UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
ENGENHARIA ELETRÔNICA E DE TELECOMUNICAÇÕES

MATEUS FLAUSINO DE ARAÚJO

ESTIMAÇÃO DA RELAÇÃO SINAL RUÍDO ÓPTICA BASEADA EM APRENDIZADO PROFUNDO UTILIZANDO DIAGRAMAS DE CONSTELAÇÃO DAS MODULAÇÕES DP-mPSK E DP-mQAM EM RECEPTORES ÓPTICOS COERENTES FLEXÍVEIS

UBERLÂNDIA
2026

MATEUS FLAUSINO DE ARAÚJO

ESTIMAÇÃO DA RELAÇÃO SINAL RUÍDO ÓPTICA BASEADA EM APRENDIZADO PROFUNDO UTILIZANDO DIAGRAMAS DE CONSTELAÇÃO DAS MODULAÇÕES DP-mPSK E DP-mQAM EM RECEPTORES ÓPTICOS COERENTES FLEXÍVEIS

Trabalho apresentado como um dos requisitos parciais para aprovação na disciplina de Trabalho de Conclusão de Curso do curso de Engenharia Eletrônica e de Telecomunicações da Universidade Federal de Uberlândia – Campus Uberlândia.

Orientador: Prof. Dr. André Luiz Aguiar da Costa

Co-orientador: Dr. Myke Douglas de Medeiros Valadão

UBERLÂNDIA
2026

MATEUS FLAUSINO DE ARAÚJO

ESTIMAÇÃO DA RELAÇÃO SINAL RUÍDO ÓPTICA BASEADA EM APRENDIZADO PROFUNDO UTILIZANDO DIAGRAMAS DE CONSTELAÇÃO DAS MODULAÇÕES DP-mPSK E DP-mQAM EM RECEPTORES ÓPTICOS COERENTES FLEXÍVEIS

Trabalho apresentado como um dos requisitos parciais para aprovação na disciplina de Trabalho de Conclusão de Curso I do curso de Engenharia Eletrônica e de Telecomunicações da Universidade Federal de Uberlândia – Campus Uberlândia.

Uberlândia, 16 de março de 2026.

COMISSÃO EXAMINADORA:

Prof. Dr. André Luiz Aguiar da Costa

Universidade Federal de Uberlândia

Orientador

Dra. Indayara Bertoldi Martins

Ekinops, Lannion França

Examinadora

Prof. Dr. Éderson Rosa da Silva

Universidade Federal de Uberlândia

Examinador

Resumo

A Relação Sinal Ruído Óptica (OSNR, *Optical Signal Noise Ratio*) é um parâmetro crítico para o funcionamento, gestão e manutenção de redes de comunicação óptica coerentes. Dada a sua importância, este trabalho propõe e avalia uma abordagem baseada em aprendizagem profunda (DL, *deep learning*) para a estimação da OSNR a partir de diagramas de constelação. Para o desenvolvimento dos modelos, gerou-se um conjunto de dados diversificado, composto por mais de 19.000 imagens, através da simulação de sinais com formatos de modulação DP m-PSK e DP m-QAM sob diferentes configurações visuais. No total, foram avaliadas 15 arquiteturas de Redes Neurais Convolucionais (CNNs, *Convolutional Neural Networks*) representativas do estado da arte, incluindo as famílias *MobileNetV3*, *ConvNeXt*, *DenseNet* e *EfficientNet*. A arquitetura *ConvNeXtBase* alcançou o melhor desempenho preditivo do estudo, com um Erro Absoluto Médio (MAE, *Mean Absolute Error*) inferior a 0,43 dB e um coeficiente de determinação (R^2) superior a 0,98. Os resultados obtidos confirmam o elevado potencial das técnicas de visão computacional para a monitorização contínua e não intrusiva de desempenho em redes óticas de próxima geração.

Palavras-chaves: aprendizado profundo, diagrama de constelação, estimação, OSNR, redes ópticas coerentes.

Abstract

The Optical Signal to Noise Ratio (OSNR) is a critical parameter for the operation, management, and maintenance of coherent optical communication networks. Given its importance, this work proposes and evaluates a deep learning (DL) based approach for OSNR estimation from constellation diagrams. For the development of the models, a diverse dataset comprising over 19,000 images was generated through the simulation of signals with DP m-PSK and DP m-QAM modulation formats under different visual configurations. In total, 15 representative state-of-the-art Convolutional Neural Network (CNN) architectures were evaluated, including the MobileNetV3, ConvNeXt, DenseNet, and EfficientNet families. The ConvNeXtBase architecture achieved the best predictive performance in the study, with a Mean Absolute Error (MAE) below 0.43 dB and a coefficient of determination (R^2) above 0.98. The obtained results confirm the high potential of computer vision techniques for the continuous and non-intrusive performance monitoring of next-generation optical networks.

Keywords: coherent optical communications, constellation diagrams, deep learning, OSNR.

Agradecimentos

Agradeço antes de tudo e todos, a Deus, que me sustentou e me guiou, dando saúde, sabedoria e conhecimento.

Aos meus pais, José e Clara e meus irmãos, Lauro e Otávio, que foram suporte e inspiração em todos os momentos sendo fundamentais em todo o processo.

Agradeço ao meu orientador Prof. Dr. André Luiz Aguiar da Costa por todos os ensinamentos enquanto professor e pela orientação na pesquisa e publicação.

Agradeço ao meu co-orientador Dr. Myke Douglas de Medeiros Valadão pela orientação na pesquisa e publicação.

Agradeço ao Prof. Dr. Waldir Sabino da Silva Júnior pela parceria na publicação do artigo no XLIII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais em 2025.

Agradeço à Universidade Federal de Uberlândia (UFU) por proporcionar um ensino de qualidade e acessível.

Agradeço à Universidade Federal do Amazonas (UFAM) pelo apoio nas pesquisas e publicação.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de abreviaturas e siglas

Sigla	Definição (*Definição em Inglês*)
AI	Inteligência Artificial (*Artificial Intelligence*)
ASE	Emissão Espontânea Amplificada (*Amplified Spontaneous Emission*)
BBU	Unidade de Processamento Banda Base (*Baseband Unit*)
CAGR	Taxa de Crescimento Anual Composta (*Compound Annual Growth Rate*)
CD	Dispersão Cromática (*Chromatic dispersion*)
CNN	Redes Neurais Convolucionais (*Convolutional Neural Networks*)
DCI	Interconexão de Datacenters (*Data Center Interconnect*)
DL	Aprendizado Profundo (*Deep Learning*)
DP m-PSK	Chaveamento por Desvio de Fase Dupla Polarização (*Dual-Polarization Phase shift Keying*)
DP m-QAM	Modulação de Amplitude em Quadratura Dupla Polarização (*Dual-Polarization Quadrature Amplitude Modulation*)
DP MZM	Modulador Mach-Zehnder de Dupla Polarização (*Dual-Polarization Mach-Zehnder Modulator*)
DPI	Pontos por Polegada (*Dots Per Inch*)
DSF	Fibras de dispersão deslocada (*Dispersion-Shifted Fibers*)
DSP	Processamento Digital de Sinais (*Digital Signal Processing*)
EDFA	Amplificadores de Fibra Dopada com Érbio (*Erbium-Doped Fiber Amplifiers*)
EON	Redes Ópticas Elásticas (*Elastic Optical Networks*)
FTTH	Fibra até a Casa (*Fiber to the Home*)
FWM	Mistura de Quatro Ondas (*Four-Wave Mixing*)
GaAs	Arsenieto de Gálio
InGaAsP	Índio-Gálio-Arsênio-Fósforo
ITU	União Internacional de Telecomunicações (*International Telecommunication Union*)

MAE	Erro Absoluto Médio (*Mean Absolute Error*)
ML	Aprendizado de Máquina (*Machine Learning*)
MM	Multimodo (*Multimode*)
MSE	Erro Quadrático Médio (*Mean Squared Error*)
NAS	Busca de Arquitetura Neural (*Neural Architecture Search*)
OBPF	Filtro Passa-banda Óptico (*Optical Band-pass Filter*)
ONSR	Relação Sinal Ruído Óptica (*Optical Signal Noise Ratio*)
OSA	Analisadores de Espectro Óptico (*Optical Spectrum Analyzer*)
PMD	Dispersão por Modo de Polarização (*Polarization Mode Dispersion*)
PON	Redes Ópticas Passivas (*Passive Optical Network*)
QAM	Modulação de Amplitude em Quadratura (*Quadrature Amplitude Modulation*)
QPSK	Chaveamento por Deslocamento de Fase em Quadratura (*Quadrature Phase-Shift Keying*)
R^2	Coeficiente de Determinação
ReLU	Unidade Linear Retificada (*Rectified Linear Unit*)
RMSE	Raiz do Erro Quadrático Médio (*Root Mean Squared Error*)
RRU	Unidade de Rádio Remota (*Remote Radio Unit*)
SDN	Redes Definidas por Software (*Software-Defined Networking*)
SGD	Gradiente Descendente Estocástico (*Stochastic Gradient Descent*)
SM	Monomodo (*Singlemode*)
SPM	Automodulação de Fase (*Self-Phase Modulation*)
TIAs	Amplificadores de Transimpedância (*Transimpedance Amplifiers*)
VOA	Atenuador Óptico Variável (*Variable Optical Attenuator*)
WAN	Rede de Longa Distância (*Wide Area Network*)
WDM	Multiplexação por Divisão de Comprimento de Onda (*Wavelength-Division Multiplexing*)
XPM	Modulação de Fase Cruzada (*Cross-Phase Modulation*)

Capítulo 1

Introdução

De acordo com as projeções da União Internacional de Telecomunicações (ITU, *International Telecommunication Union*) para 2024, o número de usuários de Internet alcançou 5,5 bilhões de pessoas, o que corresponde a cerca de 68% da população mundial. Esse número representa um aumento de 227 milhões de pessoas em relação às estimativas revisadas para 2023. Além disso, no mesmo ano, as redes móveis de quinta geração (5G) cobriram 51% da população global [1].

O crescimento no acesso e na tecnologia é acompanhado por um aumento expressivo no volume de dados trafegados. A ITU projeta que, em 2024, o tráfego global de banda larga móvel aproximou-se de 1,3 Zettabytes, enquanto o tráfego de banda larga fixa alcançou cerca de 6 Zettabytes. Tais números representam um aumento substancial em relação a 2023, quando os volumes foram estimados em 1 e 5,1 Zettabytes para as bandas largas móveis e fixa respectivamente [1].

Para além das estimativas anuais, os relatórios de projeção de longo prazo da indústria confirmam e detalham essa tendência de crescimento exponencial [2][3]. Segundo o relatório de um grande fabricante, há uma projeção de aumento exponencial no tráfego de rede de longa distância (WAN, *Wide Area Network*) global, com um crescimento estimado entre cinco e nove vezes na década entre 2023 e 2033. Em cenários disruptivos, o tráfego pode atingir um pico mensal de 6641 Exabytes em 2033 [2]. Num estudo mais aprofundado sobre o tráfego da rede móvel, o relatório da Ericsson de 2025 prevê um volume total de 430 Exabytes mensais até 2030, impulsionado por uma taxa de crescimento anual composta (CAGR, *Compound Annual Growth Rate*) de 17% [3].

Diante deste cenário, a comunicação óptica é a tecnologia fundamental que sustenta a infraestrutura de rede global. Sua capacidade de transmitir volumes massivos de dados simultaneamente, graças à sua vasta largura de banda, responde diretamente à demanda gerada pelo mundo conectado. Além disso, ela permite a transferência de dados por longas distâncias com mínima perda de sinal e oferece alta confiabilidade, dada a sua imunidade a interferências eletromagnéticas, consolidando-se como *backbone* indispensável das redes de telecomunicações de grandes provedores [4][5].

Essa infraestrutura crítica materializa-se em todas as escalas operacionais da rede. Em escala global, ela é materializada pelos cabos submarinos, responsáveis por transportar aproximadamente 99% de todo o tráfego de comunicações internacionais [6]. Em escala terrestre, a fibra óptica constitui o *backbone* de longa distância (*long-haul*) e as redes de Interconexão de Datacenters (DCI, *Data Center Interconnect*), que conectam localidades entre cidades, estados e países [7].

Na escala de acesso, que atende diretamente o usuário final, a dependência da fibra é igualmente crítica, manifestando-se de duas formas principais. Primeiramente, ela é a infraestrutura essencial para redes móveis 5G, sendo utilizada tanto no *backhaul*, a conexão da estação base à rede principal, quanto no *fronthaul*, o link crítico de latência ultrabaixa entre a unidade de rádio remota (RRU, *Remote Radio Unit*) e a unidade de processamento de banda base (BBU, *Baseband Unit*) [8]. Em segundo lugar, ela domina o acesso fixo, onde tecnologias como as Redes Ópticas Passivas (PON, *Passive Optical Network*) se consolidaram mundialmente para viabilizar a fibra até as casas (FTTH, *Fiber to the Home*) [9].

A atual predominância da tecnologia óptica é resultado de décadas de evolução contínua. O desenvolvimento histórico dos sistemas de comunicação óptica é frequentemente categorizado pela literatura em gerações, marcadas por avanços tecnológicos específicos iniciados na década de 1970. A primeira geração, introduzida comercialmente em 1980 após testes de campo realizados entre 1977 e 1979, operava na janela de comprimento de onda de 850 nm. Utilizando lasers semicondutores de Arsenieto de Gálio (GaAs), esses sistemas atingiam taxas de transmissão de 45 Mbit/s, com espaçamento entre repetidores limitado a cerca de 10 km, devido à atenuação das fibras da época ser superior a 3 dB/km [10][11].

Estabelecida a partir de 1985, a segunda geração caracterizou-se pela transição das fibras multimodo (MM, *multimode*) — com diâmetro de núcleo de 50 µm — para fibras monomodo (SM, *singlemode*), cujo núcleo reduzido de aproximadamente 9 µm permitiu mitigar a dispersão intra modal. Concomitantemente, houve o deslocamento da janela de operação para o comprimento de onda de 1310 nm, utilizando lasers baseados em semicondutores de InGaAsP (Índio-Gálio-Arsênio-Fósforo). Essas evoluções tecnológicas possibilitaram a extensão da distância entre repetidores para até 40 km, com taxas de transmissão superando 2 Gbps [10][11].

O avanço subsequente foi impulsionado pela necessidade de reduzir ainda mais a atenuação, marcando a migração da operação para a janela de 1550 nm. Essa terceira fase aproveitou a região onde a fibra de sílica apresenta sua perda mínima (cerca de 0,25 dB/km), mas enfrentou o desafio da elevada dispersão cromática. A solução exigiu o desenvolvimento de duas tecnologias fundamentais: a utilização de fibras de dispersão deslocada (DSF, *Dispersion-Shifted Fibers*) e a adoção de lasers operando em modo longitudinal único (*Single Longitudinal Mode*). A combinação desses avanços permitiu que, no início da década de 1990, fossem estabelecidos sistemas com alcance de até 100 km e taxas de transmissão de 10 Gbps [10][11].

Já a introdução da Multiplexação por Divisão de Comprimento de Onda (WDM, *Wavelength-Division Multiplexing*) definiu a quarta geração, representando uma verdadeira mudança de paradigma ao maximizar a utilização da banda passante combinando múltiplos canais ópticos em uma única fibra. Simultaneamente, o desenvolvimento dos Amplificadores de Fibra Dopada com Érbio (EDFAs, *Erbium-Doped Fiber Amplifiers*) eliminou a necessidade de regeneração optoeletrônica, permitindo a amplificação direta no domínio óptico. A sinergia entre WDM e EDFAs impulsionou a capacidade dos sistemas a novos patamares, viabilizando, por exemplo, um experimento de transmissão de 11 Tbps em uma distância de 117 km no ano de 2001 [10][11].

A próxima evolução, caracterizando a quinta geração, foi marcada pela viabilidade comercial dos transceptores coerentes, capazes de recuperar tanto a informação de amplitude quanto a de fase do sinal óptico. Esse avanço tecnológico permitiu a implementação de formatos de modulação mais complexos e eficientes, como o Chaveamento por Deslocamento de Fase em Quadratura (QPSK, *Quadrature Phase-Shift Keying*) e a Modulação de Amplitude em Quadratura (QAM, *Quadrature Amplitude Modulation*). A adoção dessas técnicas elevou significativamente a eficiência espectral, possibilitando a transmissão de uma maior quantidade de bits dentro da mesma largura de banda. Como reflexo desses avanços, estabeleceu-se um novo marco em 2011, com a demonstração de uma transmissão de 64 Tbps ao longo de 320 km, agregando 640 canais WDM [11].

Entretanto, a sofisticação introduzida por essas tecnologias, notadamente os sistemas coerentes, resultou em um aumento exponencial na complexidade de gerenciamento das redes. O número de parâmetros que demandam ajustes dinâmicos e

precisos — tais como formato de modulação, taxa de símbolo, codificação adaptativa e espaçamento flexível entre canais — tornou a otimização manual ou baseada em regras estáticas um desafio considerável. É neste cenário que o Aprendizado de Máquina (ML, *Machine Learning*) desperta grande interesse na área de redes de comunicação, oferecendo a capacidade de automatizar tarefas complexas de projeto e operação através do processamento de dados e aprendizado de padrões [12].

Essa complexidade operacional é agravada por três fatores adicionais intrínsecos às redes modernas. O primeiro reside na natureza física do canal de fibra óptica, sujeito a distorções lineares, como a Dispersão Cromática (CD, *Chromatic dispersion*) e a Dispersão por Modo de Polarização (PMD, *Polarization Mode Dispersion*), e a efeitos não lineares de difícil predição, como a Automodulação de Fase (SPM, *Self-Phase Modulation*), a Modulação de Fase Cruzada (XPM, *Cross-Phase Modulation*) e a Mistura de Quatro Ondas (FWM, *Four-Wave Mixing*). O segundo fator é a adoção de Redes Ópticas Elásticas (EON, *Elastic Optical Networks*), que, embora permitam uma alocação de espectro mais eficiente, elevam a dificuldade de gerenciamento de recursos. Por fim, o terceiro desafio advém do controle dinâmico via Redes Definidas por Software (SDN, *Software-Defined Networking*), onde reconfigurações em tempo real podem introduzir instabilidades ou problemas de desempenho imprevistos, dificultando a determinação do design ótimo da rede pelos engenheiros [12].

Diante desses obstáculos, a implementação de ML constitui uma transformação significativa na operação de redes ópticas. A estratégia envolve o uso de algoritmos inteligentes para examinar a grande quantidade de dados obtidos pelos monitores de rede, relacionando padrões de tráfego e métricas de qualidade de sinal. Assim, o sistema adquire a capacidade de "aprender" o comportamento complexo da infraestrutura, permitindo a tomada de decisões autônomas e preditivas, como prever falhas e calcular rotas otimizadas, superando as restrições dos métodos determinísticos convencionais [12].

1.1 Problematização

Os métodos tradicionais para a estimação da Relação Sinal Ruído Óptica (OSNR, *Optical Signal Noise Ratio*) enfrentam limitações significativas diante da evolução das redes ópticas atuais. A técnica que utiliza Analisadores de Espectro Óptico (OSA, *Optical Spectrum Analyzer*) para comparar a potência do sinal com o ruído, torna-se um desafio para ser aplicada em larga escala, principalmente devido ao alto custo.

Além disso, a implementação de redes densas ou elásticas aumentou a complexidade da medição, uma vez que a filtragem do ruído é dificultada pela extrema proximidade entre os canais, demandando maior treinamento dos operadores e equipamentos mais robustos [13].

Paralelamente, as abordagens baseadas em métodos estatísticos convencionais, que analisam os momentos da distribuição nos diagramas de constelação, exigem a aquisição de um elevado volume de informações de cada símbolo para garantir confiabilidade. Essa alta complexidade computacional e a necessidade de equipamentos de alto custo tornam esses métodos desafiadores para aplicações em tempo real, inviabilizando a implementação eficiente de sistemas autônomos e operações de rede inteligentes [13], [14].

1.2 Objetivo Geral

Este trabalho tem como objetivo principal desenvolver e avaliar um método de estimação da OSNR, baseado em aprendizado de máquinas profundo, utilizando como base de dados os diagramas de constelação de sinais com modulações DP m-PSK (*Dual-Polarization Phase-shift Keying*) e DP m-QAM (*Dual-Polarization Quadrature Amplitude Modulation*) em receptores ópticos coerentes flexíveis.

1.2.1 Objetivos Específicos

Para cumprir o objetivo geral, foram definidos os seguintes objetivos específicos:

- Revisar a literatura sobre os conceitos de receptores ópticos coerentes flexíveis.
- Revisar a literatura sobre os conceitos teóricos de aprendizado de máquinas profundo.
- Gerar um conjunto de dados por meio de simulações de sinais ópticos coerentes, contemplando diversos formatos de modulação (DP m-PSK e DP m-QAM) sob uma ampla faixa de valores de OSNR.
- Extrair e processar os diagramas de constelação dos sinais gerados, convertendo-os em um formato de imagem adequado para o treinamento de modelos de visão computacional.
- Desenvolver rotinas computacionais para o pré-processamento, a organização e o aumento de dados (*data augmentation*) do conjunto de

imagens, a fim de otimizar o treinamento e a capacidade de generalização dos modelos.

- Implementar, treinar e validar múltiplas arquiteturas de Redes Neurais Convolucionais (CNNs) para a tarefa de regressão da OSNR.
- Determinar e comparar o desempenho dos modelos de CNNs utilizando métricas de regressão para identificar a arquitetura de maior eficácia.

1.3 Justificativa

A relevância deste estudo fundamenta-se no papel crítico da OSNR para a eficiência das redes de próxima geração, que operam cada vez mais próximas aos seus limites físicos de capacidade. Atualmente, o monitoramento dessa métrica impõe um dilema operacional: equipamentos dedicados, como OSA, são imprecisos para implantação em redes densas ou elásticas, enquanto métodos estatísticos convencionais muitas vezes carecem da agilidade necessária para o controle dinâmico da rede.

Nesse cenário, a proposta de aplicar Aprendizado Profundo (DL, *Deep Learning*) sobre imagens de diagramas de constelação surge como uma solução capaz de superar essas barreiras. Ao abordar a estimação de OSNR através da Visão Computacional, torna-se possível realizar um monitoramento preciso e não intrusivo oferecendo uma alternativa escalável que viabiliza a implementação de redes ópticas autônomas.

1.4 Proposta e Organização do Trabalho

Neste trabalho propõe-se uma abordagem baseada em aprendizagem profundo para estimação de OSNR a partir de diagramas de constelação gerados em uma configuração flexível de receptor óptico coerente.

A organização do trabalho está dividida em cinco capítulos. O Capítulo 2 apresenta a fundamentação teórica sobre redes neurais e aprendizado profundo. O Capítulo 3 detalha a metodologia aplicada na pesquisa. O Capítulo 4 expõe e discute os resultados obtidos, enquanto o Capítulo 5 sintetiza as conclusões e elenca sugestões para estudos futuros.

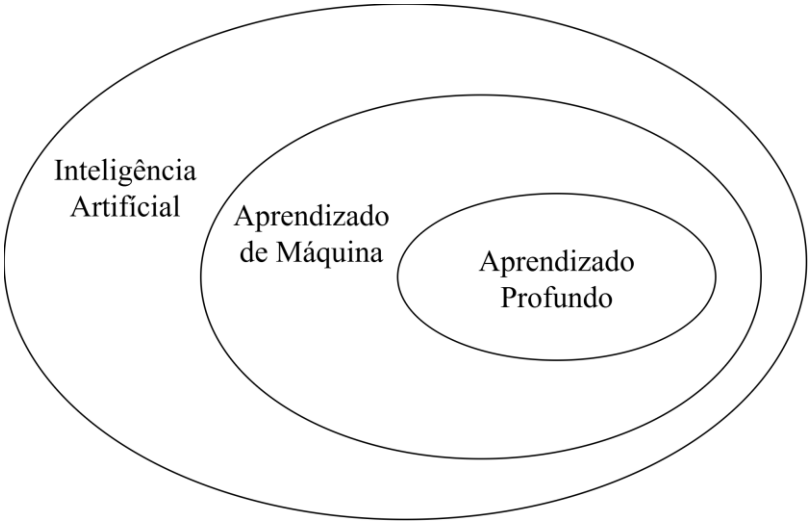
Capítulo 2

Redes neurais e aprendizagem profunda

2.1 Inteligência Artificial e seus subconjuntos

Inicialmente, é fundamental estabelecer a relação hierárquica entre Inteligência Artificial (AI, *Artificial Intelligence*), ML e DL[15]. A Figura 1 ilustra a estrutura em que estes domínios operam de maneira concêntrica demonstrando que o Aprendizado Profundo é um subconjunto especializado do Aprendizado de Máquina que, por sua vez, constitui uma área abrangente dentro da Inteligência Artificial [16].

Figura 1: Classificação inteligência artificial, aprendizado de máquina e aprendizado profundo



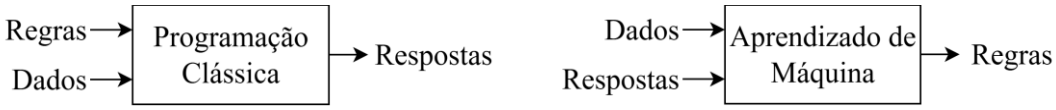
Fonte [15]

A AI pode ser definida como a área dedicada a criar máquinas aptas a executar atividades intelectuais que, tradicionalmente, exigiriam a cognição humana. Trata-se de um campo vasto e multidisciplinar: ele abarca desde a automação rígida e sistemas fundamentados em regras lógicas até fronteiras mais dinâmicas, como a robótica e o próprio aprendizado de máquina [15].

Este último, o ML representa uma verdadeira ruptura de paradigma. Na programação clássica ilustrado na Figura 2 (a) o fluxo é dedutivo: humanos codificam regras explícitas que, ao processarem dados, geram respostas. O ML inverte essa lógica: o sistema é alimentado com os dados e as respostas esperadas e, por meio do treinamento, infere as regras matemáticas que correlacionam um ao outro, conforme Figura 2 (b).

Portanto, em vez de ser programada para realizar uma tarefa passo a passo, a máquina é treinada para aprender a executá-la [15].

Figura 2:(a) Programação clássica (b) *Machine Learning*



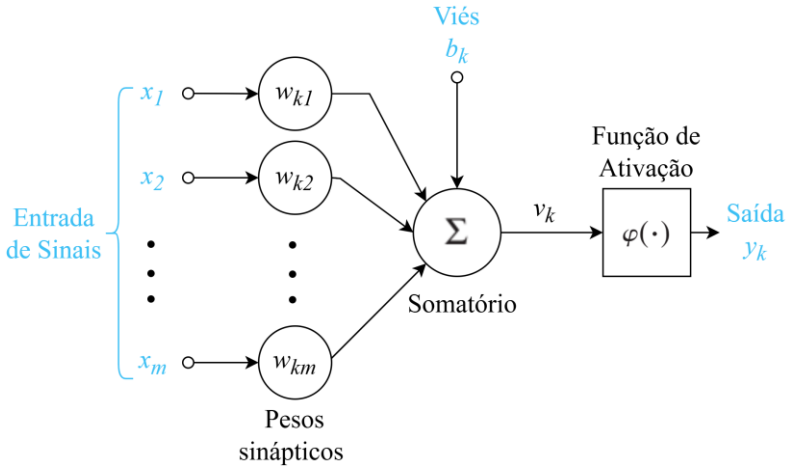
Fonte [15]

Assim, no ML o aprendizado ocorre a partir de exemplos contidos nos dados, conjunto frequentemente chamado na literatura de *dataset*. A presença ou ausência de respostas nesses dados define duas categorias principais de aprendizado: o supervisionado e o não supervisionado. No aprendizado supervisionado, cada exemplo no *dataset* possui um rótulo associado, indicando qual é a resposta correta. Já no caso não supervisionado, os dados não possuem rótulos; cabe ao próprio algoritmo identificar padrões ocultos e agrupar as informações com base na similaridade entre elas [16][17].

2.2 Neurônio Artificial

Para compreender as técnicas de aprendizado de máquina é fundamental entender o conceito de redes neurais artificiais. Essas redes foram inspiradas na arquitetura biológica dos cérebros humanos que possuem bilhões de neurônios e trilhões de conexões entre eles [18]. As redes neurais artificiais são compostas por unidades computacionais chamadas de neurônio artificial (ou *perceptron* em algumas literaturas), que possuem entradas x_m com pesos w_{km} associados, formando uma soma ponderada acrescida de um termo viés b_k (*bias*). A saída v_k da soma alimenta uma função de ativação, resultando no valor y_k, conforme ilustrado na Figura 3 [18][19].

Figura 3: Representação do neurônio artificial



Fonte [18]

Pode-se expressar matematicamente a saída do neurônio artificial pelas Equações (2.1) e (2.2),

$$v_k = \sum_{j=1}^{m} w_{kj} x_j + b_k \tag{2.1}$$

$$y_k = \varphi(v_k) \tag{2.2}$$

sendo m representa o número total de entradas, w_{kj} é o peso sináptico conectando a entrada j ao neurônio k, x_k é o sinal de entrada, b_k é o viés (*bias*) e $\varphi(\cdot)$ representa a função de ativação.

2.3 Funções de ativação

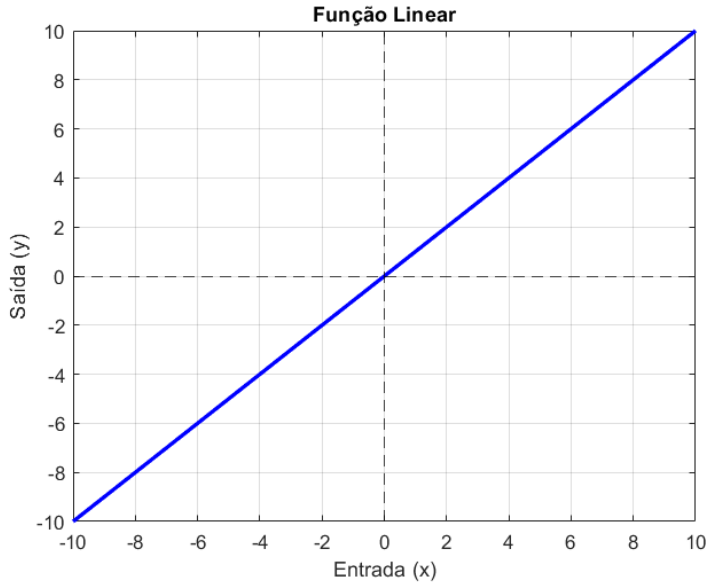
A função de ativação desempenha um papel determinante na arquitetura da rede, sendo a responsável por introduzir a não-linearidade ao modelo. Sem elas, as redes não seriam capazes de aprender padrões complexos ou resolver problemas que não sejam linearmente separáveis[18].

Existem diversas funções recomendadas pela literatura. Para problemas de regressão destaca-se a Função Linear. Ela é definida matematicamente pela Equação (2.3) uma relação de identidade, onde a saída é proporcional à entrada,

$$\varphi(v_k) = v_k \tag{2.3}$$

o comportamento gráfico desta função, que não impõe limites de saturação ao sinal, pode ser observado na Figura 4.

Figura 4: Função de ativação linear



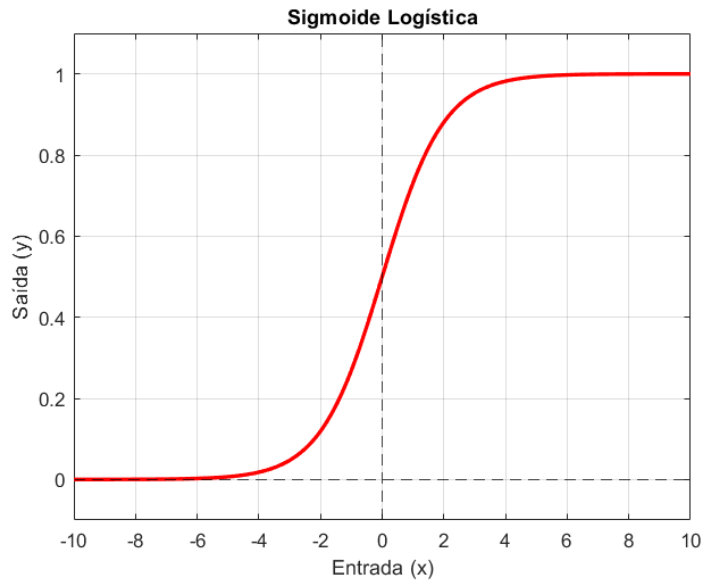
Fonte Autor

Historicamente, funções com formato sigmoidal (em "S") foram predominantes, especialmente para classificação. A mais clássica é a Sigmoide Logística, que mapeia qualquer valor de entrada para o intervalo entre 0 e 1. Sua expressão matemática é dada pela Equação (2.4):

$$\varphi(v_k) = \frac{1}{1 + e^{-v_k}} \tag{2.4}$$

A curva característica dessa função é apresentada na Figura 5.

Figura 5: Função de ativação sigmoide logística



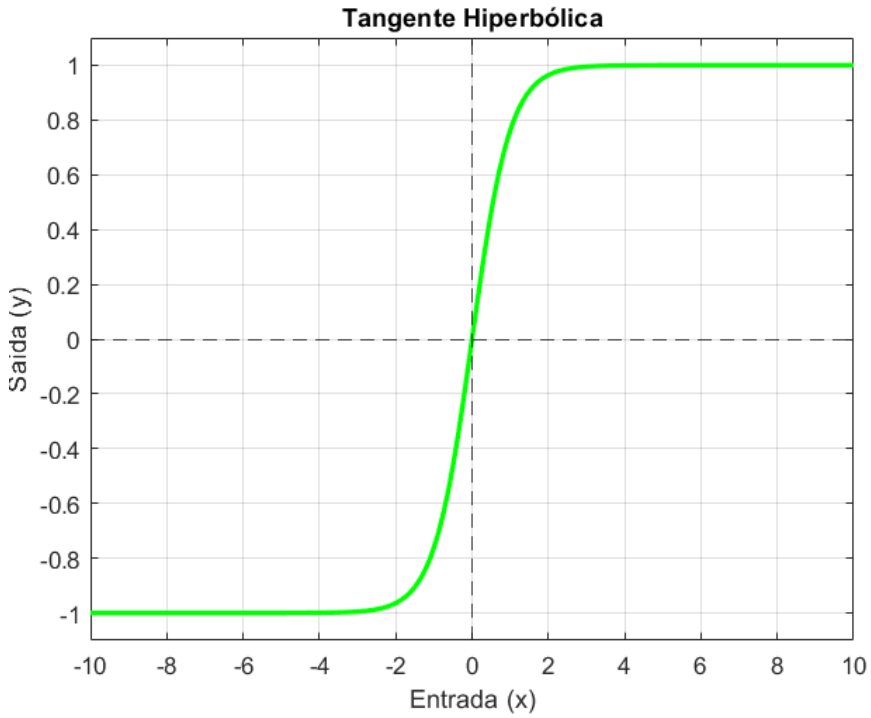
Fonte Autor

De forma análoga, a Tangente Hiperbólica também possui formato em "S", porém seus valores de saída variam entre -1 e 1, sendo centrada na origem. Matematicamente, ela é descrita pela equação (2.5):

$$\varphi(v_k) = \frac{e^{v_k} - e^{-v_k}}{e^{v_k} + e^{-v_k}} \tag{2.5}$$

Sua representação gráfica encontra-se ilustrada na Figura 6.

Figura 6: Função de ativação tangente hiperbólica



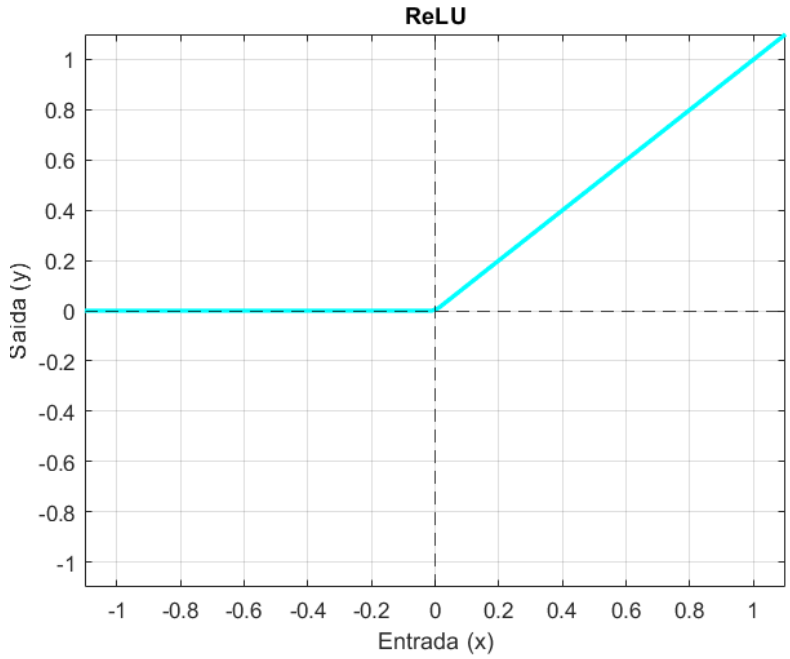
Fonte Autor

Embora as funções sigmoidais (Sigmoide e Tangente Hiperbólica) tenham sido o padrão histórico, elas apresentam desvantagens computacionais significativas quando aplicadas em redes com muitas camadas. Nesse contexto, a literatura moderna consolidou o uso da Unidade Linear Retificada (ReLU, *Rectified Linear Unit*). Sua principal vantagem reside na simplicidade e na eficiência computacional[20]. Diferentemente das curvas suaves anteriores, a ReLU atua como um filtro simples: ela zera qualquer valor negativo e mantém inalterados os valores positivos. Sua definição matemática é dada pela Equação (2.6) função máximo:

$$\varphi(v_k) = \max(0, v_k) \tag{2.6}$$

Ou seja, se a entrada for maior que zero, a saída é igual à entrada; caso contrário, a saída é nula. Essa característica permite que a rede processe informações de forma mais rápida, sendo a escolha padrão para as camadas ocultas em arquiteturas modernas, como as Redes Neurais Convolucionais[20]. O comportamento gráfico da ReLU é demonstrado na Figura 7.

Figura 7: Função de ativação ReLU



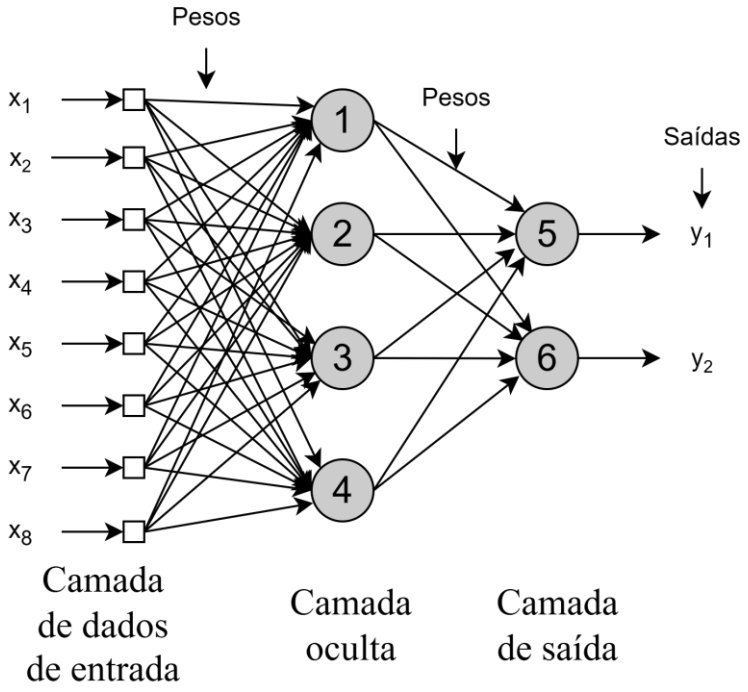
Fonte Autor

2.4 Redes Neurais

A interconexão desses neurônios dá origem às Redes Neurais Artificiais, organizadas em arquiteturas específicas. Nessas estruturas, os elementos são dispostos sequencialmente em camadas. Quando o fluxo de informação ocorre de maneira unidirecional, partindo da entrada em direção à saída e sem ciclos de retorno, a topologia é classificada como *feedforward* [18].

A primeira camada é denominada camada de entrada (*input layer*) e tem a função de receber os dados brutos. As camadas subsequentes, posicionadas entre a entrada e a camada de saída (*output layer*), são chamadas de camadas ocultas (*hidden layers*). É justamente a inserção de múltiplas camadas ocultas que define o conceito de Aprendizado Profundo, pois permite que o modelo extraia características cada vez mais abstratas dos dados. A Figura 8 exemplifica uma rede neural *feedforward* densamente conectada com uma camada oculta [18].

Figura 8: Rede neural *feedforward* densamente conectada com uma camada oculta



Fonte [18]

2.5 Algoritmos de Aprendizado

Uma vez definida a estrutura da rede, é necessário estabelecer o método de treinamento. O algoritmo fundamental é a Retropropagação do Erro (*Backpropagation*). Essencialmente, essa técnica utiliza exemplos supervisionados para ajustar iterativamente os pesos sinápticos, minimizando a diferença entre a resposta da rede e o valor desejado [19][18].

A execução desse algoritmo ocorre em duas etapas: a fase de propagação (*forward pass*) e a fase de retropropagação (*backward pass*). Na primeira etapa, o sinal de entrada percorre a rede camada por camada até gerar uma saída, sendo armazenados os níveis de ativação de cada neurônio. Ao final, calcula-se o erro global comparando a saída obtida com o rótulo esperado (*target*) [18][19].

Na segunda etapa, o erro é propagado no sentido inverso (da saída para a entrada). Utilizando a regra da cadeia para derivadas parciais, o algoritmo determina a responsabilidade de cada neurônio no erro total, grandeza denominada Gradiente Local (δ). Por fim, o ajuste dos pesos é calculado considerando a taxa de aprendizado (η), o gradiente local e o sinal de entrada do neurônio, conforme a Equação (2.7) que representa a formulação fundamental da atualização de pesos:

$$\begin{pmatrix} Correção \\ de\ peso \\ \Delta w_{ji}(n) \end{pmatrix} = \begin{pmatrix} Taxa\ de \\ aprendizado \\ \eta \end{pmatrix} \times \begin{pmatrix} Local \\ gradiente \\ \delta_j(n) \end{pmatrix} \times \begin{pmatrix} Entrada\ do \\ neuronio \\ y_i(n) \end{pmatrix} \quad (2.7)$$

Entretanto, em cenários de alta complexidade, a aplicação pura do gradiente descendente pode apresentar convergência lenta ou oscilações excessivas. Para mitigar essas limitações, a literatura avançou para algoritmos de otimização adaptativos [19].

Dentre as variações do Gradiente Descendente Estocástico (SGD, *Stochastic Gradient Descent*), este trabalho adotou o otimizador Adam *(Adaptive Moment Estimation*). O Adam combina as vantagens de duas outras extensões populares: o AdaGrad, que funciona bem com gradientes esparsos, e o RMSProp, que lida bem com objetivos não estacionários) [21].

Sua principal inovação consiste em calcular taxas de aprendizado adaptativas individuais para cada parâmetro, baseando-se em estimativas do primeiro momento (a média dos gradientes passados) e do segundo momento (a média dos gradientes quadrados não centralizada). Essa abordagem confere ao Adam robustez e eficiência computacional, justificando sua escolha para o treinamento da rede neural proposta [21].

2.6 Redes Neurais Convolucionais

Com o estabelecimento de algoritmos de otimização robustos como o Adam e funções de ativação eficientes, tornou-se viável o treinamento de arquiteturas profundas e especializadas para diferentes tipos de dados. Nesse contexto, para o processamento de sinais e imagens, a literatura consagrou o uso das Redes Neurais Convolucionais (CNNs, *Convolutional Neural Networks*) [16].

Essas redes diferenciam-se das arquiteturas clássicas por serem projetadas especificamente para processar dados com topologia em forma de grade, como séries temporais (1D), imagens (2D) e volumes de dados (3D). A operação fundamental dessa estrutura é a convolução matemática: em vez de conectar cada neurônio a todos da camada anterior, a rede utiliza um filtro (ou kernel) de dimensões reduzidas que desliza sobre a matriz de entrada [16].

Em cada passo desse deslizamento, o kernel realiza um produto escalar com a região local da entrada, gerando um mapa de ativação denominado mapa de características (*feature map*). Essa mecânica confere à CNN uma propriedade crucial conhecida como invariância à translação. Isso significa que a rede é capaz de identificar

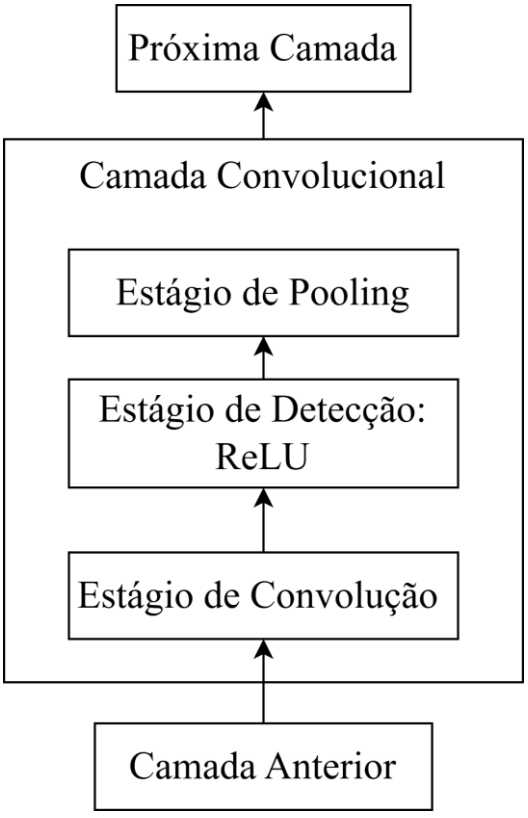
um padrão relevante independentemente de sua posição temporal ou espacial na entrada, garantindo robustez ao modelo [16].

Uma camada convolucional típica é estruturada em três estágios sequenciais. No primeiro, executa-se o processo de convolução, gerando um conjunto de ativações lineares. No segundo estágio, conhecido como estágio de detecção, cada ativação é submetida a uma função não linear, como a ReLU, anteriormente descrita.

Por fim, aplica-se uma função de agrupamento (*pooling function*). Essa função tem o objetivo de substituir a saída da rede em um determinado local por um resumo estatístico das saídas vizinhas. A técnica mais utilizada é o *Max Pooling*, que particiona a entrada em regiões retangulares e retorna apenas o valor máximo de cada uma. Essa operação reduz a dimensionalidade espacial dos dados e o custo computacional, além de conferir à rede robustez a pequenas translações na entrada [16][19].

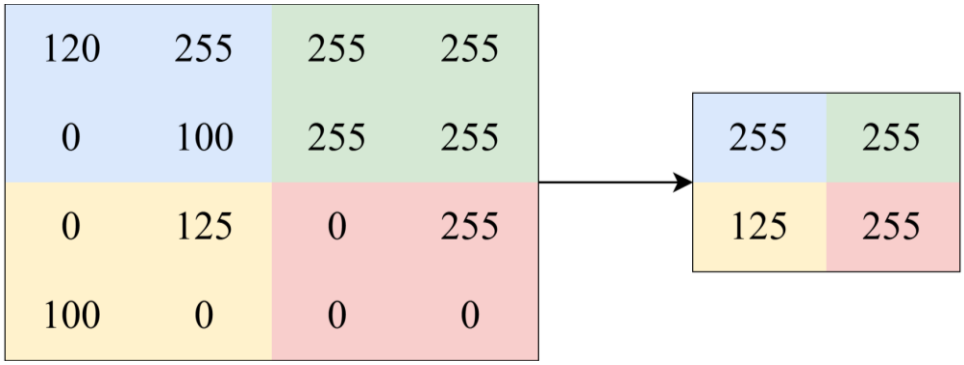
As Figuras 9 e 10 ilustram respectivamente o diagrama de blocos dessa camada e o funcionamento prático da operação de *Max Pooling*.

Figura 9: Diagrama de bloco de uma camada convolucional



Fonte [16]

Figura 10: Operação *max pooling*



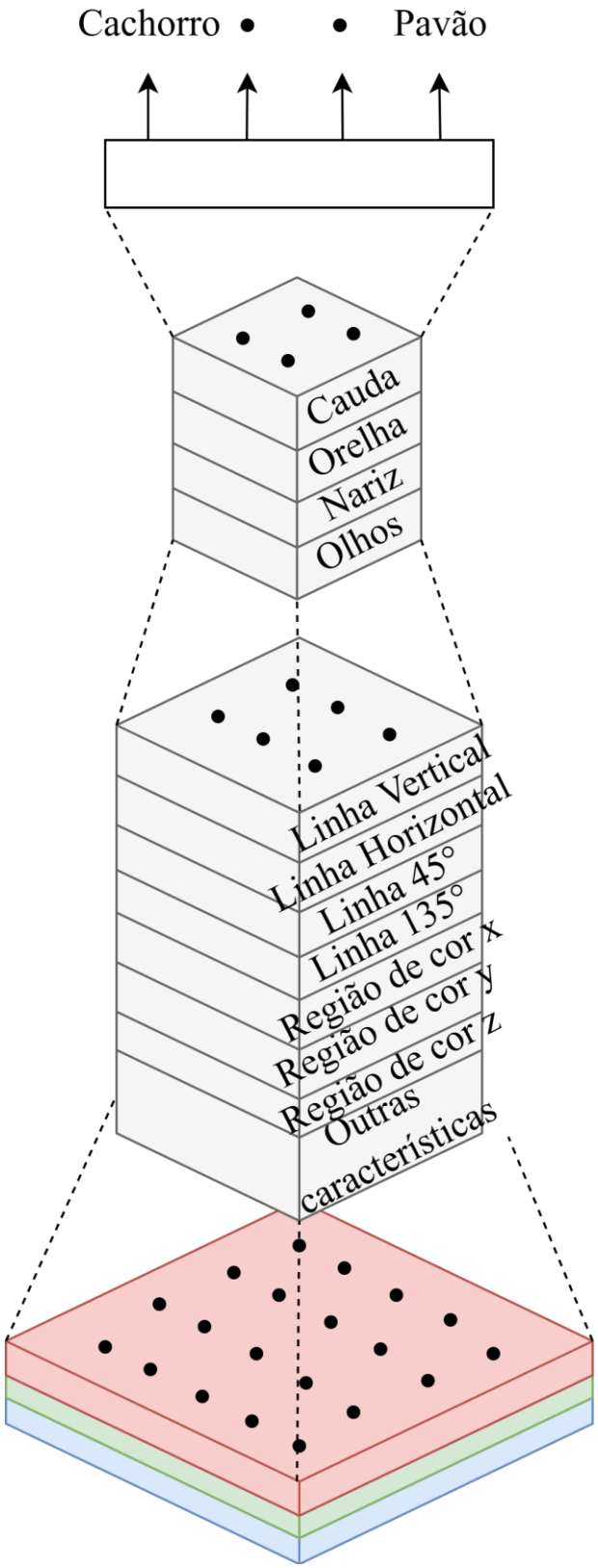
Fonte Autor

A Figura 10 ilustra o funcionamento prático da operação de *Max Pooling*. Neste exemplo, uma matriz de entrada de dimensões 4 x 4 é percorrida por um filtro (ou janela) de 2 x 2 com um passo (*stride*) de 2, configurando regiões retangulares não sobrepostas. O algoritmo atua extraindo o valor máximo de ativação de cada sub-região mapeada, resultando em uma matriz de saída reduzida de 2 x 2. Conforme exemplificado visualmente, os valores máximos extraídos dos quatro quadrantes da matriz original foram 255, 255, 125 e 255, respetivamente. Essa etapa de subamostragem reduz drasticamente a dimensionalidade dos dados e a quantidade de parâmetros da rede, além de criar uma invariância de posição espacial que ajuda a prevenir o sobreajuste (*overfitting*) do modelo.

Com os fundamentos das camadas convolucionais estabelecidos, é possível integrá-las às camadas densamente conectadas (*fully connected layers*). Uma arquitetura típica de DL opera em dois estágios: primeiramente, uma sequência de camadas convolucionais e de *pooling* atua como extratora de características, transformando os dados brutos em representações abstratas. Posteriormente, antes de conectar essa estrutura à camada final, realiza-se uma operação de vetorização ou achatamento (*flatten*). Essa etapa converte os mapas de características multidimensionais em um vetor unidimensional. Este vetor alimenta as camadas densas, que são responsáveis pela tarefa final de inferência.

A camada de saída é configurada conforme o objetivo do problema: para classificação, utilizam-se múltiplos neurônios (geralmente com ativação *Softmax*, onde cada um representa uma classe) e para regressão, utiliza-se tipicamente um único neurônio com ativação linear, capaz de estimar um valor contínuo. A Figura 11 ilustra a topologia completa de uma rede, evidenciando a transição entre a extração de características e a classificação.

Figura 11: Topologia da rede com camadas convolutivas e densamente conectada



Fonte [19]

Na Figura 11 o processamento inicia-se na base da rede, onde a imagem de entrada, com seus respetivos canais espaciais, é analisada pelas primeiras camadas

convolucionais. Neste estágio inicial, a rede atua sobre pequenos campos receptivos (*receptive fields*) para identificar características de baixo nível (*low-level features*), como linhas verticais, horizontais, ângulos e transições de cores.

À medida que os dados avançam para as camadas mais profundas, a rede combina esses padrões rudimentares para formular representações semânticas cada vez mais complexas, identificando partes específicas de objetos, como cauda, orelhas e olhos. Para que os neurônios nas camadas superiores consigam detectar características tão complexas, é necessário que os seus campos receptivos abranjam uma área maior da imagem original. Isso é alcançado por meio da redução sistemática da resolução espacial dos mapas de características (*feature maps*), frequentemente empregando operações de subamostragem (como o *Max Pooling*) ou aumentando o passo (*stride*) da convolução.

Na etapa final do processamento, após as sucessivas convoluções, a estrutura espacial tridimensional dos dados é achatada (*flattening*) num vetor unidimensional. Esse vetor alimenta as camadas totalmente conectadas (*fully connected layers*), responsáveis por combinar todas as características extraídas para a tomada de decisão. O modelo atribui pesos distintos a diferentes ativações. Por exemplo, uma forte ativação em neurônios que representam olhos e cauda terá um peso determinante para que a função de saída (como a *Softmax*) classifique a imagem com alta probabilidade para a classe Pavão, em detrimento da classe Cachorro.

2.7 Arquiteturas de Redes Neurais Convolucionais

Na literatura, existem diversos modelos de arquiteturas de Redes Neurais Convolucionais pré-treinadas, submetidas a um treinamento extensivo em grandes conjuntos de dados para a resolução de problemas de visão computacional. A análise comparativa e a seleção adequada desses modelos, muitas vezes utilizados como *backbones*, são fundamentais para o desempenho da aplicação final. Dentre as arquiteturas para classificação de imagem mais proeminentes, destacam-se: *ResNet, EfficientNet, DenseNet, Xception, Inception, MobileNet e ConvNeXt* [22][23].

A arquitetura *Inception* (especificamente a versão V3) explorou formas de escalar as redes neurais, aumentando sua largura e profundidade de maneira computacionalmente eficiente. Para isso, a arquitetura introduziu o conceito de Módulos *Inception*, que implementam múltiplas operações convolucionais com filtros de tamanhos variados (1x1,3x3) processados em paralelo dentro da mesma camada. Essa abordagem

permite que a rede capture padrões em diferentes escalas simultaneamente. Além disso, a InceptionV3 aprimorou esse conceito utilizando a fatoração de convoluções (decompondo filtros maiores em menores) e técnicas de regularização agressiva para evitar o sobreajuste (*overfitting*) [22], [23], [24].

Proposta como uma interpretação extrema dos princípios da *Inception*, surge a arquitetura *Xception* (*Extreme Inception*). Sua premissa fundamental é a de que o mapeamento de correlações entre canais (profundidade) e o mapeamento de correlações espaciais podem ser inteiramente desacoplados. Para implementar essa hipótese, a arquitetura substitui os módulos Inception originais por convoluções separáveis em profundidade (*depthwise separable convolutions*). Essa operação divide o processamento em duas etapas: uma convolução espacial (realizada independentemente para cada canal) seguida de uma convolução pontual (1 x 1) para combinar as saídas. Essa modificação estrutural permitiu que a *Xception* superasse o desempenho da InceptionV3, apresentando maior eficiência no uso de parâmetros [22], [23], [25]

Em uma direção complementar, buscando viabilizar redes com centenas de camadas, a *ResNet* (*Residual Network*) introduziu o conceito fundamental de blocos residuais. Nessa estrutura, utilizam-se conexões de atalho (*skip connections*) que pulam camadas intermediárias, somando a informação da entrada do bloco diretamente à sua saída. Esse método foi desenvolvido para solucionar o problema de degradação do gradiente, viabilizando o treinamento de redes significativamente mais profundas do que as possíveis até então. Dentre as variações mais comuns presentes na literatura, destacam-se a ResNet-50, ResNet-101 e ResNet-152 [22], [23], [26].

Como uma evolução alternativa às conexões residuais da *ResNet*, foi proposta a *DenseNet* (*Densely Connected Convolutional Network*). Enquanto a *ResNet* combina as características através da soma de valores (*element-wise*), a *DenseNet* propõe conectar cada camada a todas as camadas subsequentes através da concatenação de mapas de características. Essa arquitetura promove um intenso reaproveitamento de características (*feature reuse*), o que fortalece a propagação dessas informações ao longo da rede e reduz a quantidade total de parâmetros necessários, embora aumente o consumo de memória durante o treinamento devido à concatenação de tensores. Suas variações mais utilizadas são a *DenseNet-121*, *DenseNet-169* e *DenseNet-201* [22], [23], [27].

Compartilhando o princípio das convoluções separáveis em profundidade, a família MobileNet foi projetada explicitamente para operar em ambientes com restrição de recursos computacionais e requisitos de baixa latência, como dispositivos móveis. A versão mais recente, *MobileNetV3*, representa um avanço significativo ao empregar técnicas de Busca de Arquitetura Neural (NAS, *Neural Architecture Search*) combinadas com o algoritmo *NetAdapt*. Isso significa que a estrutura da rede foi otimizada automaticamente por algoritmos para encontrar o melhor equilíbrio entre precisão e latência. Além disso, a V3 incorpora módulos de atenção do tipo *Squeeze-and-Excitation* (SE) e novas funções de ativação (*h-swish*), disponibilizando-se nas variantes *Small* e *Large* para adequar-se a diferentes orçamentos computacionais [22], [23], [28].

Seguindo a vertente da eficiência, a família *EfficientNet* propôs uma mudança de paradigma no redimensionamento de modelos. Diferentemente das abordagens anteriores que aumentavam arbitrariamente apenas uma dimensão da rede (como profundidade ou largura), a *EfficientNet* introduziu o método de Escalonamento Composto (*Compound Scaling*). Esse método utiliza um coeficiente composto (∅) para ajustar, de maneira balanceada e uniforme, as três dimensões críticas da rede: profundidade, largura e resolução da imagem de entrada. A arquitetura base, denominada *EfficientNet-B0*, foi projetada utilizando *Neural Architecture Search* (NAS) para otimizar a relação entre precisão e custo computacional (FLOPs). As variantes subsequentes (B1 a B7) são obtidas escalonando essa base, resultando em modelos que atingem acurácia superior com significativamente menos parâmetros que as arquiteturas convencionais[22], [23], [29].

Recentemente, a hegemonia das redes convolucionais foi desafiada pela ascensão dos *Vision Transformers* (ViTs), que demonstraram desempenho superior em tarefas globais. Em resposta a esse cenário, surgiu a família *ConvNeXt*, proposta como uma modernização das CNNs puras para competir diretamente com os Transformers hierárquicos (como o *Swin Transformer*). A arquitetura *ConvNeXt* não introduz novos operadores matemáticos complexos, em vez disso, ela reestrutura uma *ResNet* padrão adotando decisões de design típicas de Transformers. Isso inclui o uso de kernels maiores (7×7) para aumentar o campo receptivo, a substituição de funções de ativação (ReLU por GELU), a redução de normas e o uso de camadas de convolução patchify. Com essas adaptações, a ConvNeXt alcançou precisão comparável ou superior ao estado da arte dos

Transformers, mantendo a simplicidade e a eficiência inerentes às convoluções[22], [23], [30].

Neste capítulo, foram apresentados os fundamentos teóricos que sustentam o Aprendizado de Máquina e o Aprendizado Profundo. O texto explorou desde a estrutura elementar do neurônio artificial, suas funções de ativação e o mecanismo de otimização por retropropagação (*backpropagation*), até o funcionamento detalhado das Redes Neurais Convolucionais (CNNs). Compreendeu-se como as sucessivas operações de convolução e *pooling* atuam em conjunto para realizar a extração hierárquica de características, permitindo que o modelo aprenda padrões visuais a partir de matrizes bidimensionais.

A consolidação desse arcabouço teórico é indispensável para o desenvolvimento das etapas subsequentes deste trabalho. É a base matemática e arquitetural detalhada neste capítulo que fundamenta e justifica a proposta de tratar os diagramas de constelação DP m-PSK e DP m-QAM como imagens, transformando a rede neural em um instrumento robusto para a estimação da OSNR.

Capítulo 3

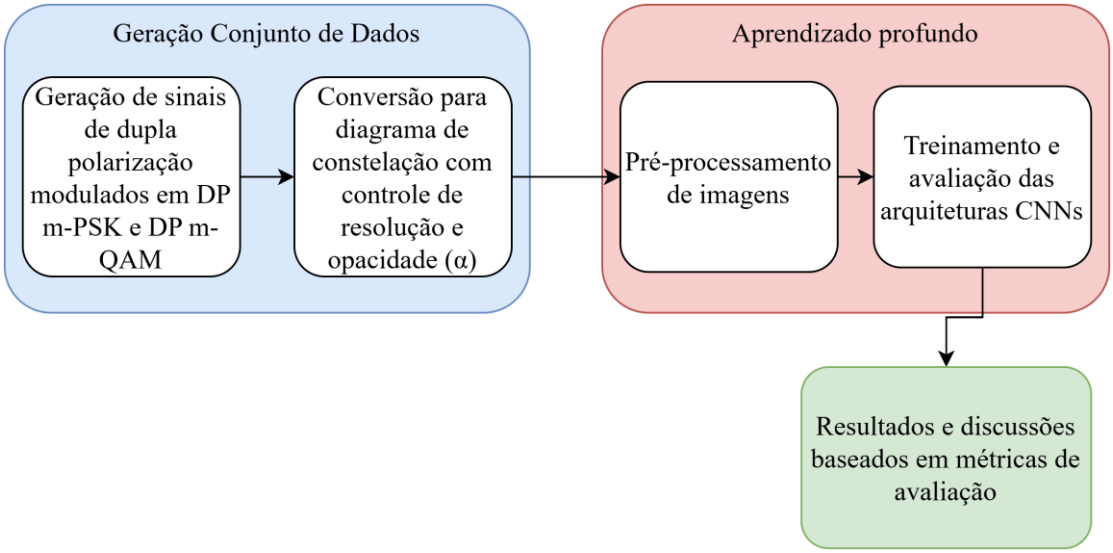
Metodologia

Este capítulo tem como objetivo principal descrever a metodologia utilizada para o processo de estimação de OSNR, proposto neste trabalho, em redes ópticas, por meio da utilização do aprendizado profundo. Tais processos serão descritos em detalhes nas seções subsequentes do trabalho.

3.1 Modelo do sistema

O modelo de sistema proposto segue um fluxo de trabalho estruturado em três etapas sequenciais. A primeira, referente à geração dos dados, subdivide-se na simulação de sinais de dupla polarização modulados em DP m-PSK e DP m-QAM, e na sua conversão para diagramas de constelação em formato de imagem RGB, com controle de resolução e opacidade de símbolo (α). Na segunda etapa, denominada aprendizado profundo, as imagens são submetidas a técnicas de pré-processamento (redimensionamento, normalização e aumento de dados) para, posteriormente, treinar e avaliar um conjunto diversificado de arquiteturas de CNNs. Por fim, a terceira etapa consiste na análise dos resultados e discussão baseada em métricas de avaliação. A Figura 12 ilustra visualmente o diagrama de blocos desse fluxo.

Figura 12: Fluxo do modelo do sistema

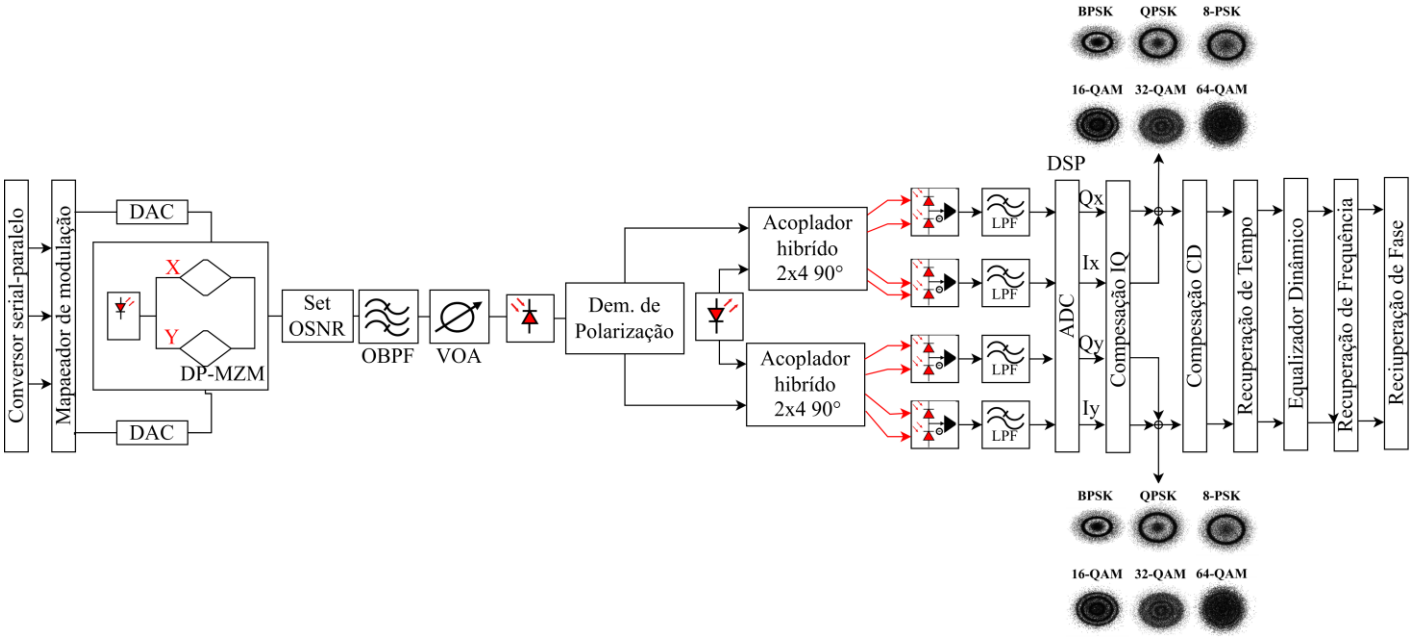


Fonte Autor

3.2 Geração do conjunto de dados

A geração do conjunto de dados foi realizada através de simulações numéricas utilizando o software *VPIphotonics* [31]. A escolha desta ferramenta justifica-se por sua ampla consolidação e validação na literatura científica especializada, sendo considerada um padrão da indústria para a modelagem de sistemas de comunicação óptica devido à sua alta fidelidade na emulação de componentes físicos e efeitos de canal. O cenário experimental configurado segue uma topologia *back-to-back* (B2B), onde o transmissor é conectado diretamente ao receptor sem a interposição de enlace de fibra. A Figura 13 ilustra o diagrama esquemático da configuração simulada.

Figura 13: Configuração da simulação *back-to-back* no software VPIphotonics.



Fonte Autor[32][33]

Na primeira parte do transmissor o fluxo de bits é gerado e processado por um mapeador de modulação. Os sinais mapeados são convertidos de digital para analógico e usados para modular uma portadora óptica em $f_0 = 192{,}1\ THz$ utilizando um Modulador Mach-Zehnder de Dupla Polarização (DP-MZM, *Dual-Polarization Mach-Zehnder Modulator*). Isso resulta em sinais ópticos multiplexados nos estados de polarização ortogonais $\overrightarrow{E_x}$ e $\overrightarrow{E_y}$. Após isso, o controle do nível de OSNR é realizado através do bloco (*Set OSNR*) que injeta ruído Emissão Espontânea Amplificada (ASE, *Amplified Spontaneous Emission*), permitindo a varredura de diferentes níveis de relação sinal-ruído para a composição do conjunto de dados (*dataset*) [32].

Na parte do receptor, os sinais modulados passam por um filtro passa-banda óptico (OBPF, *Optical Band-pass Filter*) centrado em $f_c = 192{,}1\,THz$, seguido por atenuador óptico variável (VOA, *Variable Optical Attenuator*) para manter uma potência constante de -2 dBm no fotodetector. Após a atenuação, os estados de polarização são desmultiplexados e alimentam os acopladores híbridos 2x4 de 90° juntamente com o sinal do oscilador local centrado em $f_{LO} = 192{,}1\,THz$. As componentes em fase e quadratura resultantes são detectadas por fotodetectores, amplificados por amplificadores de Transimpedância (TIAs, *Transimpedance Amplifiers*) e filtrados por filtros passa-baixa (LPFs), antes de serem processados pelo processamento digital de sinais (DSP, *Digital Signal Processing*) [32].

No interior do DSP, o sinal é convertido de analógico para digital, e os atrasos de fase e quadratura são compensados no bloco de compensação IQ. A dispersão cromática e os desvios de temporização são corrigidos pelos blocos de compensação de dispersão e recuperação de tempo. A dispersão do modo de polarização, o desvio de frequência e o ruído de fase são subsequentemente tradados pelos blocos de equalizador dinâmico, recuperação de frequência e recuperação de fase, respectivamente [32].

Nesse estudo, não é necessário o processamento completo do sinal digital para obter os diagramas de constelação. Após compensar os atrasos de fase e quadratura, os sinais BPSK, QPSK, 8PSK, 16QAM, 32QAM e 64QAM de polarização dupla são extraídos e usados para gerar amostras de treino e teste para os modelos de rede neural propostos. Eles são capturados neste ponto para preservar as características de ruído necessárias para a estimação de OSNR. Para todas as simulações, fixou-se a taxa de símbolos em 28 Gbaud, de modo que a taxa de bits variou naturalmente de acordo com a ordem de cada formato de modulação.

Os diagramas de constelação resultantes são salvos como imagens, cuja clareza visual dos conjuntos de símbolos é influenciada por dois parâmetros chave: pontos por polegada (DPI*, Dots Per Inch*), que define a resolução da imagem, e o α, que determina a opacidade e a visualização da sobreposição dos símbolos. Ao variar esses parâmetros, é possível simular diferentes níveis de ruído visual e granularidade de representação, que afetam diretamente o comportamento de aprendizagem dos modelos durante o treinamento e a avaliação.

3.3 Pré-processamento

O pré-processamento das imagens consistiu em etapas sistemáticas destinadas a padronizar e enriquecer o conjunto de dados. Incialmente, as imagens foram carregadas a partir de caminhos especificados em um arquivo .csv e redimensionadas para uma resolução fixa de 224x224 pixels, garantindo dimensões de entrada uniformes para a rede neural. Após o redimensionamento, as imagens foram convertidas em matrizes numéricas de 8 bits (uint8).

Para expandir a variabilidade do conjunto de treino e aprimorar a capacidade de generalização dos modelos, empregou-se a estratégia de aumento de dados (*data augmentation*). Esta técnica enriquece a base de dados e previne o sobreajuste ao multiplicar as amostras de aprendizagem. Este processo é realizado através da aplicação sistemática de transformações geométricas e manipulações de intensidade nas imagens originais, tais como redimensionamento, translações, rotações, além de ajustes de brilho e contraste. A Figura 14 ilustra um exemplo prático desta abordagem, na qual uma única imagem original deu origem a oito novas amostras sintéticas a partir das técnicas computacionais referidas.

Figura 14: Exemplo de aumento de dados (*data augmentation*)

Fonte Autor

As transformações aplicadas englobaram a inversão horizontal e vertical, rotações aleatórias, ajustes estocásticos de brilho e contraste, além de combinações de pequenas translações e mudanças de escala. Na sequência, procedeu-se à normalização dos dados. Este passo consistiu na divisão das matrizes representativas das imagens por

255,0, convertendo os valores de intensidade do intervalo numérico inteiro [0, 255] para a escala de ponto flutuante [0, 1]. Esta operação matemática finaliza a etapa de preparação dos dados, adequando as entradas para o processamento pelas camadas da rede neural e facilitando o processo de convergência [34].

É importante ressaltar que essas operações de aumento dados foram aplicadas exclusivamente ao conjunto de treinamento. As imagens destinadas ao conjunto de teste foram submetidas apenas ao redimensionamento e à normalização. Essa separação rigorosa garante que as características originais dos dados de teste sejam preservadas, permitindo uma avaliação imparcial do desempenho do modelo em cenários reais.

3.4 Modelos propostos de aprendizado profundo

Para a construção das redes, avaliou-se um conjunto diversificado de arquiteturas CNNs, conforme apresentado na Tabela 1. Eles vão desde modelos leves que são otimizados para ambientes com recursos limitados até arquiteturas mais profundas com capacidade representacional aprimorada. Os modelos selecionados, *MobileNetV3*, *ConvNeXt*, *DenseNet*, *ResNet*, *EfficientNet*, *Xception*, e *InceptionV3*, representam um amplo espectro de estratégias de design [22], [23]. Essa diversidade permitiu uma comparação completa de desempenho na estimação de OSNR a partir de diagramas de constelação, considerando tanto a precisão quanto a eficiência computacional.

Tabela 1: Resumo das arquiteturas CNNs e suas variações

Modelos	Variações
MobileNetV3 [28]	*Small, Large*
ConvNeXt [30]	*Tiny, Small, Base*
DenseNet [27]	121, 169, 201
ResNet [26]	50, 101, 152
EfficientNet [29]	B0, B3
Xception [25]	-
InceptionV3 [24]	-

3.5 Métricas de avaliação

Para quantificar a eficácia dos modelos de regressão na estimação da OSNR, adotaram-se quatro métricas estatísticas padrão.

A primeira métrica é o Erro Absoluto Médio (MAE, *Mean Absolute Error*), apresentado na Equação (3.1). Ele calcula a média aritmética das diferenças absolutas entre os valores preditos ($\widehat{y_i}$) e os reais (y_i), fornecendo uma medida direta da magnitude média do erro, sendo menos sensível a outliers (valores discrepantes),

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.1}$$

no qual n representa o número de amostras.

Em complemento, utilizou-se o Erro Quadrático Médio (MSE, *Mean Squared Error*), conforme a Equação (3.2),

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.2}$$

e a Raiz do Erro Quadrático Médio (RMSE, *Root Mean Squared Error*), definida na Equação (3.3).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.3}$$

Ao elevar as diferenças ao quadrado, o MSE e o RMSE penalizam erros maiores de forma mais severa que o MAE, sendo cruciais para identificar se o modelo está cometendo grandes falhas pontuais. Além disso, o RMSE apresenta a vantagem de manter a mesma unidade da variável de saída (dB), facilitando a interpretação.

Por fim, avaliou-se o Coeficiente de Determinação (R^2), descrito na Equação (3.4),

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3.4}$$

onde $\bar{y}$ é a média dos valores reais.

Diferentemente das anteriores que medem o erro, o R^2 (que varia de 0 a 1) indica a proporção da variância na variável dependente que é previsível a partir das variáveis independentes, servindo como um indicador global da qualidade do ajuste do modelo aos dados de teste.

Neste capítulo, detalhou-se o percurso metodológico adotado para viabilizar a estimação da OSNR utilizando técnicas de visão computacional. Inicialmente, demonstrou-se a geração de uma base de dados robusta por meio de simulações no

software VPI. Após isso, estruturou-se o rigoroso pipeline de pré-processamento das imagens, englobando o ajuste de resolução (DPI), o controle de opacidade (α) e o uso de aumento de dados (*Data Augmentation*), culminando na definição das arquiteturas de redes neurais convolucionais (CNNs) e das métricas estatísticas utilizadas para a avaliação.

A consolidação detalhada dessa metodologia é a garantia de que o modelo computacional foi treinado e testado sobre um ambiente que emula com alta fidelidade as condições operacionais de um receptor óptico coerente real.

Capítulo 4

Experimentos e resultados

Este capítulo apresenta a configuração experimental, a configuração de treino, as características do conjunto de dados e os resultados quantitativos obtidos a partir da avaliação de múltiplas arquiteturas CNNs na tarefa de estimação OSNR utilizando imagens de diagramas de constelação.

4.1 Parâmetros

Os hiperparâmetros e as configurações de treinamento, detalhados na Tabela 2, foram estabelecidos em consonância com as melhores práticas documentadas na literatura especializada para tarefas de Transferência de Aprendizado (*Transfer Learning*) [15][16].

Tabela 2: Configuração e parâmetros de treino

Parâmetros	Valor
Tamanho imagem de entrada	224 x 224 pixels
Tamanho do Lote (*Batch Size*)	32
Número de Épocas (*Epoch*)	100
Divisão de Validação	15% (dos dados de treinamento)
Divisão de Treino / Teste	80% treino / 20% teste
Embaralhamento (*Shuffle*)	Verdadeiro
Estado Aleatório (Semente)	Fixo (para reprodutibilidade)
Taxa de Aprendizado Inicial	1×10^{-4}
Taxa de Aprendizado Ajuste-Fino	1×10^{-5}
Otimizador (*Optimizer*)	Adam
Função Perda	MSE
Métrica de Avaliação	MAE
Pesos Pré-treinados	*ImageNet*
Modelo de Base Treinável	Sim (Ajuste fino ativado)
Camada *Pooling* Global	*GlobalAveragePooling2D*
Ativação Final	Linear
Dimensão de saída	1 (problema de regressão)

Para a implementação das arquiteturas, adotou-se a estratégia de Transferência de Aprendizado (*Transfer Learning*), inicializando todos os modelos com pesos pré-treinados na base de dados *ImageNet*. Como o objetivo deste trabalho é a estimação de um valor contínuo (OSNR) e não a classificação categórica, a camada superior original das redes foi descartada. No seu lugar, adicionou-se uma camada *Global Average Pooling* seguida por uma camada densa de saída com um único neurônio e função de ativação

linear. O processo de otimização foi conduzido pelo algoritmo Adam, configurado com uma taxa de aprendizado inicial de 1×10^{-4}. Para o ajuste fino (*fine-tuning*), onde os pesos da base convolucional tornam-se treináveis, essa taxa foi reduzida para 1×10^{-5}, permitindo um ajuste mais preciso dos parâmetros. O treinamento estendeu-se por 100 épocas (*epochs*) com um tamanho de lote (*batch size*) de 32 imagens.

Em relação aos dados, o conjunto total foi particionado mantendo 80% das amostras para o treinamento e 20% reservados exclusivamente para o teste final. Durante a fase de treino, 15% dos dados foram separados para validação interna, auxiliando no monitoramento da convergência. Para garantir a reprodutibilidade dos resultados apresentados, fixou-se uma semente aleatória (*random seed*) antes do embaralhamento e distribuição dos dados. A função de perda utilizada para guiar o aprendizado foi o MSE, enquanto o MAE serviu como a métrica principal para a avaliação de desempenho.

4.2 Conjunto de dados

O conjunto de dados gerado para o treinamento e avaliação das arquiteturas totaliza 19.050 imagens de diagramas de constelação. A distribuição das amostras foi realizada de acordo com o formato de modulação, cobrindo diferentes faixas operacionais de OSNR. Para garantir uma amostragem densa e representativa da evolução do ruído, os valores de OSNR foram incrementados com um passo fixo de 0,5 dB. A Tabela 3 detalha a quantidade de imagens geradas e a respectiva faixa de OSNR avaliada para cada formato de modulação.

Tabela 3: Resumo Estatístico de cada conjunto de dados

Modulação	Quantidade de imagem	mín OSNR (dB)	max OSNR (dB)
64QAM	4452	22	33
16QAM	3799	17	26
QPSK	3400	12,0	20,0
BPSK	2999	8,0	15,0
32 QAM	2600	15,0	21,0
8PSK	1800	17,0	21,0

Com o intuito de analisar o impacto das características visuais das imagens no desempenho das CNNs, os níveis de resolução, definidos em pontos por polegada (DPI), e o fator de opacidade dos símbolos α, foram sistematicamente variados durante a geração dos dados. É importante ressaltar que essas variações não foram agrupadas em um único conjunto de treinamento global. Em vez disso, criaram-se conjuntos de dados distintos e independentes. Essa abordagem permitiu treinar e avaliar os modelos separadamente para

cada cenário, identificando qual configuração visual proporciona a melhor extração de características para a estimação da OSNR. As cinco combinações paramétricas adotadas para compor esses subconjuntos estão descritas na Tabela 4.

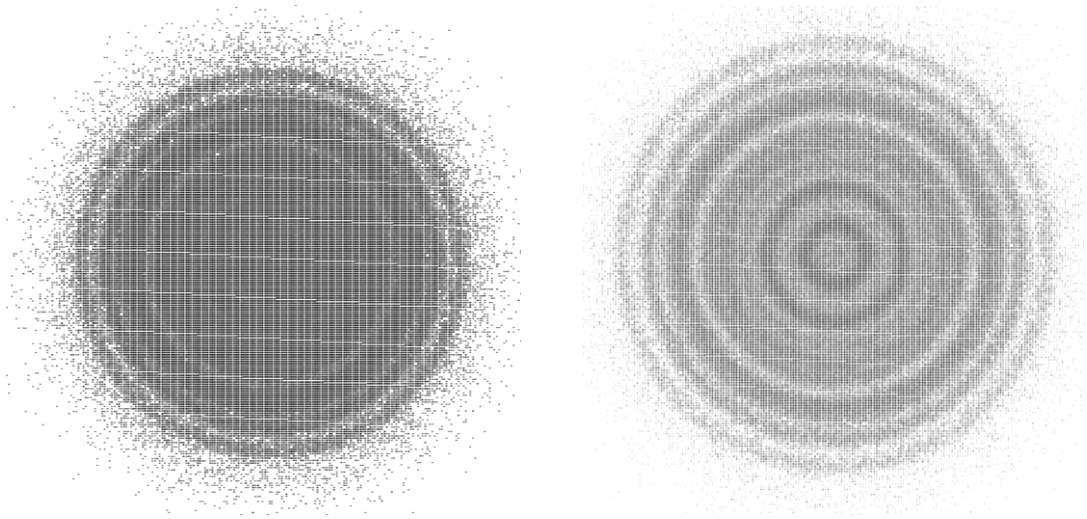
Tabela 4: Conjuntos de dados gerados

DPI	Opacidade (α)
100	1
150	0,2
150	0,4
150	0,6
250	0,2

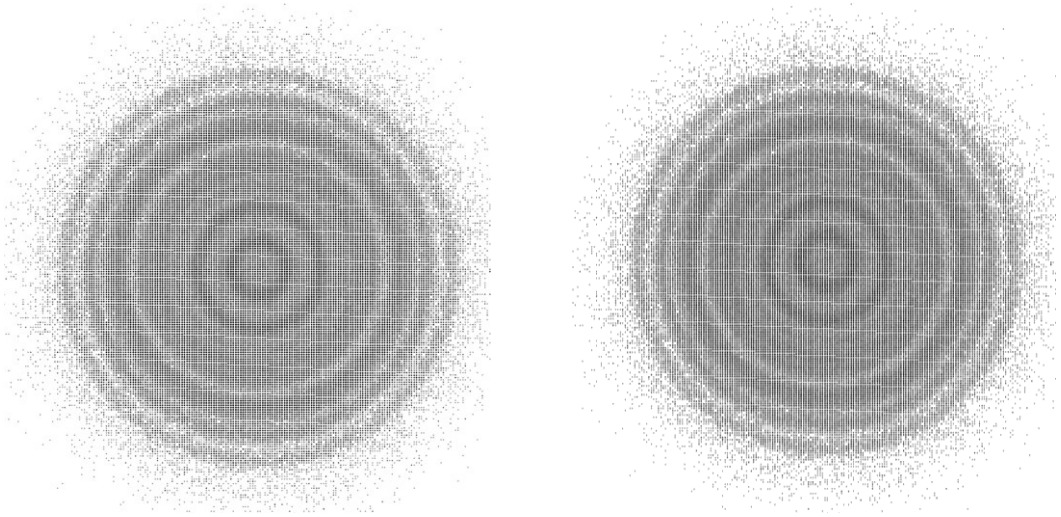
Para exemplificar o impacto visual dessas configurações, a Figura 15 ilustra os diferentes arranjos apresentados na Tabela 4 aplicados a um sinal 64-QAM com OSNR fixo em 33 dB. É possível observar como a variação simultânea da resolução e da transparência altera a densidade percebida e a dispersão das amostras na imagem, simulando diferentes condições de aquisição e granularidade visual.

Figura 15:(a) 100 DPI e α=1 (b) 150 DPI e α=0,2 (c) 150 DPI e α=0,4 (d) 150 DPI e α=0,6 (e) 250 DPI e α=0,2

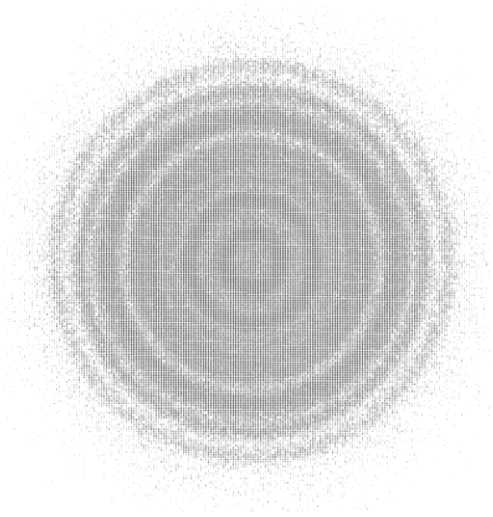
(a) (b)



(c) (d)



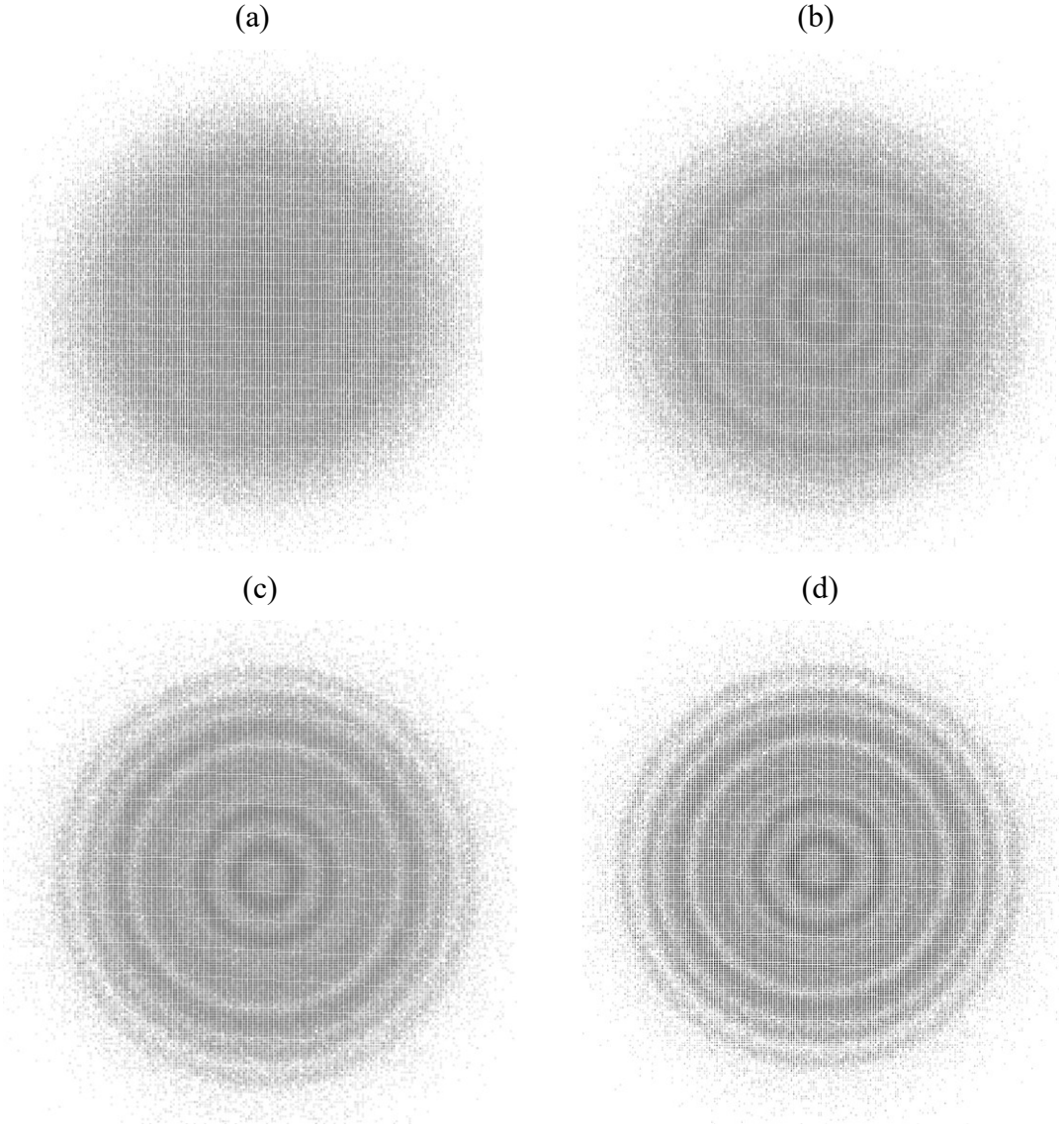
(d)



Fonte Autor

A Figura 16 apresenta os diagramas de constelação de um sinal 64 QAM sob quatro condições distintas de OSNR: 22 dB, 26 dB, 30 dB e 33 dB. Esses diagramas fornecem uma representação visual de como o ruído afeta a qualidade do sinal em sistemas de comunicação óptica. Em um cenário de menor OSNR, ilustrado na Figura 16 (a), os pontos da constelação exibem uma dispersão severa e sobreposição significativa entre as nuvens de símbolos. Esse espalhamento indica uma alta probabilidade de erros na região de decisão devido à interferência do ruído. Conforme o nível de OSNR aumenta, os símbolos tornam-se progressivamente mais agrupados e definidos em suas coordenadas ideais, refletindo um sinal mais limpo e com maior confiabilidade de detecção, conforme evidenciado nas Figuras 16(b), (c) e (d).

Figura 16:(a) Constelação 64 QAM a 22 dB OSNR (b) Constelação 64 QAM a 26 dB OSNR (c) Constelação 64 QAM a 30 dB OSNR (d) Constelação 64 QAM a 33 dB OSNR



Fonte Autor

Além disso, com o objetivo de melhorar a capacidade de generalização e reduzir o risco de sobreajuste (*overfitting*), foi aplicada a técnica de aumento de dados (*data augmentation*) utilizando a biblioteca em Python *Albumentations*. As seguintes transformações foram feitas:

- Inversão horizontal (*horizontal flip*) com probabilidade de 50%;
- Inversão vertical (*vertical flip*) com probabilidade de 50%;
- Rotação aleatória dentro de um intervalo de ±30 graus, aplicada com 50% de probabilidade;
- Ajuste aleatório de brilho e contraste com 50% de probabilidade;

- Deslocamento aleatório de até 5% da imagem com 50% de probabilidade;
- Escala aleatória de até 10% da imagem com 50% de probabilidade.

Esta estratégia aumenta a variabilidade do conjunto de treino e simula diferentes condições do mundo real, tornando assim o modelo mais robusto e generalista.

4.3 Estimação de OSNR

Os modelos foram avaliados em quatro métricas de regressão: RMSE, MAE, MSE e R^2. Inicialmente, para estabelecer um cenário de referência, as arquiteturas CNNs foram treinadas e avaliadas no conjunto de dados gerado com resolução de 100 DPI e opacidade total ($\alpha = 1$). A Tabela 5 detalha os resultados desta etapa inicial.

Tabela 5: Resultados para cada modelo utilizando as métricas RMSE, MAE, MSE e R^2 (DPI = 100 e $\alpha = 1$)

Modelo	RMSE (dB)	MAE (dB)	MSE (dB)	R^2
ConvNeXtBase	1,0073	0,7260	1,0146	0,9698
ConvNeXtSmall	1,4429	1,0824	2,0821	0,9392
ConvNeXtTiny	1,3413	1,0056	1,7991	0,9475
DenseNet121	1,1965	0,8514	1,4317	0,9573
DenseNet169	1,3416	0,9723	1,7998	0,9464
DenseNet201	1,1847	0,8448	1,4036	0,9582
EfficientNetB0	1,9216	1,3805	3,6924	0,8924
EfficientNetB3	1,6092	1,2015	2,5894	0,9245
InceptionV3	1,8921	1,3888	3,5799	0,8957
MobileNetV3Large	1,8833	1,3889	3,5467	0,8465
MobileNetV3Small	2,6867	1,9759	7,2183	0,7893
ResNet101	1,4209	1,0141	2,0189	0,9412
ResNet152	1,2849	0,9035	1,6510	0,9519
ResNet50	1,5255	1,0931	2,3270	0,9307
Xception	2,1604	1,5914	4,6673	0,8640

A partir desses primeiros resultados, observou-se que as arquiteturas com maior profundidade e capacidade representacional, principalmente as famílias *ConvNeXt* e *DenseNet*, obtiveram desempenho superior. O modelo *ConvNeXtBase* destacou-se como a melhor arquitetura neste cenário, atingindo um MAE de 0,7260 dB.

Com o objetivo de investigar o impacto da granularidade visual e otimizar o desempenho das redes, conduziu-se uma bateria de treinamentos incrementando a resolução para 150 DPI e variando sistematicamente o fator de opacidade ($\alpha = 0,6$, $\alpha = 0,4$ e $\alpha = 0,2$). Os resultados comparativos para esses três subconjuntos são apresentados nas Tabelas 6, 7 e 8, respectivamente.

Tabela 6: Resultados para cada modelo utilizando as métricas RMSE, MAE, MSE e R^2 (DPI = 150 e $\alpha = 0,6$)

Modelo	RMSE (dB)	MAE (dB)	MSE (dB)	R^2
ConvNeXtBase	0,7999	0,5797	0,6398	0,9816
ConvNeXtSmall	1,1714	0,8387	1,3721	0,9605
ConvNeXtTiny	1,1881	0,8537	1,4117	0,9593
DenseNet121	1,3525	0,9853	1,8292	0,9473
DenseNet169	1,2207	0,8879	1,4900	0,9571
DenseNet201	1,2377	0,9005	1,5318	0,9559
EfficientNetB0	4,3904	3,3170	19,2755	0,4506
EfficientNetB3	2,1643	1,6206	4,6844	0,8665
InceptionV3	1,6879	1,2607	2,8491	0,9188
MobileNetV3Large	1,9420	1,4105	3,7714	0,8913
MobileNetV3Small	3,4718	2,6986	12,0537	0,6527
ResNet101	1,3860	1,0042	1,9211	0,9446
ResNet152	1,1908	0,8823	1,4180	0,9596
ResNet50	1,4706	1,0914	2,1626	0,9377
Xception	2,1008	1,5424	4,4134	0,8742

Tabela 7: Resultados para cada modelo utilizando as métricas RMSE, MAE, MSE e R^2 (DPI = 150 e $\alpha = 0,4$)

Modelo	RMSE (dB)	MAE (dB)	MSE (dB)	R^2
ConvNeXtBase	0,7793	0,5569	0,6073	0,9822
ConvNeXtSmall	1,0697	0,7570	1,1443	0,9664
ConvNeXtTiny	1,1707	0,8612	1,3704	0,9598
DenseNet121	1,2387	0,9123	1,5345	0,9550
DenseNet169	1,1483	0,8512	1,3187	0,9613
DenseNet201	1,1504	0,8455	1,3233	0,9612
EfficientNetB0	1,5615	1,1443	2,4383	0,9284
EfficientNetB3	1,2647	0,9192	1,5996	0,9531
InceptionV3	1,4491	1,0715	2,0999	0,9414
MobileNetV3Large	1,7965	1,3423	3,2274	0,9053
MobileNetV3Small	4,1553	3,2559	17,2668	0,4932
ResNet101	1,2259	0,8886	1,5027	0,9559
ResNet152	0,9799	0,7094	0,9602	0,9718
ResNet50	1,2952	0,9586	1,6774	0,9508
Xception	1,8806	1,3905	3,5368	0,8962

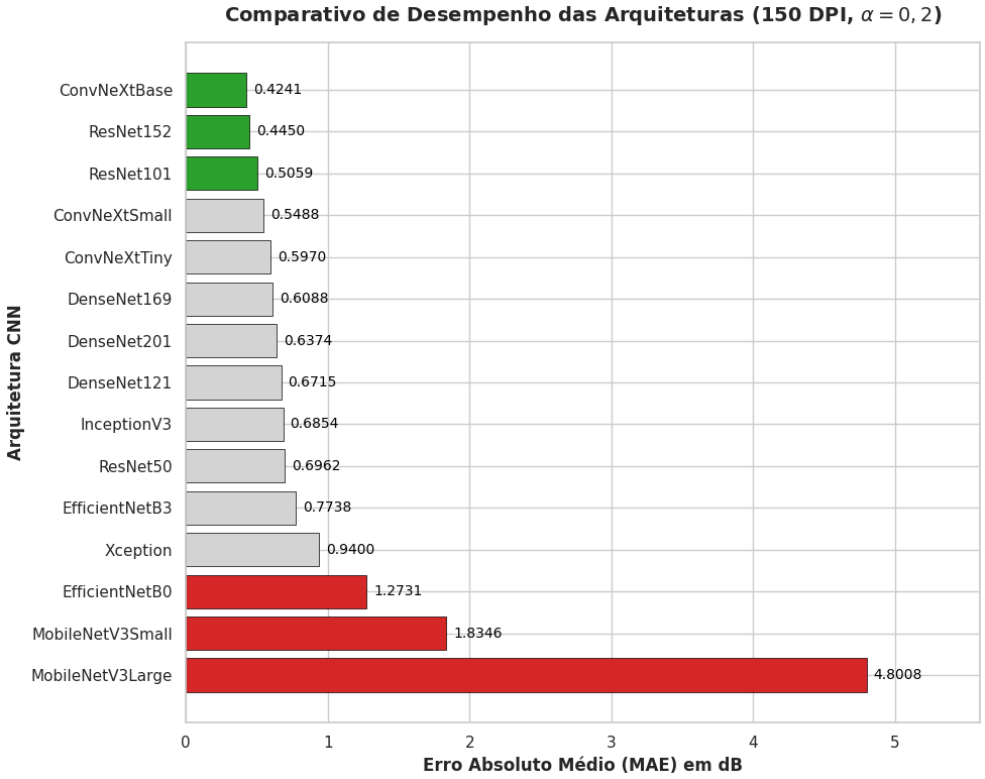
Tabela 8: Resultados para cada modelo utilizando as métricas RMSE, MAE, MSE e R^2 (DPI = 150 e $\alpha = 0,2$)

Modelo	RMSE (dB)	MAE (dB)	MSE (dB)	R^2
ConvNeXtBase	0,6007	0,4241	0,3608	0,9900
ConvNeXtSmall	0,7869	0,5488	0,6192	0,9828
ConvNeXtTiny	0,8381	0,5970	0,7024	0,9805
DenseNet121	0,9412	0,6715	0,8859	0,9755
DenseNet169	0,8425	0,6088	0,7099	0,9803

DenseNet201	0,8925	0,6374	0,7966	0,9779
EfficientNetB0	1,5831	1,2731	2,5062	0,9295
EfficientNetB3	1,0524	0,7738	1,1077	0,9689
InceptionV3	0,9616	0,6854	0,9246	0,9740
MobileNetV3Large	6,1760	4,8008	38,1512	-0,0549
MobileNetV3Small	2,4648	1,8346	6,0754	0,8292
ResNet101	0,7314	0,5059	0,5349	0,9850
ResNet152	0,6487	0,4450	0,4208	0,9882
ResNet50	0,9637	0,6962	0,9287	0,9743
Xception	1,3035	0,9400	1,6991	0,9522

A análise comparativa evidencia uma clara tendência de melhora no desempenho preditivo com a utilização da resolução de 150 DPI associada à redução da opacidade. Valores menores de α permitiram que as redes distinguissem com maior clareza a densidade de pontos nas regiões de sobreposição dos símbolos, sendo o subconjunto com α = 0,2 o que proporcionou a melhor extração de características. Neste cenário ótimo, o modelo *ConvNeXtBase* atingiu o menor MAE de todo o estudo, com 0,4241 dB, e R^2 de 0,9900. Para melhor visualização do desempenho comparativo das arquiteturas no cenário ótimo de 150 DPI e α = 0,2, a Figura 17 apresenta o ranqueamento dos modelos em função do MAE. Observa-se claramente a superioridade das arquiteturas *ConvNeXt* e *ResNet*, que concentram os menores índices de erro.

Figura 17: Comparativo de desempenho das arquiteturas para o subconjunto com 150 de DPI e α = 0,2



Fonte Autor

Diante da premissa de que o aumento da resolução poderia continuar beneficiando o aprendizado, realizou-se um teste final elevando a resolução para 250 DPI, mantendo a opacidade ideal de α 0,2. Os resultados são ilustrados na Tabela 9.

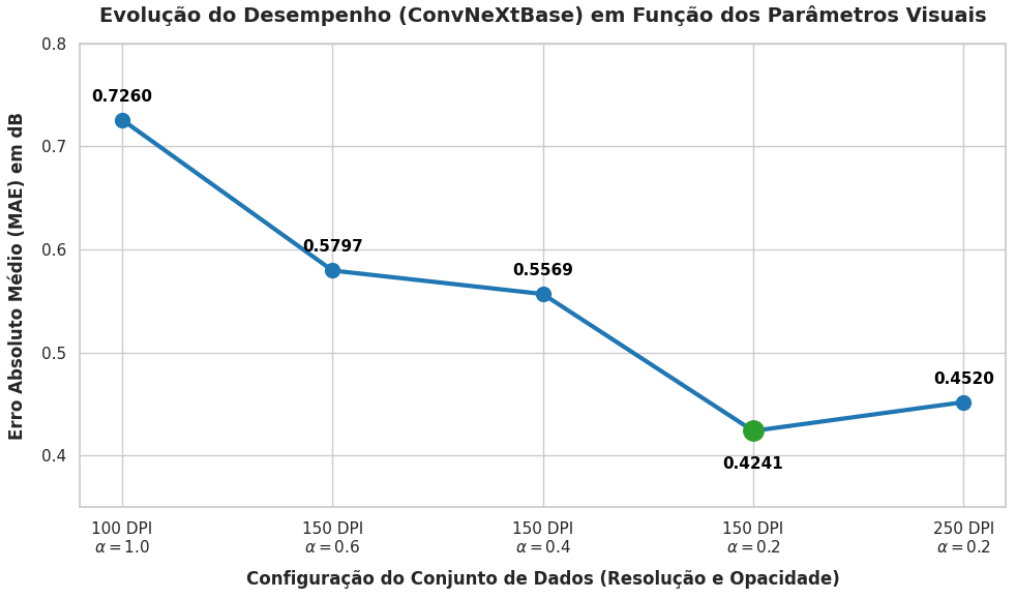
Tabela 9: Resultados para cada modelo utilizando as métricas RMSE, MAE, MSE e R^2 (DPI = 250 e α = 0,2)

Modelo	RMSE (dB)	MAE (dB)	MSE (dB)	R^2
ConvNeXtBase	0,6335	0,4520	0,4014	0,9882
ConvNeXtSmall	0,8284	0,5849	0,6863	0,9799
ConvNeXtTiny	0,8971	0,6328	0,8048	0,9764
DenseNet121	0,9253	0,6716	0,8562	0,9749
DenseNet169	0,8679	0,6305	0,7533	0,9779
DenseNet201	0,7867	0,5710	0,6188	0,9818
EfficientNetB0	1,4982	1,1155	2,2446	0,9341
EfficientNetB3	1,1651	0,8561	1,3574	0,9602
InceptionV3	1,0191	0,7498	1,0387	0,9695
MobileNetV3Large	5,8293	4,4703	33,9810	0,0027
MobileNetV3Small	4,3716	3,3217	19,1108	0,4391
ResNet101	0,8723	0,6414	0,7609	0,9777
ResNet152	0,7285	0,5263	0,5307	0,9844
ResNet50	0,9729	0,7168	0,9465	0,9722
Xception	1,4541	1,0581	2,1144	0,9379

Contrariando a expectativa inicial de melhoria contínua, os resultados da Tabela 9 indicam uma saturação no ganho de desempenho. O MAE do modelo *ConvNeXtBase* regrediu ligeiramente para 0,4520 dB. Esse fenômeno sugere que uma resolução excessivamente alta pode não apenas aumentar o custo computacional, mas também dispersar demasiadamente a representação espacial dos símbolos. Dessa forma, consolida-se a configuração de 150 DPI e α = 0,2 como a melhor configuração para a metodologia proposta.

Para sintetizar, a Figura 18 ilustra a curva de evolução do erro (MAE) do modelo de melhor desempenho, *ConvNeXtBase*, ao longo das cinco configurações de imagens avaliadas. O gráfico evidencia claramente o comportamento de minimização do erro à medida que a resolução aumenta e a opacidade diminui, culminando no ponto ótimo absoluto em 150 DPI e α = 0,2, seguido pela inflexão e leve degradação preditiva ao se adotar a resolução máxima de 250 DPI.

Figura 18: Comparativo de desempenho das arquiteturas para o subconjunto com 150 de DPI e α = 0,2



Fonte Autor

Capítulo 5

Conclusões e Estudos Futuros

Este estudo apresentou uma abordagem baseada em aprendizado profundo de máquinas para estimar OSNR, a partir de diagramas de constelação em receptores ópticos coerentes. Demonstrou-se a viabilidade e a alta precisão da estimação utilizando um conjunto de dados diversificado abrangendo múltiplos formatos de modulação m-PSK e m-QAM. Além disso, avaliou-se um amplo espectro de arquiteturas de CNNs de última geração.

A análise paramétrica revelou que a qualidade da predição é sensível às características visuais de renderização da constelação. Sob condições otimizadas de resolução e opacidade (150 DPI e $\alpha = 0{,}2$), o modelo *ConvNeXtBase* destacou-se com o melhor desempenho, sendo capaz de prever a OSNR com um Erro Absoluto Médio (MAE) de apenas 0,4241 dB e um coeficiente de determinação (R^2) de 0,99. Tais resultados confirmam o potencial das técnicas de visão computacional para a monitorização contínua de desempenho em redes ópticas, oferecendo uma alternativa simplificada, eficiente e não intrusiva em comparação aos métodos tradicionais de medição.

Como desdobramentos lógicos desta pesquisa, pretende-se aumentar ainda mais a robustez preditiva do modelo através da expansão do conjunto de dados, introduzindo maior variabilidade em parâmetros como largura de linha, taxa de símbolo, DPI, α e efeitos de polarização. Além disso, planeja-se explorar arquiteturas de aprendizado profundo baseadas em transformadores, incluindo *Vision Transformers* (ViT), que podem oferecer melhores capacidades de generalização em cenários complexos devido aos seus mecanismos de atenção global.

Publicação

M. F. de Araújo, M. Valadão, A. M. C. Pereira, É. R. da Silva, W. Silva, and A. L. A. Costa, “Deep Learning-Based OSNR Estimation from Constellation Diagrams of DP m-PSK and DP m-QAM in Flexible Coherent Optical Receivers,” in *Anais do XLIII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*, Sociedade Brasileira de Telecomunicações, 2025. doi: 10.14209/sbrt.2025.1571151702.

Referências

[1] International Telecommunication Union (ITU), “Measuring digital development.” Accessed: Oct. 21, 2025. [Online]. Available: https://www.itu.int/hub/publication/D-IND-ICT_MDD-2024-4/

[2] Nokia, “Global network traffic report,” 2024. Accessed: Nov. 09, 2025. [Online]. Available: https://www.nokia.com/asset/213660/

[3] Ericsson, “Ericsson Mobility Report,” Jun. 2025. Accessed: Nov. 09, 2025. [Online]. Available: https://www.ericsson.com/49e9b6/assets/local/reports-papers/mobility-report/documents/2025/ericsson-mobility-report-june-2025.pdf

[4] F. Musumeci *et al.*, “A Survey on Application of Machine Learning Techniques in Optical Networks,” Dec. 2018, [Online]. Available: http://arxiv.org/abs/1803.07976

[5] G. Sanjay Kumar, S. A. Harini, M. Sentil Kumar, and B. A. Therese, “Optical Fiber Communication: A Comprehensive Review,” vol. 19, no. 2, pp. 17–23, doi: 10.9790/2834-1902021723.

[6] D. Brake, “Submarine Cables: Critical Infrastructure for Global Communications,” 2019.

[7] B. Zhu *et al.*, “High-Capacity 400Gb/s Real-Time Transmission over SCUBA110 Fibers for DCI/Metro/Long-haul Networks,” 2022. [Online]. Available: www.openzrplus.org/documents

[8] M. Jaber, M. A. Imran, R. Tafazolli, and A. Tukmanov, “5G Backhaul Challenges and Emerging Research Directions: A Survey,” 2016, *Institute of Electrical and Electronics Engineers Inc.* doi: 10.1109/ACCESS.2016.2556011.

[9] Harshit Gupta, Priyal Gupta, Praveen Kumar, Atul Kumar Gupta, and Praveen Kumar Mathur, “Passive Optical Networks: Review and Road Ahead,” IEEE, 2018.

[10] Rongqing. Hui, *Introduction to fiber-optic communications*. Academic Press, 2020.

[11] M. D. Al-Amri, M. M. El-Gomati, and M. Suhail Zubairy, *Optics in Our Time*. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-31903-2.

[12] F. Musumeci *et al.*, "An Overview on Application of Machine Learning Techniques in Optical Networks," Dec. 2018, doi: 10.1145/3310986.3311000.

[13] H. J. Cho *et al.*, "Constellation-based identification of linear and nonlinear OSNR using machine learning: a study of link-agnostic performance," *Opt. Express*, vol. 30, no. 2, p. 2693, Jan. 2022, doi: 10.1364/oe.443585.

[14] D. Wang *et al.*, "Intelligent constellation diagram analyzer using convolutional neural network-based deep learning," *Opt. Express*, vol. 25, no. 15, p. 17150, Jul. 2017, doi: 10.1364/oe.25.017150.

[15] F. Chollet, *Deep Learning with Python*, 2nd Edition. 2021.

[16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. 2016. Accessed: Dec. 12, 2025. [Online]. Available: http://www.deeplearningbook.org

[17] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," May 27, 2015, *Nature Publishing Group*. doi: 10.1038/nature14539.

[18] S. S. . Haykin, *Neural networks and learning machines*. Prentice Hall/Pearson, 2009.

[19] M. Ekman, *Learning Deep Learning*. 2022.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," 2012, [Online]. Available: http://code.google.com/p/cuda-convnet/

[21] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," Jan. 2017, [Online]. Available: http://arxiv.org/abs/1412.6980

[22] S. Deepa, J. L. Zeema, and S. Gokila, "Exploratory Architectures Analysis of Various Pre-trained Image Classification Models for Deep Learning," *Journal of Advances in Information Technology*, vol. 15, no. 1, pp. 66–78, 2024, doi: 10.12720/jait.15.1.66-78.

[23] H. Anwar, "Selecting the best backbone model: A comprehensive evaluation of deep learning models for satellite image-based land cover classification," *Earth Sci. Inform.*, vol. 18, no. 2, Feb. 2025, doi: 10.1007/s12145-025-01828-7.

[24] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Dec. 2016, pp. 2818–2826. doi: 10.1109/CVPR.2016.308.

[25] F. Chollet, “Xception: Deep Learning with Depthwise Separable Convolutions,” Apr. 2017, [Online]. Available: http://arxiv.org/abs/1610.02357

[26] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Dec. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[27] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Institute of Electrical and Electronics Engineers Inc., Nov. 2017, pp. 2261–2269. doi: 10.1109/CVPR.2017.243.

[28] A. Howard *et al.*, “Searching for MobileNetV3,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, Oct. 2019, pp. 1314–1324. doi: 10.1109/ICCV.2019.00140.

[29] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” Sep. 2020, [Online]. Available: http://arxiv.org/abs/1905.11946

[30] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, “A ConvNet for the 2020s,” Mar. 2022, [Online]. Available: http://arxiv.org/abs/2201.03545

[31] VPItransmissionMaker™ Optical Systems, “VPIPHOTONICS,” https://www.vpiphotonics.com.

[32] A. M. C. Pereira *et al.*, “Classificação Automática de Modulações DP m-PSK e DP m-QAM em Receptores Ópticos Coerentes Flexíveis,” in *Anais do XXXIX Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*, Sociedade Brasileira de Telecomunicações, 2021. doi: 10.14209/sbrt.2021.1570724020.

[33] M. F. de Araújo, M. Valadão, A. M. C. Pereira, É. R. da Silva, W. Silva, and A. L. A. Costa, “Deep Learning-Based OSNR Estimation from Constellation Diagrams of DP m-PSK and DP m-QAM in Flexible Coherent Optical Receivers,” in *Anais*

do XLIII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, Sociedade Brasileira de Telecomunicações, 2025. doi: 10.14209/sbrt.2025.1571151702.

[34] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 91–99, Jun. 2022, doi: 10.1016/j.gltp.2022.04.020.