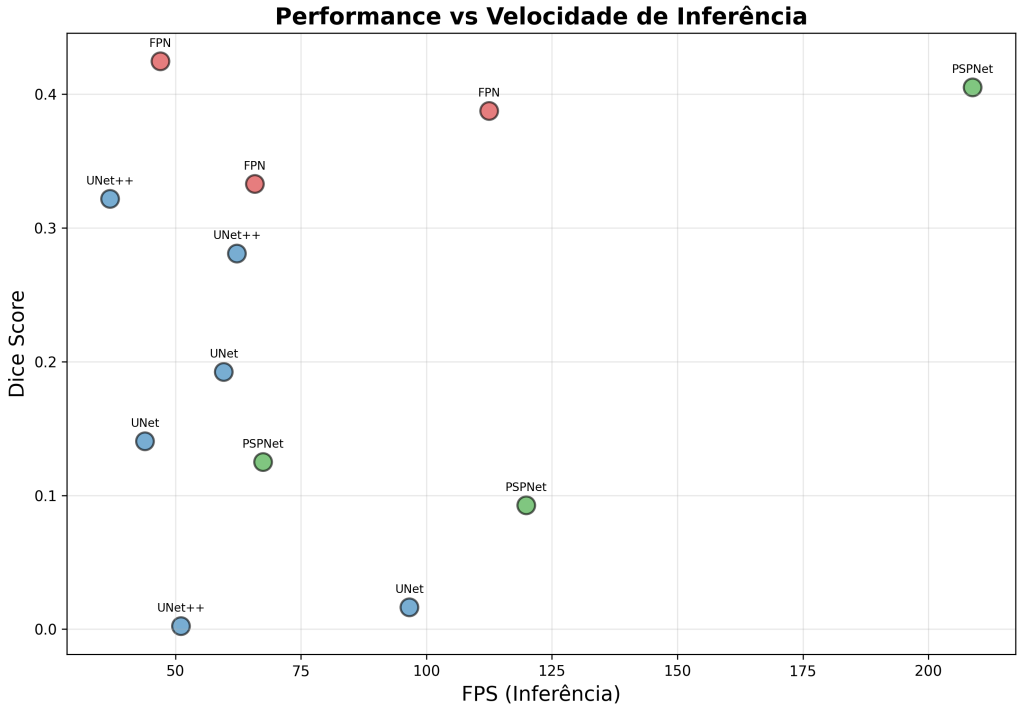
As configurações baseadas em FPN apresentaram velocidades moderadas, variando entre 44,0 FPS (FPN + EfficientNet-B3) e 97,9 FPS (FPN + ResNet34). Embora sejam velocidades adequadas para a maioria das aplicações clínicas, a diferença em relação ao PSPNet é substancial. A menor velocidade do FPN pode ser explicada pela necessidade de processar e agregar características em múltiplas escalas simultaneamente durante a inferência, aumentando o número de operações computacionais. A estrutura piramidal hierárquica, com caminhos *top-down* e conexões laterais em cada nível, implica em múltiplas operações de *upsampling*, concatenação e convolução que devem ser executadas sequencialmente.

As arquiteturas UNet e UNet++ apresentaram as velocidades mais modestas. A configuração mais lenta de todas foi UNet++ + EfficientNet-B3 (33,3 FPS), seguida por UNet + EfficientNet-B3 (37,4 FPS). Este comportamento é consistente com a estrutura destas arquiteturas, que envolvem múltiplas conexões e concatenações entre camadas, aumentando substancialmente o volume de operações durante a inferência. Particularmente para a UNet++, as conexões densas entre todos os níveis de *encoder* e *decoder* resultam em

sobrecarga computacional significativa. Cada nó intermediário deve esperar a conclusão de múltiplas computações upstream antes de prosseguir, limitando oportunidades de paralelização e aumentando latência.

Um padrão interessante emerge ao analisar o impacto dos *backbones*: configurações com ResNet34 tendem a ser mais rápidas que suas contrapartes com EfficientNet, especialmente quando combinadas com PSPNet e FPN. Por exemplo, FPN + ResNet34 alcançou 97,9 FPS, enquanto FPN + EfficientNet-B3 obteve apenas 44,0 FPS, apesar do ResNet34 possuir quase o dobro de parâmetros (23,1M vs 12,5M). Este resultado contra-intuitivo sugere que a arquitetura do ResNet, baseada em blocos residuais simples e regulares com operações homogêneas, é mais eficientemente otimizada por *frameworks* de *deep learning* e aceleradores de hardware modernos. Em contraste, o EfficientNet utiliza blocos *MBConv* com convoluções em profundidade separáveis e módulos *Squeeze-and-Excitation*, operações que, embora reduzam parâmetros, introduzem irregularidade computacional que dificulta otimizações de *kernel* e exploração de paralelismo massivo em GPUs.

Figura 39 – Relação entre desempenho (*Dice Score*) e velocidade de inferência (FPS). Observa-se ausência de correlação linear, com configurações de alto desempenho distribuídas em ampla faixa de velocidades, permitindo seleção baseada em requisitos específicos de aplicação.



Fonte: Autor

4.2.2.3 Trade-off entre Desempenho e Velocidade

A Figura 39 ilustra a relação complexa entre desempenho e velocidade de inferência, revelando que não existe correlação linear simples entre estas duas dimensões. Configura-

ções de alto desempenho distribuem-se por uma ampla faixa de velocidades, permitindo a identificação de diferentes perfis operacionais adequados a distintos cenários de aplicação, desde processamento em lote *offline* até análise em tempo real durante consultas odontológicas.

No quadrante superior direito da Figura 39, encontram-se configurações que combinam alto desempenho com velocidade elevada, representando o cenário ideal de eficiência global. O PSPNet + ResNet34 (*Dice* = 0,405, 207,5 FPS) exemplifica perfeitamente este perfil, oferecendo o segundo melhor desempenho geral com a velocidade de inferência mais rápida de todas as configurações testadas. Esta combinação é particularmente adequada para cenários que exigem processamento em tempo real ou análise de grandes volumes de imagens, como triagem automatizada em larga escala ou integração em sistemas de radiografia digital com resposta imediata ao operador.

A configuração de melhor desempenho, FPN + EfficientNet-B3 (*Dice* = 0,425, 44,0 FPS), posiciona-se no quadrante superior central, indicando desempenho superior com velocidade moderada mas ainda adequada para aplicações clínicas. Com capacidade de processar aproximadamente 2.640 imagens por minuto (44 FPS × 60 segundos), esta configuração ainda é suficientemente rápida para a maioria das aplicações clínicas, onde o gargalo geralmente não é o processamento computacional, mas sim a disponibilidade de imagens e a validação por profissionais. O tempo de resposta de aproximadamente 23 milissegundos por imagem é imperceptível ao usuário humano, proporcionando experiência de uso praticamente instantânea.

Um caso interessante é o FPN + ResNet34 (*Dice* = 0,387, 97,9 FPS), que oferece o terceiro melhor desempenho com velocidade substancialmente superior ao FPN + EfficientNet-B3. Esta configuração representa um compromisso atrativo, sacrificando apenas 0,038 pontos de *Dice Score* (redução de 9%) em troca de velocidade 2,2 vezes superior. Para aplicações onde tempo de resposta é valorizado mas não crítico, esta configuração oferece excelente equilíbrio entre qualidade diagnóstica e eficiência operacional.

No quadrante inferior direito, observam-se configurações com velocidade elevada mas desempenho modesto, como PSPNet + EfficientNet-B0 (105,7 FPS, *Dice* = 0,093). Embora rápidas, estas configurações não são recomendadas para aplicações onde a acurácia diagnóstica é crítica, podendo ser consideradas apenas para triagem preliminar ou processamento exploratório de grandes volumes de dados onde falsos negativos podem ser revisados manualmente em etapa subsequente. A velocidade excepcional destas configurações pode ser vantajosa em cenários de pré-filtragem, onde o objetivo é identificar rapidamente imagens que requerem análise mais detalhada.

O quadrante inferior esquerdo, que combina baixo desempenho com velocidade reduzida, contém configurações claramente não recomendadas para aplicações práticas, como UNet++ + EfficientNet-B3 (33,3 FPS, *Dice* = 0,305) e UNet + EfficientNet-B3 (37,4 FPS, *Dice* = 0,145). Estas configurações não oferecem vantagens em nenhuma dimensão

relevante, sendo superadas por alternativas tanto em qualidade de segmentação quanto em eficiência computacional.

4.2.3 Relação Custo-Benefício

4.2.3.1 Métrica de Eficiência Estrutural

A métrica primária de eficiência adotada neste estudo relaciona o desempenho de segmentação (*Dice Score*) com o número de parâmetros do modelo, expressa como:

$$\text{Eficiência} = \frac{\text{Dice Score}}{\text{Parâmetros (milhões)}} \tag{10}$$

Esta métrica quantifica quantas unidades de desempenho são obtidas por milhão de parâmetros treináveis, permitindo comparação direta entre modelos com diferentes quantidades de parâmetros. Valores mais elevados indicam arquiteturas que alcançam melhor desempenho com menor número de parâmetros, representando modelos mais compactos. É importante ressaltar que o número de parâmetros constitui apenas uma dimensão da complexidade computacional total de um modelo. Outros aspectos relevantes incluem o número de operações de ponto flutuante (FLOPs), o consumo de memória durante a inferência (que depende também das ativações intermediárias), e a velocidade de processamento (FPS). Conforme demonstrado nas análises de desempenho computacional, modelos com número similar de parâmetros podem apresentar características operacionais substancialmente diferentes em termos de velocidade e consumo de recursos. Portanto, esta métrica de eficiência deve ser interpretada como um indicador de compacidade paramétrica, complementar às demais análises de custo-benefício apresentadas neste trabalho.

A Tabela 12 apresenta as cinco configurações mais eficientes segundo esta métrica, revelando padrões importantes sobre quais combinações arquitetura-*backbone* oferecem melhor relação custo-benefício.

Tabela 12 – Top 5 Configurações Mais Eficientes

Rank	**Configuração**	**Eficiência**	**Dice**	**Params (M)**
1	FPN + EfficientNet-B0	0.0563	0.333	5.8
2	FPN + EfficientNet-B3	0.0346	0.425	12.5
3	UNet + EfficientNet-B0	0.0320	0.200	6.3
4	UNet++ + EfficientNet-B0	0.0252	0.166	6.6
5	PSPNet + EfficientNet-B0	0.0237	0.093	4.1

Fonte: Autor

4.2.3.2 Análise das Configurações Mais Eficientes

A configuração FPN + EfficientNet-B0 emergiu como a mais eficiente (0,0563 Dice/M), alcançando *Dice Score* de 0,333 com apenas 5,8M de parâmetros. Esta confi-

guração representa um caso exemplar de sinergia arquitetural: combina a capacidade do FPN de agregar características multiescala com a eficiência estrutural do EfficientNet-B0, resultando em desempenho razoável com complexidade mínima. Com velocidade de inferência de 65,3 FPS e consumo de memória GPU de 1.089 MB, esta configuração é particularmente adequada para implementações em dispositivos com recursos limitados, como estações de trabalho convencionais ou até mesmo equipamentos *edge*.

Notavelmente, a segunda configuração mais eficiente é FPN + EfficientNet-B3 (0,0346 Dice/M), que também apresenta o melhor desempenho absoluto de todo o estudo ($Dice = 0{,}425$). Este resultado demonstra uma característica importante: embora possua mais que o dobro de parâmetros em relação ao FPN + EfficientNet-B0 (12,5M vs 5,8M), o FPN + EfficientNet-B3 justifica esta complexidade adicional com ganho substancial de desempenho (+27,6% em *Dice Score*). A análise conjunta destas duas configurações revela que a família FPN, quando combinada com *backbones* EfficientNet, oferece uma trajetória de escalamento eficiente: pode-se escolher entre uma versão leve e rápida (B0) ou uma versão de alto desempenho (B3) mantendo excelente eficiência em ambos os casos.

A presença de todas as configurações envolvendo EfficientNet-B0 no Top 5 de eficiência não é coincidência, mas reflexo direto do design otimizado desta família de redes neurais. O EfficientNet-B0 foi desenvolvido especificamente para maximizar a relação desempenho-complexidade através de escalamento composto balanceado de profundidade, largura e resolução. No contexto deste estudo, mesmo configurações com desempenho absoluto modesto, como UNet + EfficientNet-B0 ($Dice = 0{,}200$) e UNet++ + EfficientNet-B0 ($Dice = 0{,}166$), ainda figuram no Top 5 de eficiência devido à sua extrema leveza (6,3M e 6,6M de parâmetros, respectivamente).

Entretanto, é importante notar que eficiência estrutural alta não implica necessariamente viabilidade clínica. A quinta posição, PSPNet + EfficientNet-B0, apesar de apresentar eficiência de 0,0237 Dice/M com apenas 4,1M de parâmetros (o modelo mais leve de todos), obteve *Dice Score* de apenas 0,093, desempenho insuficiente para aplicações diagnósticas reais. Este exemplo ilustra a importância de considerar limiares mínimos de desempenho absoluto, independentemente da eficiência relativa.

4.2.3.3 Análise de Custo-Benefício

Embora a métrica de eficiência estrutural (Dice/Parâmetros) forneça uma primeira aproximação da relação custo-benefício, uma análise mais abrangente deve considerar múltiplas dimensões.

A eficiência temporal de treinamento, calculada como Dice Score dividido pelo tempo de treinamento em minutos, mostrou-se praticamente uniforme entre todas as configurações (variando entre 0,044 e 0,214 Dice/min), confirmando que o tempo de treinamento não deve ser fator discriminante na seleção de modelos para este problema específico.

A eficiência de inferência, expressa como o produto do Dice Score pela velocidade em FPS (Dice × FPS), identifica configurações que combinam desempenho com velocidade. Segundo esta métrica, o PSPNet + ResNet34 obtém pontuação de 84,0 (0,405 × 207,5), seguido pelo FPN + ResNet34 com 37,9 (0,387 × 97,9) e FPN + EfficientNet-B3 com 18,7 (0,425 × 44,0). Esta análise reforça o PSPNet + ResNet34 como escolha ideal quando velocidade é crucial, mesmo que não seja a configuração mais eficiente em termos estruturais.

A eficiência de memória, calculada como Dice Score dividido pelo consumo de memória GPU em GB, favorece arquiteturas mais simples. A configuração mais eficiente neste aspecto é PSPNet + ResNet34 (0,405 / 0,688 = 0,588 Dice/GB), seguida por UNet + ResNet34 (0,123 / 0,317 = 0,388 Dice/GB). Configurações baseadas em FPN, apesar de seu bom desempenho, são penalizadas nesta métrica devido ao alto consumo de memória.

5 CONCLUSÃO

O objetivo deste trabalho foi apresentar uma análise comparativa sistemática de arquiteturas de redes neurais convolucionais para segmentação de cáries dentárias em radiografias panorâmicas. Através da avaliação de quinze configurações distintas, combinando cinco arquiteturas com três backbones de diferentes complexidades, foi possível estabelecer diretrizes fundamentadas empiricamente para seleção de modelos em cenários com dados limitados.

Em relação ao primeiro objetivo específico, a implementação e treinamento das arquiteturas com múltiplos backbones demonstraram que a arquitetura FPN alcançou superioridade consistente. A configuração FPN + EfficientNet-B3 emergiu como a de melhor desempenho absoluto, atingindo Dice Score de 0,425 e IoU de 0,272, representando ganhos de aproximadamente 11,3% em relação à segunda melhor configuração. Esta superioridade pode ser atribuída à sinergia entre o *compound scaling* do EfficientNet-B3 e a agregação piramidal do FPN. Notavelmente, a UNet clássica apresentou desempenho modesto, evidenciando que arquiteturas consolidadas em outros domínios não garantem resultados superiores para este problema específico.

Quanto ao segundo objetivo específico, a análise comparativa quantitativa revelou padrões consistentes através das métricas de segmentação. A precisão média variou substancialmente entre arquiteturas (0,116 a 0,535), com FPN demonstrando capacidade superior de minimizar falsos positivos. A análise de custo-benefício, identificou que não existe correlação linear entre complexidade estrutural e desempenho. A configuração FPN + EfficientNet-B0, com apenas 5,8 milhões de parâmetros, superou significativamente modelos 4,2 vezes mais complexos. A homogeneidade nos tempos de treinamento contrasta com a ampla variação na velocidade de inferência, demonstrando que o custo temporal de desenvolvimento é desprezível, enquanto as características de inferência divergem substancialmente.

O terceiro objetivo específico foi atendido através da análise do comportamento de backbones em cenários de dados limitados. O ResNet34 demonstrou melhor desempenho médio, atribuível às conexões residuais que facilitam extração de características multiescala. Entretanto, a família EfficientNet demonstrou superioridade em eficiência computacional, com o EfficientNet-B0 oferecendo relação custo-benefício excepcional. Crucialmente, a análise revelou que a sinergia arquitetura-backbone é mais determinante que a capacidade individual de cada componente, exemplificado pelo PSPNet que alcançou Dice de 0,405 com ResNet34 versus apenas 0,093 com EfficientNet-B0.

Este estudo apresenta limitações, sabendo que o dataset de 100 radiografias pode limitar a generalização para populações com características radiográficas distintas. Estudos futuros deveriam avaliar o desempenho em *datasets* com maior diversidade demográfica. A análise focou em segmentação binária, enquanto a prática clínica demanda diferenciação

de estágios de progressão cariosa. Trabalhos subsequentes poderiam explorar segmentação multiclasse através de funções de perda apropriadas como *Focal Loss* ou *Generalized Dice Loss*. A velocidade de inferência foi quantificada em ambiente controlado com GPU dedicada, mas aplicações reais operam sob condições heterogêneas. Estudos prospectivos em ambientes clínicos reais permitiriam avaliar robustez a qualidade de imagem e variabilidade inter-operador.

Em via dos resultados apresentados, este trabalho demonstrou que a seleção fundamentada de arquiteturas, baseada em análise multidimensional integrando desempenho, eficiência e adequação ao problema, constitui fator determinante para o sucesso de sistemas de auxílio diagnóstico em odontologia digital. A superioridade consistente do FPN sugere que mecanismos de agregação piramidal hierárquica são particularmente adequados para lesões com variabilidade morfológica elevada.

Em última análise, o objetivo de desenvolver sistemas automatizados não visa substituir o julgamento clínico, mas fornecer ferramentas complementares que potencializem capacidades diagnósticas através da redução de variabilidade inter-observador e identificação automatizada de regiões suspeitas. A integração apropriada em fluxos clínicos reais demanda não apenas precisão algorítmica, mas também interfaces intuitivas, tempos de resposta compatíveis e transparência interpretativa.

REFERÊNCIAS

AHMED, Tashin; SABAB, Noor. Classification and Understanding of Cloud Structures via Satellite Images with EfficientUNet. **SN Computer Science**, v. 3, jan. 2022.

ALHARBI, Shuaa S; ALRUGAIBAH, Athbah A; ALHASSON, Haifa F; KHAN, Rehan Ullah. Detection of cavities from dental panoramic x-ray images using nested u-net models. **Applied Sciences**, MDPI, v. 13, n. 23, p. 12771, 2023.

ALHUSSEIN, G. *et al.* Cavity detection in panoramic dental X-ray images using U-Net models. **Applied Sciences**, v. 13, p. 12771, 2023.

BADRINARAYANAN, Vijay; KENDALL, Alex; CIPOLLA, Roberto. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. **IEEE transactions on pattern analysis and machine intelligence**, 2017.

BEAINI, Thiago Leite. **Padronização das tomadas radiográficas periapicais e panorâmicas com finalidade forense**. 2010. Tese (Doutorado) – Universidade de São Paulo.

BUSLAEV, Alexander; IGLOVIKOV, Vladimir I; KHVEDCHENYA, Eugene; PARINOV, Alex; DRUZHININ, Mikhail; KALININ, Alexandr A. Albumentations: fast and flexible image augmentations. **Information**, MDPI, v. 11, n. 2, p. 125, 2020.

CHEN, J. *et al.* TransUNet: Rethinking the U-Net architecture design for medical image segmentation through the lens of transformers. **Medical Image Analysis**, 2024.

CHEN, Liang-Chieh; PAPANDREOU, George; SCHROFF, Florian; ADAM, Hartwig. Rethinking atrous convolution for semantic image segmentation. **arXiv preprint arXiv:1706.05587**, 2017.

CHEN, Liang-Chieh; ZHU, Yukun; PAPANDREOU, George; SCHROFF, Florian; ADAM, Hartwig. Encoder-decoder with atrous separable convolution for semantic image segmentation. *In*: PROCEEDINGS of the European conference on computer vision (ECCV). [*S.l.*: *s.n.*], 2018. P. 801–818.

CHEN, X. *et al.* Improving Breast Cancer Detection and Diagnosis through Semantic Segmentation Using the Unet3+ Deep Learning Framework. **Diagnostics**, 2023.

COSTA, Simone de Melo; VASCONCELOS, Mara;
ABREU, Mauro Henrique Nogueira Guimarães. Impacto da cárie dentária na qualidade de vida de adultos residentes no entorno de Belo Horizonte, MG, Brasil. **Ciência & Saúde Coletiva**, SciELO Public Health, v. 18, p. 1971–1980, 2013.

DAYI, Burak; ÜZEN, Hüseyin *et al.* A Novel Deep Learning-Based Approach for Segmentation of Different Type Caries Lesions on Panoramic Radiographs. **Diagnostics**, 2023.

DIPU, Nadim; SHOHAN, Sifatul; K. M. A., Salam. Ocular Disease Detection Using Advanced Neural Network Based Classification Algorithms. **ASIAN JOURNAL OF CONVERGENCE IN TECHNOLOGY**, v. 7, p. 91–99, ago. 2021.

DURMUŞ, M. *et al.* Comparative Analysis of Pixel-Based Segmentation Models for Accurate Detection of Impacted Teeth on Panoramic Radiographs. **IEEE Access**, v. 13, 2025.

GARCIA-GARCIA, Alberto; ORTS-ESCOLANO, Sergio; OPREA, Sergiu; VILLENA-MARTINEZ, Victor; GARCIA-RODRIGUEZ, Jose. A review on deep learning techniques applied to semantic segmentation. **arXiv preprint arXiv:1704.06857**, 2017.

GEETHA, V; APRAMEYA, KS; HINDUJA, Dharam M. Dental caries diagnosis in digital radiographs using back-propagation neural network. **Health information science and systems**, Springer, v. 8, p. 1–14, 2020.

HE, Kaiming; GIRSHICK, Ross; DOLLÁR, Piotr. Rethinking imagenet pre-training. *In*: PROCEEDINGS of the IEEE/CVF International Conference on Computer Vision. [*S.l.*: *s.n.*], 2019.

HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing; SUN, Jian. Deep residual learning for image recognition. *In*: PROCEEDINGS of the IEEE conference on computer vision and pattern recognition. [*S.l.*: *s.n.*], 2016. P. 770–778.

HOU, S; ZHOU, T; LIU, Y; DANG, P; LU, H; SHI, H. Teeth U-Net: a segmentation model of dental panoramic X-ray images for context semantics and contrast enhancement. **Computers in Biology and Medicine**, Elsevier, v. 152, p. 106296, 2023.

HOWARD, Andrew G; ZHU, Menglong; CHEN, Bo; KALENICHENKO, Dmitry; WANG, Weijun; WEYAND, Tobias; ANDREETTO, Marco; ADAM, Hartwig.

Mobilenets: Efficient convolutional neural networks for mobile vision applications. **arXiv preprint arXiv:1704.04861**, 2017.

HUANG, Y.; LIU, W.; TANG, S. A multimodal dental dataset facilitating machine learning research and clinic services. **Scientific Data**, v. 11, p. 1291, 2024.

IBTEHAZ, Nabil; RAHMAN, M Sohel. MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation. **Neural networks**, Elsevier, v. 121, p. 74–87, 2020.

IMAK, A. *et al.* Detection of impacted teeth with improved U-Net architecture. **Oral Radiology**, 2021.

IMAK, A. *et al.* Detection of impacted teeth with improved U-Net architecture. **Oral Radiology**, v. 39, p. 614–628, 2021.

IMAK, Andac; CELEBI, Adalet; SIDDIQUE, Kamran; TURKOGLU, Muammer; SENGUR, Abdulkadir; SALAM, Iftekhar. Dental caries detection using score-based multi-input deep convolutional neural network. **IEEE Access**, IEEE, v. 10, p. 18320–18329, 2022.

IOFFE, Sergey; SZEGEDY, Christian. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *In*: PMLR. INTERNATIONAL conference on machine learning. [*S.l.*: *s.n.*], 2015. P. 448–456.

KOONCE, Brett. **Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization**. [*S.l.*: *s.n.*], jan. 2021. ISBN 978-1-4842-6167-5.

KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. Imagenet classification with deep convolutional neural networks. **Advances in neural information processing systems**, v. 25, 2012.

KUMAR, S. *et al.* A comparative study of breast tumour detection using a semantic segmentation network coupled with different pretrained CNNs. **Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization**, 2024.

LAI, Matthew. Deep learning for medical image segmentation. **arXiv preprint arXiv:1505.02000**, 2015.

LANGLAND, Olaf E; LANGLAIS, Robert P; PREECE, John W. **Principles of dental imaging**. [*S.l.*]: Lippincott Williams & Wilkins, 2002.

LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. **nature**, Nature Publishing Group UK London, v. 521, n. 7553, p. 436–444, 2015.

LEE, Jae-Hong; KIM, Do-Hyung; JEONG, Seong-Nyum; CHOI, Seong-Ho. Detection and diagnosis of dental caries using a deep learning-based convolutional neural network algorithm. **Journal of dentistry**, Elsevier, v. 77, p. 106–111, 2018.

LI, W. *et al.* Attention U-Net for caries segmentation. **Journal Name**, 2023.

LI, X. *et al.* Efficient few-shot medical image segmentation via self-supervised variational autoencoder. **Medical Image Analysis**, 2025.

LIAN, L; ZHU, T; ZHU, F; ZHU, H. Deep learning for caries detection and classification. **Diagnostics**, MDPI, v. 11, n. 9, p. 1672, 2021.

LIM, Dong-Jun; LUO, Si; KIM, Min-Hee; KO, Sun-Hee; KIM, Yongmin. Interobserver agreement and intraobserver reproducibility in thyroid ultrasound elastography. **American Journal of Roentgenology**, American Roentgen Ray Society, v. 198, n. 4, p. 896–901, 2012.

LIN, Tsung-Yi; DOLLÁR, Piotr; GIRSHICK, Ross; HE, Kaiming; HARIHARAN, Bharath; BELONGIE, Serge. Feature pyramid networks for object detection. *In*: PROCEEDINGS of the IEEE conference on computer vision and pattern recognition. [*S.l.*: *s.n.*], 2017. P. 2117–2125.

LIN, Tsung-Yi; GOYAL, Priya; GIRSHICK, Ross; HE, Kaiming; DOLLÁR, Piotr. Focal loss for dense object detection. *In*: PROCEEDINGS of the IEEE international conference on computer vision. [*S.l.*: *s.n.*], 2017. P. 2980–2988.

LIPTON, Zachary C; ELKAN, Charles; NARYANASWAMY, Balakrishnan. Optimal thresholding of classifiers to maximize F1 measure. *In*: SPRINGER. MACHINE Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part II 14. [*S.l.*: *s.n.*], 2014. P. 225–239.

LONG, Jonathan; SHELHAMER, Evan; DARRELL, Trevor. Fully convolutional networks for semantic segmentation. *In*: PROCEEDINGS of the IEEE conference on computer vision and pattern recognition. [*S.l.*: *s.n.*], 2015.

MADHIARASAN, Manogaran; LOUZAZNI, Mohamed. Analysis of artificial neural network: Architecture, types, and forecasting applications. **Journal of Electrical and Computer Engineering**, Wiley Online Library, v. 2022, n. 1, p. 5416722, 2022.

MATSUMOTO, M. *et al.* Publicly Available Dental Image Datasets for Artificial Intelligence. **Journal of Dental Research**, 2024.

MEI, Xueyan *et al.* RadImageNet: An Open Radiologic Deep Learning Research Dataset for Effective Transfer Learning. **Radiology: Artificial Intelligence**, 2022.

MÜLLER, Dominik; SOTO-REY, Iñaki; KRAMER, Frank. Towards a guideline for evaluation metrics in medical image segmentation. **BMC Research Notes**, Springer, v. 15, n. 1, p. 210, 2022.

NISHITANI, Y.; NAKAYAMA, R. *et al.* Segmentation of teeth in panoramic dental X-ray images using U-Net with a loss function weighted on the tooth edge. **Radiological Physics and Technology**, 2021.

O'SHEA, Keiron; NASH, Ryan. An introduction to convolutional neural networks. **arXiv preprint arXiv:1511.08458**, 2015.

OKAZAKI, Shota; MINE, Yuichi *et al.* RadImageNet and ImageNet as Datasets for Transfer Learning in the Assessment of Dental Radiographs: A Comparative Study. **Journal of Imaging Informatics in Medicine**, v. 38, n. 1, 2024.

ORGANIZATION, World Health *et al.* WHO highlights oral health neglect affecting nearly half of the world's population. **News Release**, 2022.

OZTEKIN, F. *et al.* An Explainable Deep Learning Model to Prediction Dental Caries. **Diagnostics**, 2023.

PANYARAK, W. *et al.* Automated detection and segmentation of dental caries using a novel cascaded learning approach. **Biomedical Signal Processing and Control**, 2024.

PARK, W.; HUH, J.K.; LEE, J.H. Automated deep learning for classification of dental implant radiographs using a large multi-center dataset. **Scientific Reports**, v. 13, p. 4862, 2023.

POWERS, David MW. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. **arXiv preprint arXiv:2010.16061**, 2020.

PRIYA, J.; RAJA, S. Caries-segnet: multi-scale cascaded hybrid spatial channel attention encoder-decoder for semantic segmentation of dental caries. **Biomedical Engineering / Biomedizinische Technik**, 2025.

PURWONO, Purwono; MA'ARIF, Alfian; RAHMANIAR, Wahyu; FATHURRAHMAN, Haris Imam Karim; FRISKY, Aufaclav Zatu Kusuma; HAQ, Qazi Mazhar ul. Understanding of convolutional neural network (cnn): A review. **International Journal of Robotics and Control Systems**, v. 2, n. 4, p. 739–748, 2022.

QAYYUM, A.; TAHIR, A.; BUTT, M. A. *et al.* Dental caries detection using a semi-supervised learning approach. **Scientific Reports**, 2023.

RAMACHANDRAN, Prajit; ZOPH, Barret; LE, Quoc V. Searching for activation functions. **arXiv preprint arXiv:1710.05941**, 2017.

REDMON, Joseph; DIVVALA, Santosh; GIRSHICK, Ross; FARHADI, Ali. You only look once: Unified, real-time object detection. *In*: PROCEEDINGS of the IEEE conference on computer vision and pattern recognition. [*S.l.*: *s.n.*], 2016. P. 779–788.

REZATOFIGHI, Hamid; TSOI, Nathan; GWAK, JunYoung; SADEGHIAN, Amir; REID, Ian; SAVARESE, Silvio. Generalized intersection over union: A metric and a loss for bounding box regression. *In*: PROCEEDINGS of the IEEE/CVF conference on computer vision and pattern recognition. [*S.l.*: *s.n.*], 2019. P. 658–666.

RODRIGUES, Raires Chaves da Silva *et al.* Análise econômica das técnicas para tratamento periodontal não cirúrgico e o seu impacto na saúde pública. Universidade Federal da Paraíba, 2024.

RONNEBERGER, Olaf; FISCHER, Philipp; BROX, Thomas. U-net: Convolutional networks for biomedical image segmentation. *In*: SPRINGER. INTERNATIONAL

Conference on Medical image computing and computer-assisted intervention. [*S.l.*: *s.n.*], 2015. P. 234–241.

SAENZ, A. *et al.* **Medical AI Data for All (MAIDA)**. [*S.l.*: *s.n.*], 2024.

SCHWENDICKE, Falk; SINGH, Tarry; LEE, Jae-Hong; GAUDIN, Robert; CHAURASIA, Akhilanand; WIEGAND, Thomas; URIBE, Sergio; KROIS, Joachim *et al.* Artificial intelligence in dental research: Checklist for authors, reviewers, readers. **Journal of dentistry**, Elsevier, v. 107, p. 103610, 2021.

SETIO, A. *et al.* A Benchmark Study of DeepLabV3+, U-Net++, and Attention U-Net for Blood Cell Segmentation. **Conference Proceedings**, 2024.

SHAMIR, Reuben R; DUCHIN, Yuval; KIM, Jinyoung; SAPIRO, Guillermo; HAREL, Noam. Continuous dice coefficient: a method for evaluating probabilistic segmentations. **arXiv preprint arXiv:1906.11031**, 2019.

SIVAGAMI, S; CHITRA, P; KAILASH, GSR; MURALIDHARAN, SR. UNet architecture based dental panoramic image segmentation. *In*: IEEE. 2020 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET). [*S.l.*: *s.n.*], 2020. P. 187–191.

SUN, Duolin; WANG, Jianqing *et al.* STS-TransUNet: Semi-supervised Tooth Segmentation Transformer U-Net for dental panoramic image. **Mathematical Biosciences and Engineering**, 2024.

TAN, Chuanqi; SUN, Fuchun; KONG, Tao; ZHANG, Wenchang; YANG, Chao; LIU, Chunfang. A survey on deep transfer learning. *In*: SPRINGER. ARTIFICIAL Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27. [*S.l.*: *s.n.*], 2018. P. 270–279.

TAN, Mingxing; LE, Quoc. Efficientnet: Rethinking model scaling for convolutional neural networks. *In*: PMLR. INTERNATIONAL conference on machine learning. [*S.l.*: *s.n.*], 2019. P. 6105–6114.

TASSOKER, M.; ÖZIÇ, M.Ü.; YUCE, F. Performance evaluation of a deep learning model for automatic detection and localization of idiopathic osteosclerosis on dental panoramic radiographs. **Scientific Reports**, v. 14, p. 4437, 2024.

THOMA, Martin. A survey of semantic segmentation. **arXiv preprint arXiv:1602.06541**, 2016.

TONG, Zheng; XU, Philippe; DENOEUX, Thierry. Evidential fully convolutional network for semantic segmentation. **Applied Intelligence**, Springer, v. 51, n. 9, p. 6376–6399, 2021.

TULSANI, Akshat; KUMAR, Preetham; PATHAN, Sumaiya. Automated segmentation of optic disc and optic cup for glaucoma assessment using improved UNET++ architecture. **Biocybernetics and Biomedical Engineering**, Elsevier, v. 41, n. 2, p. 819–832, 2021.

VERMA, R.; MOSTAFID, T. Tumor Segmentation with U-Net and DeepLabv3+: A Review. **Medical Image Analysis**, 2024.

WANG, D. *et al.* A dual-labeled dataset and fusion model for automatic teeth segmentation, numbering, and state assessment on panoramic radiographs. **BMC Oral Health**, 2024.

WANG, Xianyun; GAO, Sizhe; JIANG, Kaisheng; ZHANG, Huicong; WANG, Linhong; CHEN, Feng; YU, Jun; YANG, Fan. Multi-level uncertainty aware learning for semi-supervised dental panoramic caries segmentation. **Neurocomputing**, Elsevier, v. 540, p. 126208, 2023.

WHITE, Stuart C; PHAROAH, Michael J. **Oral radiology: principles and interpretation**. [*S.l.*]: Elsevier Health Sciences, 2013.

WITHEY, Daniel J; KOLES, Zoltan J. A review of medical image segmentation: methods and available software. **International Journal of Bioelectromagnetism**, v. 10, n. 3, p. 125–148, 2008.

WORLD HEALTH ORGANIZATION. **Global Initiative on AI for Health (GI-AI4H)**. [*S.l.*: *s.n.*], 2023.

WU, Jianxin. Introduction to convolutional neural networks. **National Key Lab for Novel Software Technology. Nanjing University. China**, v. 5, n. 23, p. 495, 2017.

YANG, Kwang-Tzu. Artificial neural networks (ANNs): a new paradigm for thermal science and engineering, 2008.

YASSAR, M. K. A.; FITRIA, M. *et al.* The Role of U-Net Segmentation for Enhancing Deep Learning-based Dental Caries Classification. **Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics**, 2024.

YIN, Xiao-Xia *et al.* U-Net-Based Medical Image Segmentation. **Journal of Healthcare Engineering**, 2022.

ZHANG, Hao; ZHU, Chenyang; LIAN, Xuegan; HUA, Fei. A Nested attention guided UNet++ architecture for white matter hyperintensity segmentation. **IEEE Access**, IEEE, v. 11, p. 66910–66920, 2023.

ZHANG, Y *et al.* Children's dental panoramic radiographs dataset for caries segmentation and dental disease detection. **Scientific Data**, Nature Publishing Group, v. 10, n. 1, p. 1–11, 2023.

ZHAO, Hengshuang; SHI, Jianping; QI, Xiaojuan; WANG, Xiaogang; JIA, Jiaya. Pyramid scene parsing network. *In*: PROCEEDINGS of the IEEE conference on computer vision and pattern recognition. [*S.l.*: *s.n.*], 2017. P. 2881–2890.

ZHAO, Y. *et al.* TSASNet: Tooth segmentation on dental panoramic X-ray images by Two-Stage Attention Segmentation Network. **Journal Name**, 2024.

ZHOU, Y. *et al.* Universal Lesion Segmentation Challenge 2023: A Comparative Research of Different Algorithms. **arXiv preprint**, 2024.

ZHOU, Zongwei; RAHMAN SIDDIQUEE, Md Mahfuzur; TAJBAKHSH, Nima; LIANG, Jianming. Unet++: A nested u-net architecture for medical image segmentation. *In*: SPRINGER. DEEP learning in medical image analysis and multimodal learning for clinical decision support: 4th international workshop, DLMIA 2018, and 8th international workshop, ML-CDS 2018, held in conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, proceedings 4. [*S.l.*: *s.n.*], 2018. P. 3–11.

ZHU, Haihua; CAO, Zheng; LIAN, Luya *et al.* CariesNet: a deep learning approach for segmentation of multi-stage caries lesion from oral panoramic X-ray image. **Neural Computing and Applications**, 2021.

ZHU, J.; ZENG, L.; MO, Z. *et al.* LMCD-OR: a large-scale, multilevel categorized diagnostic dataset for oral radiography. **Journal of Translational Medicine**, v. 22, p. 930, 2024.