Detecção de equipamentos de subestação em imagem térmica através de Aprendizado Profundo

Genésio Alves de Araújo Júnior
Curso de Engenharia Elétrica
Universidade Federal de São Carlos,
Brasil
genesio.junior@estudante.ufscar.br

Celso Ap.de França
Departamento de Engenharia Elétrica
Universidade Federal de São Carlos,
Brasil
celsofr@ufscar.br

***Resumo* – A detecção automática de equipamentos em imagens termográficas tem se mostrado uma solução promissora para aprimorar inspeções em subestações elétricas, tornando o processo mais eficiente e reduzindo a necessidade de intervenção humana. Neste trabalho, foi realizada uma análise comparativa entre quatro arquiteturas modernas de detecção de objetos — YOLO (*You Only Look Once*) nas versões YOLOv5 e YOLOv8, *Faster* R-CNN (*Region-Based Convolutional Neural Network*) e RetinaNet — aplicadas a um conjunto de imagens térmicas de subestações. A avaliação considerou precisão, capacidade de generalização, desempenho visual e tempo de treinamento. Os modelos foram treinados e avaliados utilizando métricas de desempenho amplamente empregadas na literatura, como a mAP (*mean Average Precision*), além da análise visual das detecções e do tempo de processamento. Os resultados indicaram que o YOLOv8 apresentou o melhor desempenho global, alcançando os maiores valores de mAP@50 e mAP@50–95, seguido de perto pelo YOLOv5, evidenciando a robustez da família YOLO para imagens térmicas de baixo contraste. O Faster R-CNN obteve desempenho intermediário, enquanto o RetinaNet apresentou métricas inferiores e maior instabilidade visual. Assim, os achados indicam que os modelos da família YOLO são os mais adequados para aplicações de inspeção termográfica, com potencial para uso direto em sistemas automatizados de monitoramento.**

Palavras-Chave: Detecção de objetos; Imagens termográficas; Redes neurais convolucionais; YOLO; R-CNN; RetinaNet.

Abstract* — *Automatic detection of equipment in thermal images has proven to be a promising solution for improving inspections in electrical substations, making the process more efficient and reducing the need for human intervention. In this work, a comparative analysis was carried out among four modern object detection architectures — YOLO (You Only Look Once) in the YOLOv5 and YOLOv8 versions, Faster R-CNN (Region-Based Convolutional Neural Network), and RetinaNet — applied to a set of thermal images from electrical substations. The evaluation considered accuracy, generalization capability, visual performance, and training time. The models were trained and evaluated using performance metrics widely adopted in the literature, such as mean Average Precision (mAP), in addition to visual analysis of the detections and processing time. The results indicated that YOLOv8 achieved the best overall performance, reaching the highest mAP@50 and mAP@50–95 values, followed closely by YOLOv5, highlighting the robustness of the YOLO family for low-contrast thermal images. Faster R-CNN showed intermediate performance, while RetinaNet presented lower metrics and greater visual instability. Thus, the findings indicate that YOLO-based models are the most suitable for thermographic inspection applications, with potential for direct use in automated monitoring systems.

Keywords: Object detection; Thermographic images; Convolutional neural networks; YOLO; R-CNN; RetinaNet.

1. INTRODUÇÃO

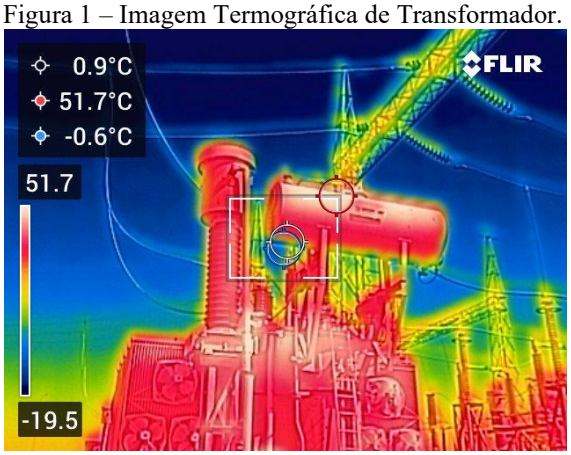
Nos últimos anos, o uso da inteligência artificial (IA) vem se tornando cada vez mais presente no cotidiano das pessoas. A evolução das tecnologias da informação vem possibilitando sua aplicação em diversas áreas importantes, como a indústria, os meios de comunicação e, mais recentemente, as redes elétricas.

No setor elétrico, a aplicação da IA tem se destacado como uma abordagem capaz de aumentar a eficiência operacional, reduzir o tempo computacional e contribuir para a diminuição de custos para concessionárias e consumidores, além de favorecer a operação confiável dos sistemas de energia. As técnicas de IA [1] apresentam maior capacidade de processamento de grandes volumes de dados e melhor desempenho em problemas não lineares e com múltiplas restrições, nos quais métodos tradicionais de otimização numérica podem se tornar lentos ou computacionalmente complexos. Dessa forma, a utilização da IA tem ampliado o nível de automação e o desempenho dos sistemas elétricos nas etapas de geração, transmissão e distribuição de energia.

Nesse contexto, surgem as subestações inteligentes, caracterizadas pela incorporação de sistemas de monitoramento, comunicação e análise de dados capazes de acompanhar continuamente o estado operacional dos equipamentos. Essas subestações [2] utilizam técnicas de monitoramento em tempo real, diagnóstico de falhas online e avaliação de condição dos ativos, substituindo abordagens predominantemente manuais por estratégias baseadas em dados. Como resultado, tornam-se possíveis a identificação antecipada de anomalias, o planejamento mais eficiente das atividades de operação e manutenção e o aumento da segurança, confiabilidade e eficiência econômica do sistema elétrico.

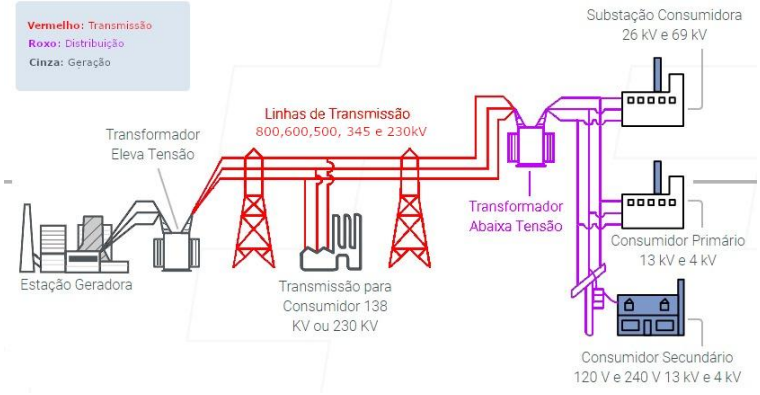
Dentre os diversos tipos de manutenção, destaca-se a inspeção visual por imagem termográfica, importante para a identificação de falhas em equipamentos como isoladores, transformadores e cabos. A Figura 1 apresenta um exemplo de imagem termográfica de um transformador, na qual é possível observar regiões com temperaturas significativamente elevadas em relação ao entorno, caracterizando a presença de um ponto crítico térmico. Esse tipo de anomalia pode estar associado a sobrecargas, falhas de contato ou degradação de componentes, sendo um indicativo relevante para ações de manutenção. Entretanto, a identificação desses pontos críticos ainda é frequentemente realizada de forma manual, tornando o processo mais lento, subjetivo e dependente da experiência do operador.

Figura 1 – Imagem Termográfica de Transformador.



Fonte: ROBOFLOW [3]

Figura 2 - Esquema da Rede Elétrica.



Fonte: Adaptado de ENERGES [5].

De acordo com WANG, L. et al [4], o uso de tecnologia de reconhecimento inteligente por video, baseadas em Aprendizado Profundo (*Deep Learning*, DL) - como Redes Neurais Convolucionais (Convolutional Neural Networks, CNN) - representa um caminho promissor para uma manutenção mais automatizada, possibilitando a detecção automática de defeitos em equipamentos elétricos.

Porém, existem desafios para sua aplicação prática, como a falta de grandes bases de dados de imagens de defeitos, a dificuldade na rotulação manual das amostras e a baixa generalização dos modelos de IA [4]. Diante disso, este trabalho de conclusão de curso tem como objetivo aplicar conceitos de CNNs e técnicas de DL para contribuir na detecção automática de equipamentos em imagens termográficas de subestações elétricas. De forma mais específica, propõe-se uma análise comparativa entre diferentes abordagens modernas de detecção de objetos, com o intuito de avaliar o desempenho, a precisão e a aplicabilidade de cada uma delas nesse tipo de cenário, identificando qual técnica apresenta os melhores resultados frente aos desafios propostos.

2. FUNDAMENTAÇÃO TEÓRICA

2.1 Rede Elétrica

A rede elétrica, ou sistema elétrico de potência, é o conjunto de elementos responsáveis por gerar, transmitir e distribuir energia elétrica até os consumidores finais. De forma geral ela é composta por usinas geradoras, linhas de transmissão, subestações, transformadores e linhas de distribuição [5].

A geração é a etapa em que a energia elétrica é produzida, geralmente em usinas termelétricas, hidrelétricas, eólicas, solares ou nucleares, por meio da conversão de diferentes formas de energia em eletricidade. Na transmissão, essa energia é transportada em altas tensões das usinas até as subestações, o que permite reduzir as perdas elétricas em longas distâncias. Por fim, a distribuição leva a energia das subestações até os consumidores residenciais, comerciais e industriais, com a tensão reduzida para níveis seguros de utilização. Em conjunto, esses três estágios formam o sistema que garante o fornecimento contínuo e confiável de energia elétrica, essencial para o funcionamento das atividades modernas. A Figura 2 traz um exemplo típico de funcionamento da rede elétrica.

Dentro da rede elétrica, as subestações desempenham um papel fundamental no sistema elétrico de potência, atuando como pontos de interligação entre os níveis de geração, transmissão e distribuição. Elas têm como principal função transformar os níveis de tensão (elevar ou baixar a tensão), permitindo o transporte eficiente de energia e sua posterior entrega aos consumidores. Além disso, as subestações possibilitam o controle, a proteção e o seccionamento do sistema, garantindo segurança operacional e continuidade no fornecimento de energia mesmo diante de falhas ou manutenções planejadas.

As subestações modernas [2][4] são constituídas por equipamentos fundamentais para a operação do sistema elétrico, como transformadores de potência e corrente, disjuntores, chaves seccionadoras, isoladores, para-raios, barramentos, bem como instrumentos de medição, proteção e controle. Tradicionalmente, a supervisão e a manutenção desses componentes dependem de inspeções periódicas e da atuação direta de operadores especializados. Nesse contexto, as subestações evoluem para o conceito de subestações inteligentes, nas quais tecnologias de automação, sensoriamento, sistemas de comunicação e inteligência artificial são integradas aos processos de operação e manutenção.

Essa evolução possibilita, por exemplo, o monitoramento contínuo do estado dos equipamentos, o diagnóstico automático de falhas e a realização de inspeções automatizadas por meio de sistemas de visão computacional, câmeras fixas e robôs de inspeção, reduzindo a dependência de avaliações manuais e aumentando a confiabilidade, a eficiência e a segurança do sistema elétrico.

2.2 Inspeção Termográfica

No Brasil, a Agência Nacional de Energia Elétrica (ANEEL) é o órgão responsável por regular e fiscalizar o setor elétrico, incluindo as práticas de operação e manutenção de subestações. Entre as diretrizes estabelecidas, destaca-se a importância da manutenção preditiva, que utiliza métodos de monitoramento contínuo e ensaios não destrutivos para garantir a confiabilidade e segurança do sistema elétrico [6].

A inspeção termográfica é um desses métodos e consiste na análise de imagens infravermelhas para detectar elevações anormais de temperatura em componentes elétricos. Essa técnica permite identificar antecipadamente falhas como mau contato,

sobrecarga, deterioração de isolamentos ou conexões defeituosas, evitando paradas não programadas e reduzindo custos de manutenção.

A aplicação dessa prática é regulamentada pela ABNT NBR 15763:2009, intitulada "Ensaios não destrutivos — Termografia — Critérios de definição de periodicidade de inspeção em sistemas elétricos de potência". A norma [7] estabelece critérios para a execução e a periodicidade das inspeções termográficas, considerando fatores como criticidade do sistema, nível de carga, histórico de falhas e ambiente de operação. Em sistemas elétricos mais suscetíveis a anomalias térmicas ou que operam sob condições severas, como ocorre em diversos equipamentos de subestações, a frequência das inspeções deve ser aumentada, conforme orientado pela norma.

Dessa forma, a termografia regulamentada pela ABNT se consolida como uma ferramenta essencial na manutenção preditiva de subestações, contribuindo para o monitoramento seguro e contínuo dos equipamentos elétricos. Essa padronização favorece maior eficiência operacional, confiabilidade e segurança do sistema elétrico, além de apoiar a modernização das práticas de manutenção no contexto da transição para subestações inteligentes.

2.3 Detecção de Objetos

O processamento de imagens digitais evoluiu significativamente nos ultimos anos, permitindo compartilhamento instantâneo e análise eficiente graças a tecnologias como *Big Data*, computação em nuvem e inteligência artificial. No contexto de visão computacional, a detecção de objetos [8] é definida como a combinação de classificação e localização de objetos em imagens, apresentando aplicações em reconhecimento facial, vigilância e diagnóstico médico.

Nos últimos anos, a detecção de objetos passou por uma evolução significativa. Conforme discutido por M. Vashisht e B. Kumar [8], os métodos tradicionais já apresentavam limitações por dependerem de etapas manuais e classificadores separados, o que reduzia a precisão e a capacidade de generalização. Técnicas clássicas [9] — baseadas em janelas deslizantes (*sliding windows*), como Histograma de Gradientes Orientados (*Histogram of Oriented Gradients*, HOG), Transformada de Característica Invariante à Escala (*Scale-Invariant Feature Transform*, SIFT) e Características Robustas Aceleradas (*Speeded-Up Robust Features,* SURF) — exigiam alto custo computacional e geravam muitas regiões redundantes. Com o avanço das CNNs [9], especialmente após o impacto do modelo conhecido como AlexNet, a extração automática e hierárquica de características se tornou mais eficiente, levando os modelos baseados em DL a superarem de forma consistente as abordagens tradicionais.

A detecção de objetos atualmente segue um *pipeline* que envolve a extração de características, a localização e a classificação dos objetos presentes na imagem [9]. Esse processo é estruturado em três componentes principais, que organizam o fluxo de informações ao longo da rede:

- ***Backbone*:** responsável por extrair os mapas de características da imagem. Geralmente consiste em uma CNN pré-treinada, que produz representações hierárquicas capazes de capturar tanto padrões simples quanto estruturas complexas [9].
- ***Neck*:** combina, refina e integra os mapas de características produzidos pelo *backbone*. Nessa etapa [9], frequentemente utilizam-se estruturas como FPN (*Feature Pyramid Network*) ou suas variantes, como NAS-FPN (*Neural Architecture Search Feature Pyramid Network*) e BiFPN (*Bidirectional Feature Pyramid Network*), que geram representações multiescala essenciais para detectar objetos de diferentes tamanhos.
- ***Head*:** realiza a predição final das caixas delimitadoras (*bounding boxes*) e das classes. O *head* [9] pode produzir predições densas, como nos modelos de um estágio — por exemplo, RetinaNet e a família YOLO (*You Only Look Once*) — ou predições esparsas, típicas dos detectores de dois estágios, como da família R-CNN (*Region-Based Convolutional Neural Network*) [9].

As caixas delimitadoras são utilizadas para indicar a localização e a extensão dos objetos presentes na imagem, sendo representadas por coordenadas que definem sua posição e dimensões. Durante o treinamento, o modelo aprende a estimar essas caixas por meio de um processo de regressão, no qual ajusta continuamente suas previsões para que se aproximem o máximo possível das caixas de referência anotadas no conjunto de dados [9]. A Figura 3 ilustra um exemplo desse processo, na qual os objetos são identificados por uma caixa delimitadora associada à sua respectiva classe, evidenciando como o modelo representa visualmente a detecção e a localização dos componentes analisados.

Figura 3 – Detecção de objetos utilizando caixas delimitadoras.

Fonte: Adaptado de AMJOUD, A. B. et al [9].

Para avaliar a qualidade dessas previsões, utiliza-se a métrica de Interseção sobre União (*Intersection over Union*, IoU), que mede o grau de sobreposição entre a caixa prevista pelo modelo e a caixa real do objeto. Essa métrica é definida como a razão entre a área de interseção e a área de união das duas caixas, demonstrada na equação 1, assumindo valores entre 0 e 1, sendo que valores mais elevados indicam melhor alinhamento entre a predição e a referência. A IoU é amplamente empregada tanto na definição de acertos durante a avaliação dos modelos quanto no cálculo das funções de perda em diferentes arquiteturas de detecção de objetos [9].

$$IoU = \frac{\text{Área da Interseção}}{\text{Área da União}} \quad (1)$$

A partir do IoU, derivam-se métricas amplamente utilizadas na avaliação de detectores de objetos, como a *Average Precision* (AP) e a *mean Average Precision* (mAP). Considerando um valor mínimo de IoU para definir se uma detecção é classificada como correta, a AP é empregada para avaliar o desempenho do detector em uma classe específica, levando em conta a relação entre precisão e revocação. Essa métrica é obtida a partir da área sob a curva precisão–revocação, construída ao variar o limiar de confiança das detecções [9]. Por exemplo, ao avaliar uma classe de equipamento, como isoladores em uma subestação, a AP sintetiza o desempenho do modelo ao equilibrar a quantidade de objetos corretamente detectados com a ocorrência de detecções incorretas, fornecendo uma medida única da qualidade do detector para essa classe.

A mAP corresponde à média dos valores de AP obtidos para todas as classes avaliadas, sendo utilizada como uma métrica global para comparar o desempenho de detectores de objetos. No protocolo de avaliação do conjunto COCO (*Common Objects in Context*), o cálculo da mAP baseia-se em valores de AP previamente obtidos considerando múltiplos limiares de IoU, variando de 0,5 a 0,95. Dessa forma, essa métrica proporciona uma avaliação mais rigorosa do desempenho do modelo, pois exige não apenas a correta detecção dos objetos, mas também maior precisão na localização das caixas delimitadoras.

Durante a inferência, os detectores frequentemente produzem diversas caixas sobre um mesmo objeto. Para evitar duplicações, aplica-se o mecanismo conhecido como a Supressão de Não-Máximos (*Non-Maximum Suppression* , NMS), que mantém apenas a caixa com maior pontuação e descarta predições redundantes cujo IoU exceda um limite pré-definido. Esse procedimento é utilizado amplamente em métodos como da família R-CNN, RetinaNet e nas versões YOLOv1–YOLOv7, sendo considerado essencial para consolidar as saídas do modelo [10].

Outro conceito central na área de detecção de objetos é a distinção entre métodos *anchor-based* e *anchor-free*. Os detectores *anchor-based* utilizam caixas âncora (*anchor boxes*) pré-definidas, com diferentes escalas e proporções, como referência para auxiliar a regressão das caixas delimitadoras. Essa abordagem é adotada por arquiteturas consolidadas, como o Faster R-CNN, o RetinaNet e diversas versões da família YOLO, incluindo YOLOv2, YOLOv3, YOLOv4, YOLOv5 e YOLOv7 [9].

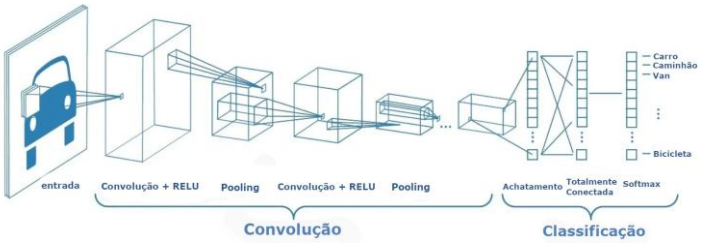
Em contraste, os métodos *anchor-free* eliminam o uso dessas caixas prévias e realizam predições diretas das propriedades dos objetos, como posição do centro, dimensões e classe. Essa estratégia reduz a quantidade de hiperparâmetros e simplifica o processo de treinamento, além de tornar o modelo mais flexível frente a diferentes cenários. Arquiteturas mais recentes da família YOLO, como o YOLOv8, adotam essa abordagem, refletindo uma tendência atual no desenvolvimento de detectores mais leves, simples e adaptáveis [9].

2.4 Redes Neurais Convolucionais

As CNNs são um dos algoritmos de DL mais amplamente aplicados em tarefas de processamento de imagens, como classificação, segmentação e detecção de objetos [11]. Inspiradas no funcionamento do córtex visual humano, as CNNs são capazes de aprender automaticamente características espaciais e hierárquicas a partir das imagens, identificando padrões simples nas camadas iniciais e padrões progressivamente mais complexos nas camadas mais profundas. A Figura 3 ilustra a estrutura geral de uma CNN.

Conforme apresentado na Figura 4, a arquitetura de uma CNN pode ser compreendida como um fluxo de processamento que se inicia com a extração de características da imagem de entrada e termina na etapa de classificação, na qual a rede produz a decisão final. Inicialmente, a imagem de entrada é processada por uma sequência de camadas responsáveis por extrair informações visuais relevantes, reduzindo gradualmente a dimensão espacial dos dados e ampliando o nível de abstração das características aprendidas. Em seguida, essas informações são reorganizadas e utilizadas na etapa de classificação, na qual a rede combina as características extraídas para produzir a decisão final associada à imagem de entrada. Essa representação fornece uma visão integrada do funcionamento da CNN ao longo de toda a rede [11].

Figura 4 – Arquitetura de Rede Neural Convolucional.



Fonte: Adaptado de PURWONO, P. et al [11].

As CNNs são compostas por quatro tipos principais de camadas [11]: camada convolucional, camada de pooling, camada totalmente conectada (*Fully Connected Layer*) e função de ativação (*Activation Function*).

A camada convolucional utiliza filtros (*kernels*) para realizar convoluções sobre a imagem de entrada, extraindo características importantes. O filtro tem dimensões iguais às da imagem, mas com valores diferentes, e percorre a imagem calculando produtos ponto entre seus pesos e os pixels. Esse processo gera um mapa de ativação (*feature map*) em 2D, que representa as características aprendidas. Tal operação é representada pela equação 2.

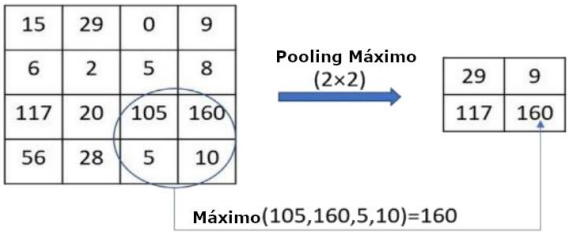
$$\text{Mapa de Ativação} = \text{Entrada} * \text{Filtro} = \sum_{y=0}^{\text{colunas}} \left(\sum_{x=0}^{\text{linhas}} \text{Entrada}(x-p, y-q) \cdot \text{Filtro}(x,y) \right) \quad (2)$$

Na equação 2, x e y representam as coordenadas internas do filtro (*kernel*), indicando a posição de cada elemento dentro dele, enquanto p e q são os deslocamentos que definem onde o filtro está posicionado sobre a imagem de entrada. Assim, para calcular cada valor do mapa de ativação, o filtro percorre a imagem, e para cada posição (p, q), soma-se o produto entre os valores da entrada deslocada e os pesos do filtro, considerando as coordenadas (x, y) do *kernel*.

A camada de *pooling* tem a função de condensar as informações geradas pela camada convolucional anterior, realizando uma redução dimensional (*downsampling*) da imagem. Essa operação diminui o tamanho espacial dos mapas de características, reduzindo a quantidade de parâmetros e o custo computacional, além de tornar o modelo menos sensível a pequenas variações na imagem. O *pooling* pode utilizar valores máximos (*max-pooling*) ou valores médios (*average pooling*) para representar regiões da imagem.

A Figura 5 ilustra o funcionamento do *max-pooling* com janela de tamanho 2×2. Nesse processo, o mapa de características de entrada é dividido em blocos 2×2, e para cada bloco é selecionado apenas o maior valor, que passa a representar toda aquela região na saída. Por exemplo, no bloco composto pelos valores (105, 160, 5 e 10), o valor máximo 160 é mantido, enquanto os demais são descartados. Esse procedimento é repetido para todos os blocos do mapa, resultando em uma nova representação com dimensões reduzidas.

Figura 5 – Camada de *Pooling*.

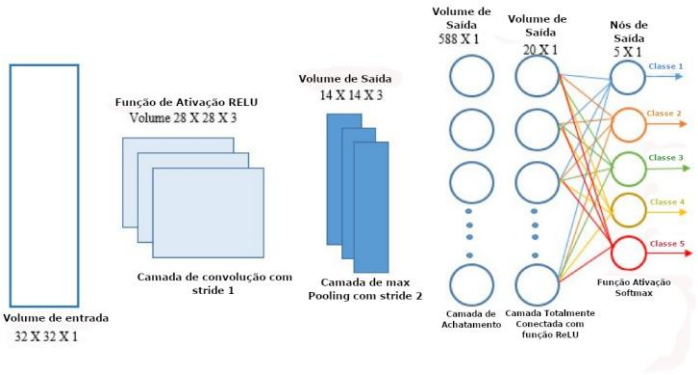


Fonte: Adaptado de PURWONO, P. et al [11].

A terceira camada é a Camada Totalmente Conectada, sendo conhecida como a camada convolucional de saida. Ela geralmente aparece na parte final da rede e recebe dados da última etapa de *pooling* ou da camada convolucional, que são achatados antes de serem processados. Esse processo transforma os resultados em um vetor, permitindo combinações não lineares de características para análises em níveis mais alto.

A Figura 6 apresenta o papel da camada totalmente conectada em uma rede neural convolucional. Nessa etapa, as informações extraídas da imagem ao longo das camadas anteriores são transformadas em um único vetor e utilizadas para tomar a decisão final do modelo. Essa camada combina todas as características aprendidas e calcula a probabilidade de a imagem pertencer a cada classe, sendo responsável pela etapa final de classificação.

Figura 6 – Camada Totalmente Conectada.



Fonte: Adaptado de PURWONO, P. et al [11].

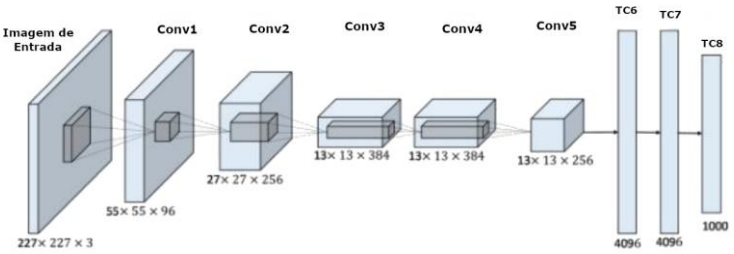
Já a função de ativação é responsável por introduzir não linearidade ao modelo, permitindo que ele aprenda relações mais complexas. Ela transforma valores da saída filtrada em valores dentro de um intervalo, como -1 e 1 ou 0 e 1. A função mais comum é a *ReLU (Rectified Linear Unit)*, usada para extrair as caracteristicas, definida pela equação 3.

$$f(x) = \max(0, x) \tag{3}$$

Entre as arquiteturas de redes convolucionais mais conhecidas, destacam-se a AlexNet, VGGNet, ResNet e as variantes baseadas em CSPNet. Essas redes desempenham um papel fundamental principalmente na extração de características da imagem, influenciando diretamente o desempenho e a eficiência dos detectores modernos.

A AlexNet [11] possui uma estrutura relativamente simples, composta por cinco camadas convolucionais seguidas por uma camada de *pooling* e três camadas totalmente conectadas. As convoluções utilizam o funcionamento da janela deslizante para gerar mapas de características, enquanto a função de ativação ReLU acelera o treinamento ao atuar como um retificador de meia onda. Além disso, o uso de *dropout* nas camadas totalmente conectadas ajuda a reduzir sobreajuste, desativando aleatoriamente alguns neurônios durante o treinamento. A Figura 7 apresenta a arquitetura completa, mostrando a progressão das camadas desde a entrada da imagem até as previsões finais.

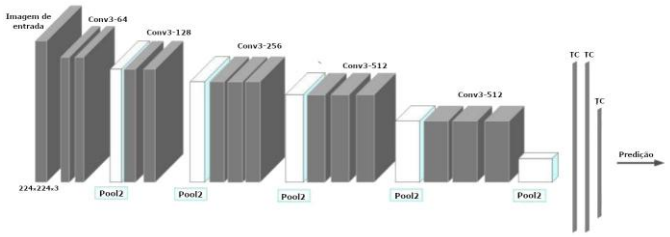
Figura 7 – Arquitetura AlexNet.



Fonte : Adaptado de PURWONO, P. et al [11].

Já a VGGNet [11], sendo a VGG-16 sua versão mais conhecida, representa um refinamento no design arquitetural das CNNs. Essa rede utiliza blocos repetidos de convoluções 3×3 com passo (*stride)* igual a 1, o que significa que o filtro se desloca um pixel por vez sobre a imagem, preservando maior detalhamento espacial. Além disso, é aplicado preenchimento (*padding*), que corresponde à adição de bordas artificiais ao redor da imagem de entrada para manter as dimensões do mapa de características após a convolução. Esses blocos são acompanhados por camadas de pooling *2×2*, que reduzem progressivamente a dimensão espacial dos mapas de características. No total, a VGG-16 apresenta 13 camadas convolucionais e 3 camadas totalmente conectadas. Conforme ilustrado na Figura 8, a resolução inicial de *224×224* é gradualmente reduzida após cada operação de *pooling*, até atingir um vetor final de *7×7×512* antes das camadas responsáveis pela classificação.

Figura 8 – Arquitetura VGGNet.



Fonte : Adaptado de PURWONO, P. et al [11].

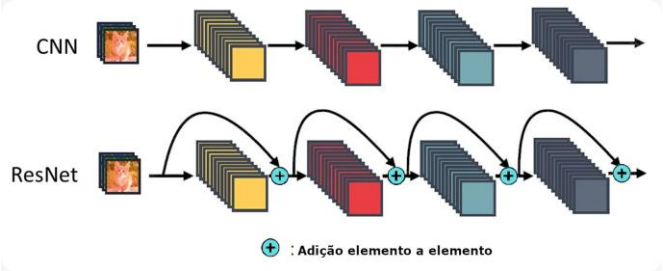
O *Cross Stage Partial Network* (CSPNet) é uma estratégia aplicada em redes convolucionais para melhorar o fluxo de gradientes e permitir o treinamento de arquiteturas mais profundas. Essa abordagem [9] cria conexões parciais entre estágios da rede, fazendo com que parte do mapa de características avance para camadas posteriores enquanto outra parte passa por novos blocos convolucionais.

A CSPDarkNet [9] é uma variante construída a partir da arquitetura Darknet, que segue uma organização sequencial de camadas convolucionais intercaladas com operações de *pooling*, priorizando uma extração de características rápida e direta. A CSPDarkNet aplica a estratégia CSPNet sobre essa base, particionando o mapa de características e incorporando mecanismos de ativação e atenção. Conforme descrito em AMJOUD, A. B.; AMROUCH, M. [9] , sua estrutura utiliza uma série de blocos CSP com número crescente de camadas, cujas saídas são concatenadas com as de blocos anteriores, permitindo que a rede aprenda simultaneamente padrões finos e mais amplos. O resultado final é obtido após camadas convolucionais aplicadas aos mapas produzidos pelo último bloco CSP, formando um backbone adequado para modelos modernos de detecção.

Outra arquitetura amplamente utilizada é a *Residual Network* (ResNet). A principal contribuição dessa arquitetura [9] é a introdução das conexões residuais (*residual connections)*, que permitem que a rede aprenda não apenas o mapeamento direto entre entrada e saída, mas também a diferença residual entre essas representações. Essa inovação resolve o problema do desaparecimento do gradiente, comum em redes muito profundas, e possibilita o treinamento de modelos com dezenas ou até centenas de camadas. Essas conexões de atalho também permitem que a rede "pule" camadas sem perda de desempenho, acelerando a propagação do gradiente e melhorando a estabilidade do treinamento.

A Figura 9 ilustra a diferença entre uma CNN tradicional e uma ResNet, destacando como as conexões residuais permitem que a informação "pule" as camadas, facilitando o fluxo de gradiente e acelerando o treinamento. Esse mecanismo melhora a capacidade de extração de características e aumenta a profundidade efetiva da rede sem comprometer o desempenho. Dentro da família ResNet, uma das versões mais utilizadas é a ResNet-50, composta por 50 camadas treináveis.

Figura 9 – Arquitetura ResNet.



Fonte: Retirado de Ultralytics [12].

A Tabela 1 apresenta um resumo das principais arquiteturas de CNNs citadas neste trabalho, destacando suas principais características e profundidade. Essa organização permite uma visualização comparativa dos aspectos mais relevantes de cada modelo, auxiliando na visualização comparativa e na compreensão da desses modelos.

Tabela 1 – Resumos das Arquiteturas de CNNs.

Arquitetura	Principais características	Profundidade
AlexNet	Convoluções profundas iniciais; ReLU; dropout; normalização local	8 camadas
VGGNet	Blocos de convoluções 3×3; stride 1; padding; pooling progressivo	Variável (ex.: VGG-16)
ResNet	Conexões residuais; aprendizado de mapeamentos residuais	Variável (ex.: ResNet-50)
CSPDarkNet	Estratégia Cross Stage Partial (CSP); particionamento do mapa de características	Blocos CSP

Fonte: Autoria própria.

2.5 Detecção de Objetos em dois estágio

Os detectores de dois estágios surgiram como uma das abordagens mais influentes na evolução das arquiteturas de detecção de objetos, principalmente pela capacidade de alcançar alta precisão mesmo em cenários complexos. Essa família de modelos realiza a detecção em duas etapas distintas [13]: primeiro, uma rede especializada gera propostas de regiões, e depois uma segunda rede classifica e refina essas regiões candidatos. Esse fluxo pode ser visualizado na Figura 10, onde a imagem de entrada passa inicialmente por uma rede *backbone*

convolucional, responsável por extrair mapas de características, e em seguida pela rede *head*, que engloba o módulo de proposta de região e o estágio final de classificação e regressão das caixas delimitadoras.

Figura 10 –Detector de Objetos em dois estágios.

Fonte: Adaptado de MOHAMMED, S. Y. [13].

2.5.1 Redes Neurais Convolucionais baseadas em região

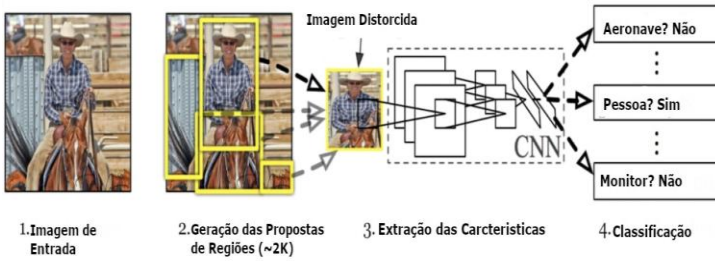
As Redes Neurais Convolucionais baseadas em região (*Region-based Convolutional Neural Network ,* R-CNN) [14], introduziram um novo paradigma ao combinar propostas de região geradas por algoritmos de Busca Seletiva (*Selective Search)* com CNNs para a extração de características e classificação de objetos.

O funcionamento básico do R-CNN pode ser dividido em três etapas:

1. **Geração de Propostas de Região:** o algoritmo de Busca Seletiva gera cerca de 2000 regiões candidatas que podem conter objetos.
2. **Extração de Características:** cada região é redimensionada e processada por uma CNN (geralmente a AlexNet), gerando vetores de características.
3. **Classificação e Regressão:** as características são classificadas por um *Support Vector Machine* (SVM) e refinadas por uma regressão linear para ajustar as caixas delimitadoras.

A Figura 11 apresenta o funcionamento do modelo R-CNN, ilustrando claramente as etapas que compõem sua arquitetura original.

Figura 11 – Modelo R-CNN.



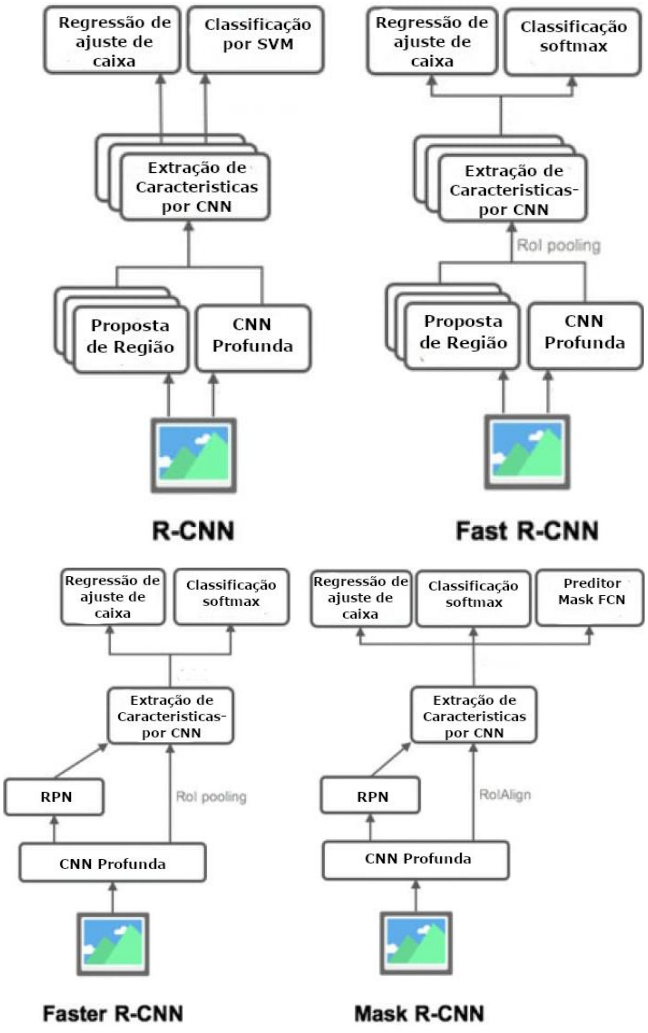
Fonte: Adaptada de Ultralytics [15].

O *Fast* R-CNN [14] surgiu para superar o elevado custo computacional do R-CNN. Em vez de processar cada região separadamente, ele extrai um único mapa de características para toda a imagem e aplica uma camada de RoI (*Region of Interest*) *Pooling,* responsável por normalizar as regiões de interesse geradas pelo algoritmo de Busca Seletiva. O *RoI Pooling* ajusta todas as regiões para um tamanho fixo por meio de operações de *pooling* máximo, permitindo que a mesma CNN seja utilizada de forma compartilhada para realizar a classificação e a regressão de forma conjunta, tornando o processo consideravelmente mais eficiente. Ainda assim, o modelo permanece dependente da Busca Seletiva para gerar as propostas de região.

A principal evolução veio com o *Faster* R-CNN [14], que incorporou uma Rede de Propostas de Regiões (*Region Proposal Network*, RPN) diretamente na arquitetura da rede. A RPN gera automaticamente as propostas de região, compartilhando as mesmas características convolucionais da rede principal. O resultado é um modelo *end-to-end*, capaz de realizar detecção de objetos de forma muito mais rápida e precisa.

Por fim, modelos derivados também expandiram essa família, como o *Mask* R-CNN [14], que adiciona uma ramificação paralela destinada à predição de máscaras em nível de pixel, permitindo que o modelo realize não apenas detecção e localização, mas também segmentação de instâncias com alto nível de detalhamento. A Figura 12 sintetiza as arquiteturas da família R-CNN, destacando de forma visual suas diferenças estruturais.

Figura 12 – Modelos da Familia R-CNN.



Fonte: Adaptado de Ultralytics [15].

Por fim, a Tabela 2 traz um resumo das arquiteturas da família R-CNN discutidas neste trabalho, permitindo uma visualização comparativa de suas diferenças estruturais e de sua evolução ao longo das diferentes versões.

Tabela 2 - Resumo das características da família R-CNN.

Arquitetura	Geração de propostas de região	Extração de características	Classificação e regressão
R-CNN	Selective Search	CNN aplicada a cada região gerada	SVM + regressão linear
Fast R-CNN	Selective Search	Mapa de características único + RoI Pooling	Classificação e regressão conjuntas
Faster R-CNN	RPN integrada à rede	Mapa de características compartilhado	Classificação e regressão conjuntas
Mask R-CNN	RPN integrada à rede	Mapa de características compartilhado	Classificação, regressão e predição de máscaras

Fonte: Autoria própria.

2.5.2 Rede de Pirâmide de Características

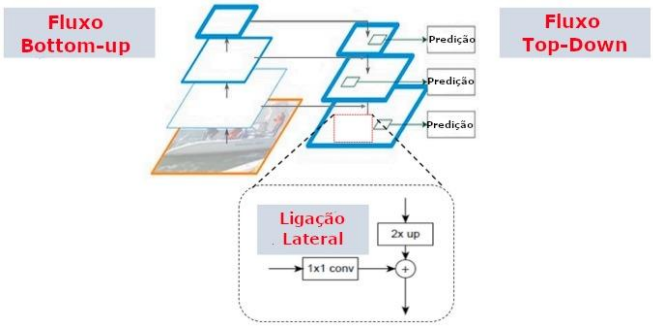
A Rede de Pirâmide de Características (*Feature Pyramid Network,* FPN) é uma arquitetura desenvolvida para superar a limitação das CNNs tradicionais, que tendem a produzir mapas profundos ricos em semântica, porém com baixa resolução, dificultando a detecção de objetos pequenos. Para contornar esse problema, a FPN combina informações de diferentes níveis hierárquicos do *backbone* em uma estrutura piramidal capaz de preservar simultaneamente detalhes espaciais e significado semântico [13].

O FPN [13] opera por meio de dois caminhos complementares: um fluxo *bottom-up*, responsável por gerar mapas de características em diferentes resoluções — preservando detalhes espaciais nas camadas rasas (como Conv2 e Conv3) e capturando semântica profunda nas camadas mais profundas (como Conv5) — e um fluxo *top-down*, que realiza o aumento de resolução (*upsampling*), isto é, a ampliação das resoluções provenientes das camadas mais profundas para reconstruir mapas maiores. Esse processo de *upsampling* torna as características profundas compatíveis em tamanho com as características geradas nas camadas rasas.

Os dois fluxos são conectados por ligações laterais compostas por convoluções *1×1*, alinhando as dimensões dos mapas antes de combiná-los via soma elemento a elemento, produzindo níveis de pirâmide consistentes (P3–P7). Essa fusão multiescala permite que a rede mantenha a precisão espacial necessária para detectar objetos pequenos, ao mesmo tempo em que preserva a riqueza semântica necessária para objetos maiores. A Figura 13 exemplifica o funcionamento da arquitetura FPN e a operação de cada fluxo.

As predições são realizadas independentemente em cada nível da pirâmide utilizando *heads* específicas para classificação e regressão, tornando o processo mais eficiente e modular. O FPN oferece melhor desempenho na detecção de objetos pequenos e grandes, melhora a localização graças às ligações laterais e proporciona ganhos expressivos de acurácia com custo computacional mínimo adicional [13].

Figura 13 – Modelo FPN.

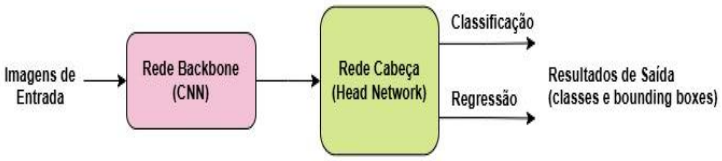


Fonte: Adaptado de MOHAMMED, S. Y. [13].

2.6 Detecção de Objetos em um estágio

Os detectores de um estágio foram desenvolvidos com o objetivo de tornar o processo de detecção de objetos mais rápido e eficiente, especialmente para aplicações em tempo real. Esse tipo de arquitetura [13] elimina a etapa intermediária de geração de propostas, característica dos métodos de dois estágios, e realiza a detecção diretamente a partir dos mapas de características produzidos pela rede *backbone*. A Figura 14 ilustra esse funcionamento simplificado: a imagem de entrada é processada por uma rede *backbone* convolucional, responsável por extrair as informações relevantes, e em seguida enviada diretamente para a rede *head*, que executa simultaneamente as tarefas de classificação e regressão das caixas delimitadoras.

Figura 14 –Detector de Objetos em um estágio.



Fonte: Adaptado de MOHAMMED, S. Y. [13].

Arquiteturas como RetinaNet e diversos modelos da família YOLO seguem esse princípio e têm sido amplamente empregadas em aplicações que demandam respostas rápidas, em função da relação entre velocidade de inferência e desempenho de detecção reportada na literatura [13].

2.6.1 RetinaNet

O RetinaNet é um detector de objetos de um estágio projetado para equilibrar precisão e velocidade. O modelo [13] utiliza como *backbone* redes profundas como ResNet-50 ou ResNet-101 pré-treinadas para extração dos mapas de características com convoluções do tipo *3×3* e *1×1*, seguidas de normalização e funções de ativação como ReLU. Após a extração inicial, o modelo emprega uma FPN com o objetivo de organizar os mapas de características em uma pirâmide multiescala (P3 a P7), permitindo a detecção de objetos em diferentes tamanhos ao combinar detalhes finos das camadas rasas com o contexto semântico das camadas profundas.

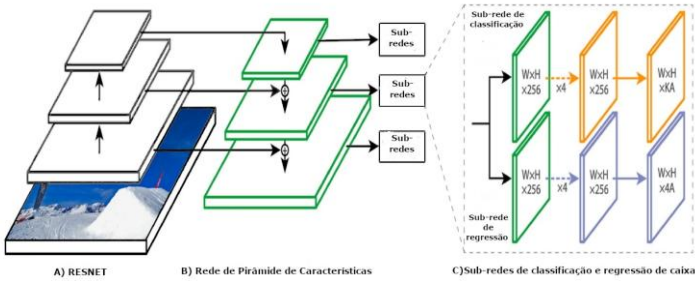
A partir dessa pirâmide, o RetinaNet se divide em duas sub-redes paralelas [13]:

- **Sub-rede de classificação:** composta por quatro convoluções *3×3* (256 filtros), seguida de uma camada final que prevê probabilidades de classe para cada âncora;
- **Sub-rede de regressão:** estruturalmente idêntica, responsável por ajustar as coordenadas das caixas delimitadoras.

A inovação central do RetinaNet é a *Focal Loss*, uma função de perda projetada para reduzir o impacto do desbalanceamento entre classes, suprimindo exemplos fáceis e destacando aqueles difíceis. Essa estratégia permite que o modelo mantenha alto desempenho mesmo em cenários com grande quantidade de âncoras negativos [13].

A Figura 15 apresenta uma visão geral da arquitetura do RetinaNet, destacando (a) o *backbone* ResNet, (b) a FPN utilizada para construção da pirâmide de recursos e (c) as sub-redes de classificação e regressão aplicadas sobre cada nível da pirâmide.

Figura 15 – Modelo RetinaNet.



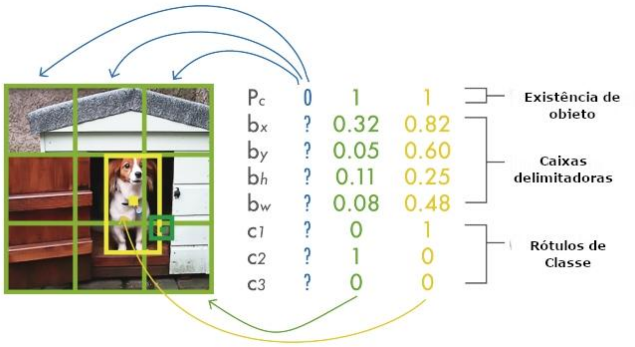
Fonte: Adaptado de MOHAMMED, S. Y. [13].

2.6.2 You Only Look Once (YOLO)

A família YOLO compreende um conjunto de detectores de objetos de um estágio amplamente empregados em aplicações de visão computacional. O método foi originalmente proposto como uma abordagem unificada, capaz de realizar a predição das classes e das caixas delimitadoras em uma única passagem pela rede. Desde sua primeira versão, o YOLO passou por sucessivas evoluções arquiteturais, resultando em diferentes variantes, como YOLOv1, YOLOv2, YOLOv3, YOLOv4, YOLOv5, YOLOv7 e YOLOv8, as quais incorporam novos backbones, módulos de detecção (*neck*) otimizados e distintas estratégias de treinamento [10] [16].

A detecção de objetos proposta pela primeira versão do YOLO, o YOLOv1, é feita ao identificar todas as caixas delitimadoras simultaneamente. Primeiro, a imagem de entrada é dividida em uma grade S×S, onde cada célula é responsável por prever: a probabilidade de existir um objeto, as coordenadas da caixa delimitadora e as probabilidades de classe. Cada predição é formada pelos valores *Pc, bx, by, bh e bw,* que representam, respectivamente, a existência do objeto (*Pc*) , o centro da caixa (*bx* e *by*) e suas dimensões (*bh* e *bw*). Após realizar todas as predições, o modelo usa o NMS para remover duplicadass [10]. A Figura 16 mostra como cada célula da grade produz vetores contendo informações de localização e classificação, evidenciando o funcionamento dessa versão.

Figura 16 – Resultado da predição do YOLOv1.



Fonte:Adaptado de TERVEN et al [10].

A família YOLO evoluiu significativamente desde sua primeira versão, ampliando progressivamente sua capacidade de detecção e eficiência computacional. As versões YOLOv2 e YOLOv3 introduziram melhorias relevantes, como a detecção em múltiplas escalas e ajustes nas caixas âncora. Posteriormente, o YOLOv4 incorporou técnicas como *Mosaic augmentation* e *Self-Adversarial Training*, equilibrando velocidade de inferência e precisão. O YOLOv5 apresentou uma arquitetura modular e altamente adaptável, enquanto versões posteriores, como o YOLOv6, incluíram módulos de concentração bidirecional e estratégias de *Anchor-Aided Training* para aprimorar a localização das características. Já o YOLOv8 removeu completamente o uso de caixas âncora e adotou uma estratégia *anchor-free*, buscando simplificar a arquitetura e reduzir a latência de inferência, aspecto relevante em aplicações que demandam processamento rápido e contínuo, como sistemas automatizados de inspeção. Com o YOLOv10, observa-se ainda a substituição do tradicional NMS por um método de atribuição dual, com o objetivo de reduzir a latência e aprimorar a eficiência do processo de detecção [13].

Apesar das versões mais recentes apresentarem diferenças internas entre si, a estrutura base do YOLO geralemnte apresenta as seguintes caracteristicas [16]:

- ***Backbone*:** O principais backbone nas variantes YOLO incluem ResNet50, ResNet101 e CSPDarkNet53, geralmente já pré-treinados.
- ***Neck***: Geralmente utiliza técnicas como FPN para aprimorar a representação em multi-escala.
- ***Head***: As *heads* do YOLO tradicionalmente utilizam predições em múltiplas escalas com *anchor boxes*, embora algumas versões modernas adotem estratégias *anchor-free*.

A Tabela 3 resume as principais inovações e técnicas empregadas ao longo das variantes do YOLO, permitindo observar a evolução da série YOLO ao longo do tempo.

Tabela 3: Evolução da Família YOLO.

Versão	Principais inovações
YOLOv1	Predição unificada de classes e *bounding boxes* em uma única etapa.
YOLOv2	Uso do backbone Darknet-19 e refinamento de *anchor boxes.*
YOLOv3	Detecção em múltiplas escalas e conexões residuais.
YOLOv4	Introdução de CSPDarknet, PANet e técnicas avançadas de aumento de dados.
YOLOv5	Arquitetura modular e estratégias modernas de treinamento.
YOLOv6	Otimizações arquiteturais para maior eficiência de inferência.
YOLOv7	Estratégias avançadas de treinamento e otimização do processo de detecção.
YOLOv8	Abordagem *anchor-free* e simplificação da arquitetura.
YOLOv10	Estratégia anchor-free com atribuição *dual assignment*, eliminando o uso do NMS e reduzindo a latência de inferência

Fonte: Adaptado de M. L. Ali and Z. Zhang [16].

3. Trabalhos Relacionados

Atualmente, diversas abordagens baseadas em técnicas de DL vêm sendo estudadas para a detecção de equipamentos elétricos de potência em imagens termográficas. Em OU et al. [17], é proposta uma versão aprimorada do *Faster* R-CNN voltada à detecção automática de equipamentos elétricos em subestações a partir de imagens termográficas. O modelo foi desenvolvido para lidar com desafios típicos desse tipo de imagem, como baixa resolução, elevado ruído térmico e fundos visuais complexos. Para isso, os autores adotaram uma modificação na rede VGG16 utilizada como *backbone*, removendo camadas convolucionais mais profundas, de modo a adequar a extração de características às particularidades das imagens infravermelhas. Além disso, foram incorporadas caixas âncora com diferentes proporções, o que favorece a detecção de equipamentos com geometria predominantemente longitudinal, como buchas, isoladores e transformadores de corrente. Os resultados experimentais indicaram alta precisão e boa capacidade de generalização na identificação dos equipamentos em cenários reais de subestações. Contudo, os autores apontam como limitações do método a maior complexidade estrutural do modelo, bem como a dependência de um conjunto de âncoras cuidadosamente ajustado, o que pode dificultar a adaptação do modelo a novos cenários ou conjuntos de dados com características distintas.

Em TANG, Z. e JIAN, X. [18], é proposta uma versão aprimorada do RetinaNet voltada à detecção de equipamentos elétricos em imagens termográficas. Em relação à arquitetura original, que utiliza apenas caixas delimitadoras horizontais e uma FPN convencional, o modelo incorpora três melhorias principais. A primeira consiste no uso de caixas rotacionadas, permitindo melhor adaptação a equipamentos inclinados ou de geometria alongada. A segunda envolve a inserção de um módulo de atenção do tipo *Convolutional Block Attention Module* (CBAM) entre o backbone e a FPN, com o objetivo de aumentar a sensibilidade da rede a regiões relevantes mesmo na presença de elevado ruído térmico. Por fim, os autores integram uma *Path Aggregation Network* (PAN) em conjunto com a FPN, adicionando um fluxo de informações *bottom-up* que reforça a fusão entre níveis rasos e profundos da rede. Como resultado, o modelo apresenta melhorias significativas na detecção em múltiplas escalas, especialmente para objetos pequenos ou com contornos pouco definidos, refletindo-se em ganhos expressivos de mAP em comparação ao RetinaNet original. Contudo, os autores destacam como limitações o aumento do custo computacional e a maior complexidade no processo de anotação, decorrente do uso de caixas rotacionadas, o que pode dificultar sua aplicação em cenários operacionais de grande escala.

Em XUAN, W. et al. [19], é proposto um método de detecção de equipamentos elétricos em imagens termográficas baseado na arquitetura YOLOv5. O modelo foi treinado com um conjunto relativamente reduzido de 880 imagens infravermelhas de subestações, abrangendo sete categorias distintas de equipamentos, e demonstrou boa capacidade de detecção mesmo em cenários com ruído térmico e fundos complexos. Os resultados indicam que a abordagem é eficaz para identificar equipamentos de diferentes tamanhos e formatos, evidenciando a robustez da arquitetura YOLO em aplicações termográficas. Entretanto, os autores apontam que o desempenho do modelo é fortemente dependente da representatividade do conjunto de treinamento, apresentando degradação quando aplicado a equipamentos ou condições não contempladas no *dataset*.

Visando mitigar as limitações associadas à detecção de equipamentos com diferentes orientações em imagens termográficas, KHANDUAL et al. [20] propõem uma abordagem baseada na arquitetura YOLOv8. O método introduz um esquema de regressão orientada, no qual o modelo é capaz de prever não apenas as coordenadas espaciais das caixas delimitadoras, mas também o ângulo de orientação dos objetos, permitindo uma representação mais fiel de equipamentos inclinados ou alongados. Além disso, os autores incorporam um prior de consistência de orientação, que contribui para maior estabilidade das detecções em cenários com baixo contraste térmico e elevado ruído. Os resultados experimentais indicam alta precisão e robustez na identificação dos componentes analisados. Entretanto, o estudo ainda apresenta limitações relacionadas ao tamanho restrito do conjunto de dados e à capacidade de generalização para diferentes tipos de equipamentos, o que sugere a necessidade de bases mais diversificadas para ampliar a aplicabilidade do método em cenários reais de inspeção.

4. Metodologia

Este trabalho foi estruturado de forma a compreender todas as etapas envolvidas no desenvolvimento, treinamento e comparação de diferentes modelos de detecção de objetos aplicados a imagens termográficas de subestações elétricas. O processo metodológico foi dividido em: preparação e caracterização do conjunto de dados, pré-processamento, definição e implementação das arquiteturas de detecção,

configuração do ambiente computacional, treinamento supervisionado e avaliação dos resultados.

O conjunto de dados utilizado neste trabalho foi o *Thermal Substation Components* (v2), disponibilizado na plataforma Roboflow [3]. Esse *dataset* é composto por imagens termográficas reais de subestações elétricas, anotadas com cinco classes de equipamentos amplamente utilizados no sistema elétrico, sendo elas: disjuntores (542 imagens), seccionadoras (474 imagens), transformadores de potência (440 imagens), para-raios (494 imagens) e filtros de ondas (404 imagens). Ao todo, o conjunto contém 2292 imagens, capturadas em condições reais de operação e com resoluções variadas, refletindo a complexidade do ambiente térmico típico de subestações. O Roboflow realizou automaticamente a divisão do conjunto de dados em 92% para treinamento, 5% para validação e 3% para teste, assegurando a separação adequada entre os conjuntos e reduzindo o risco de sobreajuste durante o treinamento dos modelos de detecção.

Antes do treinamento, o conjunto de dados passou por etapas de pré-processamento realizadas na plataforma Roboflow. As imagens foram redimensionadas para 640×640 *pixels* nos modelos da família YOLO e para 300×300 *pixels* nos modelos implementados em PyTorch, como o *Faster* R-CNN e o RetinaNet aprimorado. Nos modelos YOLO, foi utilizada a normalização padrão da biblioteca Ultralytics, enquanto nos modelos em PyTorch aplicou-se a normalização baseada no padrão ImageNet, que ajusta os valores dos canais RGB para tornar o processamento mais estável e compatível com redes pré-treinadas. Além disso, foram aplicadas técnicas de aumento de dados (*data augmentation*), como espelhamento horizontal, pequenas rotações e ajustes leves de brilho e contraste, com o objetivo de aumentar a diversidade das imagens e melhorar a capacidade de generalização dos modelos em imagens térmicas, que normalmente apresentam baixo contraste e pouca textura.

Quatro arquiteturas de detecção foram avaliadas neste trabalho devido à sua relevância na literatura e às diferentes estratégias adotadas para identificar equipamentos em imagens termográficas. Os modelos YOLOv5 e YOLOv8 foram selecionados por realizarem a detecção em uma única etapa, identificando simultaneamente a posição e a classe dos equipamentos, o que favorece uma análise global do desempenho. Além disso, foi avaliada a versão aprimorada do *Faster* R-CNN [17], que utiliza uma rede VGG-16 simplificada como backbone, permitindo maior precisão na localização dos equipamentos. Por fim, foi considerado também o RetinaNet aprimorado [18], que incorpora caixas rotacionadas, modulos de atenção e uma estrutura PAN, com o objetivo de melhorar a detecção em imagens térmicas com baixo contraste. Dessa forma, cada modelo representa uma estratégia distinta de detecção, possibilitando uma análise comparativa clara no contexto de subestações elétricas.

Todos os treinamentos foram realizados no ambiente Google Colab Pro+, utilizando GPU NVIDIA A100 de 40 GB, CUDA 11.x, 12 vCPUs e cerca de 50 GB de RAM. As bibliotecas empregadas incluíram PyTorch 2.x, Torchvision, Ultralytics YOLO, COCO API, Roboflow SDK, Numpy e Matplotlib. Esse ambiente permitiu treinar redes profundas com lotes maiores e realizar inferências complexas sem restrições de memória.

Os modelos passaram por treinamento supervisionado, iniciado com pesos pré-treinados em ImageNet. Cada arquitetura foi treinada por aproximadamente 30 épocas, e as validações foram realizadas automaticamente ao final de cada ciclo. As métricas principais — mAP@50 e mAP@50–95 — foram registradas para fins comparativos.

5. Resultados e Discussões

Os resultados apresentados na Tabela 4 evidenciam diferenças relevantes no desempenho global entre as arquiteturas avaliadas. O YOLOv8 apresentou o melhor desempenho geral, alcançando o maior valor de mAP@0.5 (0,9655) e o maior mAP@0.5:0.95 (0,6313), o que indica elevada capacidade de generalização e maior precisão na localização das caixas delimitadoras sob diferentes limiares de IoU. Em seguida, o YOLOv5 também obteve métricas globais elevadas, confirmando a eficiência da família YOLO para aplicações de inspeção termográfica em subestações. O *Faster* R-CNN apresentou desempenho intermediário, enquanto o RetinaNet obteve os menores valores globais de mAP, mesmo após a incorporação de mecanismos de atenção e fusão de características.

Tabela 4 – Métricas globais dos modelos.

Modelo	mAP@50	mAP@50-95
Faster R-CNN	0.93218	0.56958
RetinaNet	0.8756	0.4947
YOLOv5	0.9564	0.5949
YOLOv8	0.9655	0.6313

Fonte: Autoria própria.

Ao analisar os resultados por classe contidos nas Tabelas 5 e 6, nota-se que o YOLOv5s apresentou desempenho bastante consistente, especialmente para equipamentos de maior porte e geometria bem definida, como transformadores de potência, disjuntores e bobinas de onda. Para essas classes, os valores de AP@0,5 foram elevados (acima de 0,94 em todos os casos), sugerindo que o modelo consegue identificar com alta confiabilidade estruturas térmicas extensas e com padrões visuais bem característicos. No entanto, ao considerar a métrica mais restritiva AP@0,5:0,95, observa-se uma queda mais acentuada em algumas classes, como disjuntores e para-raios, indicando menor precisão na delimitação exata das caixas em diferentes escalas de IoU.

O YOLOv8, por sua vez, manteve desempenho elevado tanto no AP@0,5 quanto no AP@0,5:0,95 em praticamente todas as classes avaliadas. Em particular, destacou-se na detecção de transformadores de potência e bobinas de onda, classes nas quais obteve os maiores valores de AP@0,5:0,95 entre todos os modelos. Esse comportamento sugere maior robustez do YOLOv8 na localização precisa dos equipamentos, mesmo em cenários mais complexos, com variação térmica e presença de ruído.

O *Faster* R-CNN apresentou desempenho satisfatório em classes como bobinas de onda e transformadores de potência, evidenciando sua capacidade de capturar informações espaciais relevantes em equipamentos de maior porte e com padrões térmicos bem definidos. Entretanto, a redução observada no mAP@0,5:0,95 para classes como disjuntores e

seccionadoras indica maior sensibilidade do modelo às variações no posicionamento e na precisão das caixas delimitadoras. Por sua vez, o RetinaNet aprimorado apresentou desempenho relativamente superior na detecção de equipamentos de grande dimensão, como bobinas de onda e transformadores de potência; contudo, não alcançou resultados comparáveis aos dos demais modelos, registrando os menores valores de desempenho entre as arquiteturas avaliadas.

Tabela 5 – Métrica por Classe (AP@0.5).

Classe	YOLOv5	YOLOv8	Faster R-CNN	RetinaNet
Disjuntores	0,94	0,951	0,911	0,695
Seccionadoras	0,933	0,944	0,931	0,773
Transformadores	0,945	0,972	0,911	0,821
Para-raios	0,943	0,955	0,918	0,711
Filtros de Onda	0,995	0,995	1	0,968

Fonte: Autoria própria.

Tabela 6 – Métrica por Classe (AP@0.5:0.95).

Classe	YOLOv5	YOLOv8	Faster R-CNN	RetinaNet
Disjuntores	0,535	0,59	0,543	0,281
Seccionadoras	0,584	0,594	0,543	0,335
Transformadores	0,646	0,717	0,583	0,474
Para-raios	0,534	0,578	0,53	0,406
Filtros de Onda	0,672	0,7	0,625	0,575

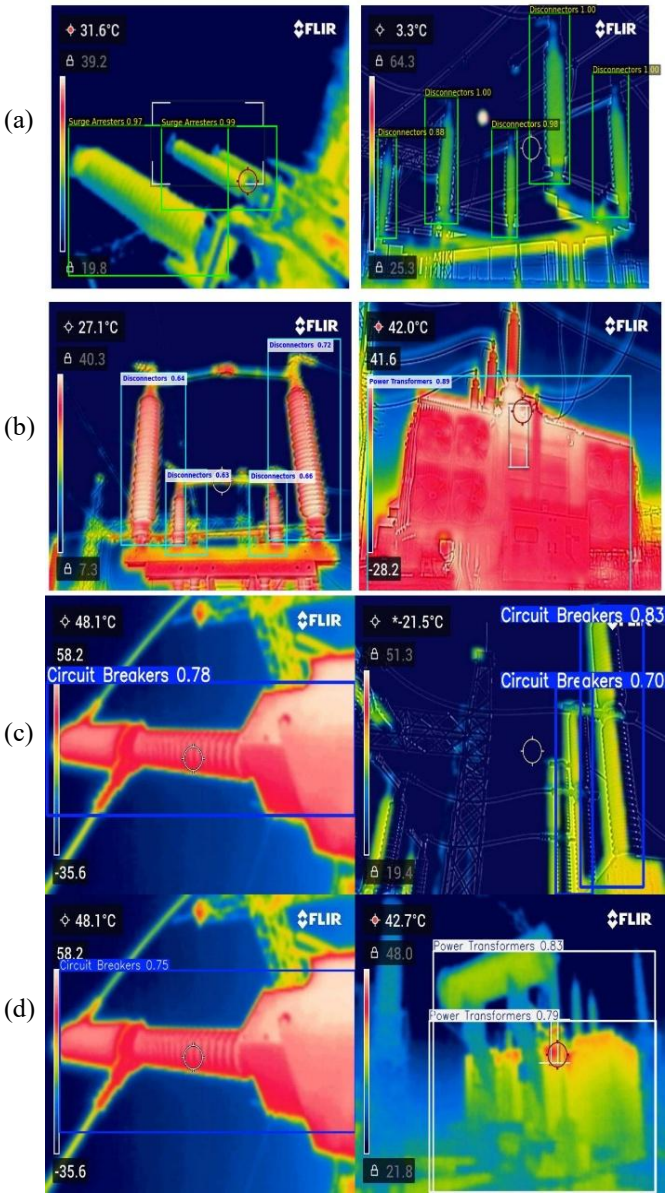
Fonte: Autoria própria.

De modo geral, a análise por classe indica que as diferenças entre os modelos não se limitam às métricas globais, mas também dependem de como cada arquitetura responde às características físicas e térmicas dos equipamentos avaliados. Tanto o YOLOv5s quanto o YOLOv8 apresentaram elevado desempenho na detecção individual de equipamentos de grande porte, como transformadores de potência e bobinas de onda. Entretanto, o YOLOv8 manteve resultados mais equilibrados entre detecção e precisão na localização das caixas delimitadoras em todas as classes, o que explica seu melhor desempenho global.

A análise visual das detecções, apresentada na Figura 17, reforça os resultados quantitativos obtidos nas métricas de avaliação. Observa-se que o YOLOv8 apresentou detecções com elevado nível de confiança, evidenciado pela consistência dos *scores* associados às caixas delimitadoras, mesmo em imagens com forte contraste térmico e presença de ruído. Além disso, as caixas geradas pelo modelo mostraram-se bem ajustadas às regiões de interesse, indicando boa capacidade de localização dos equipamentos. O YOLOv5 também demonstrou desempenho visual consistente, com *scores* elevados na identificação de equipamentos de maior porte, como transformadores e disjuntores. As detecções apresentaram boa estabilidade espacial, sugerindo que o modelo consegue reconhecer corretamente os padrões térmicos característicos desses componentes.

O *Faster* R-CNN exibiu detecções com boa precisão geométrica, ou seja, caixas bem posicionadas sobre os equipamentos identificados, porém com menor frequência de detecções e *scores* visualmente menos uniformes quando comparado aos modelos YOLO. Já o RetinaNet apresentou maior variação nos *scores* associados às detecções, além de instabilidade nas caixas delimitadoras, especialmente em regiões com maior variação térmica. Esse comportamento visual indica menor confiança do modelo nas previsões realizadas, o que está de acordo com seu desempenho inferior observado nas métricas quantitativas.

Figura 17 – Detecções dos modelos propostos .
(a) *Faster* R-CNN; (b) RetinaNet; (c) YOLOv5; (d)YOLOv8.



Fonte: Autoria própria.

De forma geral, os resultados obtidos indicam que os modelos da família YOLO, com destaque para o YOLOv8, apresentam maior adequação para aplicações de inspeção automatizada em subestações elétricas. Esses modelos demonstraram elevada precisão, boa capacidade de localização dos equipamentos e maior robustez frente às variações térmicas presentes nas imagens analisadas. O *Faster* R-CNN, embora tenha apresentado resultados satisfatórios em

algumas classes, mostrou desempenho inferior quando comparado aos modelos de estágio único. Já o RetinaNet não alcançou desempenho comparável aos demais modelos avaliados no conjunto de imagens térmicas utilizado. Dessa forma, os experimentos reforçam que a família YOLO se mostra mais eficiente e consistente para tarefas de detecção de equipamentos elétricos em imagens infravermelhas.

6. Conclusões

Os resultados obtidos neste estudo evidenciam que técnicas modernas de detecção de objetos aplicadas a imagens termográficas apresentam elevado potencial para apoiar processos de inspeção automatizada em subestações elétricas. A partir das métricas quantitativas e da análise visual das detecções, observou-se que os modelos da família YOLO se destacaram em relação às demais arquiteturas avaliadas, apresentando maior precisão, maior estabilidade nas detecções e melhor capacidade de adaptação às variações térmicas e geométricas do conjunto de dados.

Dentre os modelos analisados, o YOLOv8 alcançou os melhores resultados globais, obtendo os maiores valores de mAP@0,5 e mAP@0,5:0,95, além de apresentar detecções visualmente mais consistentes e equilibradas entre as diferentes classes de equipamentos. O YOLOv5, por sua vez, também demonstrou desempenho elevado e robusto, especialmente na detecção individual de equipamentos de grande porte, como transformadores de potência e bobinas de onda, confirmando sua eficácia frente às variações presentes no *dataset* térmico utilizado.

As arquiteturas de dois estágios, representadas pelo *Faster* R-CNN, apresentaram desempenho intermediário, superando o RetinaNet em diversas métricas, porém sem alcançar os resultados obtidos pelos modelos YOLO. O RetinaNet aprimorado, apesar das modificações propostas, apresentou os menores valores de desempenho global e por classe, indicando maior dificuldade em lidar com a complexidade térmica das imagens analisadas.

De modo geral, os resultados reforçam que arquiteturas de um estágio continuam oferecendo a melhor combinação entre desempenho, estabilidade e robustez para a detecção automática de equipamentos elétricos em imagens infravermelhas. Essa característica é especialmente relevante no contexto de manutenção preditiva, no qual a identificação confiável de componentes e potenciais anomalias térmicas é fundamental para o planejamento de intervenções e a redução de falhas operacionais. Além disso, o estudo destaca a importância do pré-processamento adequado, da normalização dos dados e da análise conjunta entre métricas quantitativas e resultados visuais na escolha do modelo mais apropriado. Por fim, os achados apresentados fornecem uma base consistente para trabalhos futuros, como a ampliação do conjunto de dados, a aplicação de técnicas avançadas de aumento de dados e a integração desses modelos em sistemas automatizados de inspeção e monitoramento de subestações.

7. Referências

[1] PANDEY, U.; PATHAK, A.; KUMAR, A.; MONDAL, S. Applications of artificial intelligence in power system operation, control and planning: a review. *Clean Energy*, v. 7, n. 6, p. 1199–1218, 1 dez. 2023. DOI: 10.1093/ce/zkad061.

[2] TAN, Y.; ZHOU, L.; XUE, X.; DUAN, B. Exploration of key technologies for equipment operation and maintenance based on new power systems. *International Journal of Thermofluids*, v. 20, p. 100482, nov. 2023. DOI: 10.1016/j.ijft.2023.100482.

[3] ROBOFLOW. Thermal Substation Components Dataset. Roboflow Universe, 2024. Acesso: 5 dez 2025. Disponível: https://universe.roboflow.com/entername-2sc0s/thermal-substation-components.

[4] WANG, L. et al. Research on application and development of intelligent video recognition technology in power transmission, transformer and distribution system. In: *2023 Asia-Europe Conference on Electronics, Data Processing and Informatics (ACEDPI)*. Prague, Czech Republic: IEEE, 2023.

[5] ENERGES. Diferença entre linha de distribuição e transmissão. Disponível em: https://energes.com.br/diferenca-entre-linha-de-distribuicao-e-transmissao/.

[6] ANEEL – Agência Nacional de Energia Elétrica. *Regras dos serviços de transmissão de energia elétrica – Módulo 4: Prestação dos serviços*. Brasília, 2022. Disponível em: https://www.gov.br/aneel/pt-br/centrais-de-conteudos/procedimentos-regulatorios/regras-de-transmissao.

[7] ABNT – Associação Brasileira de Normas Técnicas. *NBR 15763: Ensaios não destrutivos – Termografia – Critérios de definição de periodicidade de inspeção em sistemas elétricos de potência*. Rio de Janeiro, 2009.

[8] VASHISHT, M.; KUMAR, B. A survey paper on object detection methods in image processing. In: *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*. IEEE, 2020. DOI: 10.1109/ICCSEA49143.2020.9132871.

[9] AMJOUD, A. B.; AMROUCH, M. Object detection using deep learning, CNNs and vision transformers: a review. *IEEE Access*, v. 11, p. 35479–35516, 2023. DOI: 10.1109/ACCESS.2023.3266093.

[10] TERVEN, J.; CÓRDOVA-ESPARZA, D.-M.; ROMERO-GONZÁLEZ, J.-A. A comprehensive review of YOLO architectures in computer vision: from YOLOv1 to YOLOv8 and YOLO-NAS. *Machine Learning and Knowledge Extraction*, v. 5, n. 4, p. 1680–1716, 20 nov. 2023. DOI: 10.3390/make5040083.

[11] PURWONO, P. et al. Understanding of convolutional neural network (CNN): a review. *International Journal of Robotics and Control Systems*, v. 2, p. 739–748, 15 jan. 2023. DOI: 10.31763/ijrcs.v2i4.888.

[12] ULTRALYTICS. O que é ResNet-50 e qual é a sua relevância em visão computacional? Disponível em: https://www.ultralytics.com/pt/blog/what-is-resnet-50-and-what-is-its-relevance-in-computer-vision.

[13] MOHAMMED, S. Y. Architecture review: two-stage and one-stage object detection. *Franklin Open*, v. 12, p. 100322, set. 2025. DOI: 10.1016/j.fraope.2025.100322.

[14] SUMIT; BISHT, S.; JOSHI, S.; RANA, U. Comprehensive review of R-CNN and its variant architectures. *International Research Journal on Advanced Engineering Hub (IRJAEH)*, v. 2, n. 4, p. 959–966, 22 abr. 2024. DOI: 10.47392/IRJAEH.2024.0134.

[15] ULTRALYTICS. What is R-CNN? A quick overview. Disponível em: https://www.ultralytics.com/blog/what-is-r-cnn-a-quick-overview. Acesso em: 01 dez. 2025.

[16] ALI, M. L.; ZHANG, Z. The YOLO framework: a comprehensive review of evolution, applications, and benchmarks in object detection. *Computers*, v. 13, n. 12, p. 336, 14 dez. 2024. DOI: 10.3390/computers13120336.

[17] OU, J. et al. Infrared image target detection of substation electrical equipment using an improved Faster R-CNN. *IEEE Transactions on Power Delivery*, v. 38, n. 1, p. 387–396, fev. 2023.

[18] TANG, Z.; JIAN, X. Thermal fault diagnosis of complex electrical equipment based on infrared image recognition. *Scientific Reports*, 2024.

[19] XUAN, W. et al. Exploration on intelligent detection methods for substation equipment based on deep learning. In: *2024 IEEE 4th International Conference on Power, Electronics and Computer Applications (ICPECA)*. IEEE, 2024. DOI: 10.1109/ICPECA60615.2024.10471052.

[20] KHANDUAL, B. et al. Power system equipment detection in thermal images by deep learning approach. In: *2025 3rd IEEE International Conference on Industrial Electronics: Developments & Applications (ICIDeA)*. IEEE, 2025. DOI: 10.1109/ICIDeA64800.2025.10962932.