

# Detecção Eficiente de Incêndios Florestais em Dispositivos de Borda Utilizando Aprendizado Profundo e Quantização Pós-treinamento

Pedro Marcio Raposo Pereira

*Doutorado*

*Inatel*

Santa Rita do Sapucaí, Brasil

pedro.marcio@inatel.br

**Resumo**—Este artigo apresenta uma abordagem abrangente para a detecção de incêndios florestais, começando pela construção de um modelo de aprendizado profundo baseado na arquitetura MobileNet e utilizando técnicas de aprendizado por transferência. A quantização pós-treinamento foi aplicada para tornar o modelo adequado para implementação em dispositivos de borda, proporcionando eficiência computacional. A implementação prática foi realizada na plataforma *Edge Impulse* (EI), integrando o modelo em placas Arduino. A avaliação do desempenho incluiu métricas detalhadas, como precisão, *recall* e *F1-score*, bem como testes em ambientes reais. Apesar de limitações como o ângulo da câmera, o modelo quantizado demonstrou resultados robustos, superando ligeiramente o modelo em ponto flutuante. A proposta é relevante não apenas para detecção de incêndios florestais, como também em outras áreas como indústrias.

**Index Terms**—Detecção de Incêndios, Aprendizado Profundo, Quantização Pós-treinamento, Dispositivos de Borda.

## I. INTRODUÇÃO

Os incêndios florestais emergiram como uma ameaça crescente e devastadora em escala global, acarretando sérios riscos para o meio ambiente, a fauna e a saúde humana. Incêndios de alta intensidade desencadeiam repercussões de largo alcance, ocasionando danos extensivos às florestas, liberando substâncias nocivas de gases de efeito estufa e perturbando ecossistemas delicados. A intensificação tanto na frequência quanto na gravidade desses eventos demanda a implementação de medidas urgentes para atenuar seus impactos e salvaguardar ecossistemas vulneráveis. Anualmente, mais de 370 milhões de hectares de terras sucumbem a incêndios florestais, resultando na emissão de aproximadamente 1,8 bilhão de toneladas de gases de efeito estufa na atmosfera. As consequências dos incêndios florestais extrapolam o impacto ambiental imediato, repercutindo significativamente na saúde e bem-estar humanos. A poluição do ar induzida por incêndios florestais está associada a cerca de 340.000 mortes prematuras anualmente, predominantemente atribuídas a complicações respiratórias e cardiovasculares [1], [2].

A detecção eficiente de incêndios florestais desempenha um papel crucial na gestão desses eventos, possibilitando intervenções precoces e esforços de supressão destinados a mi-

nimizar danos a propriedades, recursos naturais e vidas humanas. Diversos métodos são empregados para atingir esse objetivo, cada um com características distintas. Tradicionalmente, entre as ferramentas disponíveis para a detecção de incêndios florestais, destacam-se: i) Relatos Públicos: O público frequentemente observa incêndios em áreas remotas antes que os sistemas de detecção oficiais possam reagir. A prontidão e a colaboração do público podem reduzir consideravelmente os tempos de resposta, resultando em esforços de supressão mais eficazes. Adicionalmente, Patrulhas Aéreas equipadas com observadores treinados realizam monitoramento regular sobre áreas de alto risco, oferecendo uma visão abrangente e a capacidade de detectar incêndios que poderiam ser negligenciados por métodos terrestres. Estações de observação estrategicamente posicionadas, equipadas com observadores, também são altamente eficazes em áreas remotas, onde outros métodos de detecção podem ser limitados. A aplicação de tecnologia infravermelha, tanto terrestre quanto aérea, contribui para a detecção e monitoramento de incêndios, mesmo em condições adversas. ii) Detecção Auxiliada por Computador: Sistemas de gestão de incêndios fazem uso de dados meteorológicos em tempo real para avaliar riscos e prever comportamentos de fogo. A detecção de raios, um importante desencadeador de incêndios, é monitorada por sistemas específicos que fornecem alertas em tempo real para facilitar uma resposta rápida em áreas propensas a atividade elétrica atmosférica. Além disso, modelos estatísticos e de aprendizado de máquina são empregados para prever a probabilidade e a localização de incêndios, baseando-se em dados históricos, condições climáticas e características do terreno. Esses modelos oferecem uma valiosa contribuição na identificação de áreas de alto risco, permitindo estratégias proativas de mitigação. Uma vez que um incêndio é detectado, modelos de propagação do fogo são utilizados para prever seu comportamento sob diferentes condições, fornecendo informações cruciais para o planejamento estratégico de supressão e proteção de vidas e propriedades [3].

A proposta deste trabalho visa superar os desafios enfrentados pelos métodos tradicionais em termos de escalabilidade, monitoramento contínuo e detecção em tempo real. Em con-

traste, embora as técnicas auxiliadas por computador forneçam percepções valiosas, frequentemente demandam considerável poder computacional e conectividade extensa, resultando em investimentos elevados. Diante dessas limitações, a presente pesquisa propõe um sistema fundamentado em aprendizado profundo, especialmente adaptado para ambientes com recursos limitados. O sistema proposto almeja ser economicamente viável, fazendo uso de hardware de baixo custo e algoritmos energeticamente eficientes. Prioriza-se, ainda, a otimização da largura de banda, um fator crucial em áreas remotas com conectividade limitada. Com uma vida útil estendida devido ao baixo consumo de energia, o sistema procura superar as frequentes limitações de manutenção associadas a ambientes externos adversos. Para reduzir a dependência de infraestrutura de comunicação de alta largura de banda, o modelo de aprendizado profundo é otimizado para operar eficientemente em dispositivos de borda, como câmeras de vigilância e Veículos Aéreos Não Tripulados (VANTs). Isso possibilita o processamento e a tomada de decisões no local de aquisição de dados, contribuindo para uma detecção de incêndios mais eficiente e autônoma. Dessa forma, são superadas as limitações dos métodos tradicionais e auxiliados por computador na detecção de incêndios florestais. Além de oferecer uma solução eficiente e escalável, a abordagem proposta busca ser economicamente viável, representando um avanço significativo na proteção de florestas e comunidades contra os impactos devastadores dos incêndios florestais.

A literatura existente apresenta diversas abordagens para a detecção precoce de incêndios florestais, utilizando diferentes fontes de dados e técnicas de aprendizado profundo. Um framework de aprendizado profundo para a detecção precoce de incêndios florestais, fazendo uso de imagens multiespectrais provenientes dos Satélites Ambientais Operacionais Geoestacionários (*Geostationary Operational Environmental Satellites*, GOES) foi proposto em [4]. O framework demonstra desempenho promissor na detecção de incêndios florestais, no entanto, é necessária uma avaliação mais aprofundada para analisar sua eficiência computacional e robustez em relação às limitações dos dados de treinamento e possíveis desafios nos dados de imagens de satélite, tais como resolução espacial e cobertura de nuvens. Além disso, o custo de implementação é elevado. Em [5] é proposto utilizar-se *transformers* para a segmentação de incêndios florestais por VANTs, permitindo uma segmentação e identificação focos de incêndios, para isso, o conjunto de dados FLAME, contendo imagens aéreas capturadas por VANTs, foi utilizado. O modelo proposto de classificação alcançou uma precisão de 85,12%, superando outros métodos. Embora o artigo mencione eficiência computacional como uma métrica de avaliação, não há discussão específica sobre a otimização para dispositivos computacionais de baixo desempenho. Um protótipo de sistema funcional capaz de detectar a fumaça de incêndios florestais em até 15 minutos após a ignição, com uma média de menos de um falso positivo por dia por câmera, é proposto em [6]. Esse sistema fundamenta-se no modelo Inception V3 para fazer o reconhecimento de imagens. No entanto, devido a sua

decisão centralizada, necessita de que haja uma infraestrutura de rede que dê vazão aos dados produzidos pelas câmeras, fato que aumenta consideravelmente o custo de implementação e de manutenção do projeto. Uma revisão detalhada de diversas técnicas de aprendizado profundo empregadas para a detecção precoce de incêndios florestais a partir de imagens de VANTs é aferida por [7]. Os autores avaliam o desempenho de várias arquiteturas de Redes Neurais Convolucionais (*Convolutional Neural Networks*, CNN), incluindo AlexNet, GoogLeNet, VGG-Net, e versões modificadas de GoogLeNet e VGG-Net, na detecção de incêndios florestais. Além disso, examinam a eficácia de algoritmos de segmentação semântica, como DeepLab, U-Net, SegNet e CTNet, na identificação de incêndios em imagens e vídeos digitais capturados por plataformas de VANTs. O presente artigo se diferencia ao propor um modelo de classificação de imagens que busca ser de baixo custo e capaz de processar informações localmente, evitando assim a necessidade de uma infraestrutura de rede centralizada e, conseqüentemente, mitigando custos associados à implementação.

Este artigo apresenta um sistema de detecção de incêndios florestais baseado em aprendizado profundo, projetado para operar em dispositivos de borda, como microcontroladores. A metodologia incorpora a rede MobileNet pré-treinada e utiliza técnicas de aprendizado por transferência para adaptar o modelo a um conjunto de dados específico de detecção de incêndios. A integração do sistema completo é posta em forma de diagrama na Figura 1. Um aspecto crucial da implementação foi a quantização pós-treinamento, tornando o modelo adequado para implementação em dispositivos de borda ao converter pesos e ativações para inteiros de 8 bits. A implementação prática ocorreu na plataforma EI, utilizando a Biblioteca Arduino para integração com placas de desenvolvimento baseadas em ARM, como o Arduino Nano 33 BLE. Uma indicação visual da detecção por LEDs comunica o resultado do classificador, sendo azul para o caso normal e vermelho para o caso em que o incêndio foi detectado.

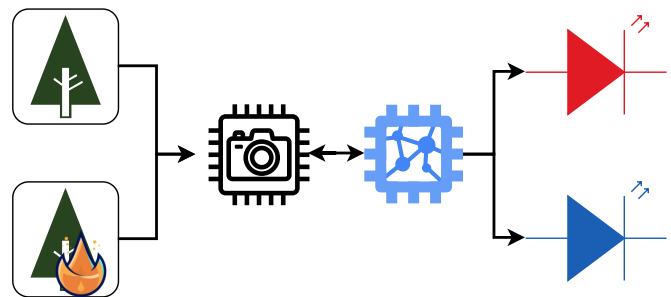


Figura 1. Diagrama do sistema de detecção de incêndios.

O restante do artigo se divide da seguinte forma, na Seção II são discutidos os materiais necessários para a implementação do projeto, na Seção III, detalha-se a metodologia adotada para o desenvolvimento e implementação do modelo de detecção de incêndios florestais. Na Seção IV, apresenta-se os resultados

obtidos durante a avaliação do modelo proposto. Por fim, conclui-se o projeto na Seção IV.

## II. MATERIAIS

A seção a seguir aborda os componentes fundamentais empregados na implementação do projeto de detecção de incêndios florestais. Inicialmente, destaca-se a utilização da plataforma de hardware Arduino Nano 33 BLE Sense e o sensor óptico OV7675. Para otimizar a implementação no Arduino, adotou-se a plataforma EI, construída sobre o framework Keras de aprendizado profundo. Por fim, são mencionados outros elementos, incluindo o ambiente de desenvolvimento e softwares essenciais.

### A. Arduino nano Ble 33

A plataforma de hardware em utilizada é o Arduino Nano 33 BLE Sense, um microcontrolador projetado para aplicações de inteligência artificial. Essa placa apresenta um CPU ARM Cortex-M4 de 32 bits, operando a 64 MHz, para o qual estão acessíveis 1MB de memória *flash* e 256KB de memória Memória de Acesso Aleatório (*Random Access Memory*, RAM), portanto, sua capacidade de processamento é limitada. Além disso, a placa incorpora uma variedade de sensores embutidos, incluindo um sensor inercial de 9 eixos, sensor de umidade e temperatura, sensor barométrico, microfone [8]. O Arduino

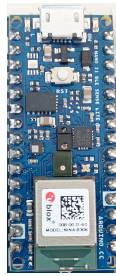


Figura 2. Placa Nano 33 BLE Sense.

destaca-se como uma plataforma de eletrônica de código aberto de hardware e software, principalmente por sua sintaxe simplificada baseada em C++. A Figura 2 demonstra a placa utilizada para o projeto.

### B. Sensor Óptico OV7675

O OV7675 é um sensor de imagem colorido de baixa voltagem que oferece uma solução compacta e eficiente para captar imagens VGA (640x480) de alta qualidade. Ele apresenta uma boa sensibilidade à baixa luz (1800 mV/lux-segundo) [9]. O pequeno tamanho e os requisitos de baixa potência do sensor o tornam adequado para integração em dispositivos compactos, entregando uma alta qualidade de imagem comparado com seu tamanho. A Figura 3 demonstra o sensor utilizado para o projeto.

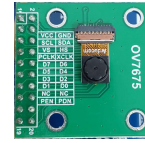


Figura 3. Sensor Óptico OV7675.

### C. Edge Impulse

Para simplificar a implementação do projeto e facilitar a implantação no microcontrolador Arduino, utiliza-se a plataforma EI [10], uma ferramenta construída com base no framework de aprendizado profundo Keras. A interface amigável do EI e seu conjunto abrangente de recursos, que engloba o pré-processamento de dados e capacidades de análise de dados, auxiliam o desenvolvimento e aprimoramento do modelo de detecção.

### D. Outros

Além dos itens anteriormente citados, também se utiliza de um computador com sistema operacional Ubuntu 22.04.3 LTS com os programas: i) Arduino IDE 2.2.1; ii) Visual Studio 1.84; iii) Python 3.11.4; iv) TensorFlow 2.13.0; v) CUDA 12.2; vi) Driver NVIDIA 535.129.03. Os itens ii-vi não são necessários para a implementação do projeto, uma vez que ela pode ser feita toda utilizando apenas a plataforma do EI porém, a análise do modelo fica limitada aos recursos disponíveis na plataforma. Os recursos de *hardware* utilizados foram: i) Processador AMD Ryzen 5600H; ii) GPU NVIDIA RTX 3050; iii) 16 GB de memória RAM.

## III. METODOLOGIA

Nesta seção, detalha-se a metodologia adotada para o desenvolvimento e implementação do modelo de detecção de incêndios florestais. Detalha-se o processo de aquisição e preparação dos dados, a arquitetura do classificador, estratégias de treinamento e transferência de aprendizado, além da etapa crucial de quantização pós-treinamento.

### A. Processamento dos dados

O projeto se inicia pela aquisição de um banco de dados que seja representativo do problema. O banco *Forest Fire Images* [11] é um compilado outros bancos de dados e está publicamente disponível na plataforma Kaggle. No pré-processamento, faz-se uma exclusão de imagens repetidas e uma nova subdivisão dos conjuntos de treinamento e teste na proporção 80%-20%. A divisão final dos conjuntos está disposta na Tabela I.

Em seguida, as imagens passam por um processo de subamostragem, assim, as diferentes dimensões das imagens que compõem cada conjunto são padronizadas na resolução de 96x96. Por fim, é feita a normalização das imagens multiplicando-as por um fator de escala de 1/255, de forma que o valor de todos os pixels estará continuamente distribuído

Tabela I  
DIVISÃO DO BANCO DE DADOS EM CONJUNTO DE TREINAMENTO E  
VALIDAÇÃO.

	fogo	normal	total
treinamento	1667	1657	3324
teste	424	418	842
			4166

no intervalo  $(0, 1)$ , garantindo maior capacidade de aprendizagem do modelo. A Figura 4 exemplifica quatro amostras do conjunto de testes após serem processadas.



Figura 4. Exemplo de imagens após o processamento.

### B. MobileNet

O MobileNet é uma arquitetura leve e eficiente de CNN projetada especificamente para aplicações de visão computacional em dispositivos móveis e embarcados. Ele alcança desempenho notável mantendo baixa complexidade computacional e requisitos de memória, tornando-o adequado para dispositivos com recursos limitados. As CNNs tradicionais empregam convoluções padrão, que realizam uma única operação de convolução em todos os canais de entrada simultaneamente. Em contraste, o MobileNet emprega o uso de convoluções separáveis em profundidade, as quais dividem a convolução padrão em duas etapas distintas: convolução em profundidade e convolução pontual [12]. A convolução em profundidade e a convolução pontual são elementos distintivos da arquitetura MobileNet, fundamentais para sua eficiência computacional. Na convolução em profundidade, um único filtro é aplicado de maneira independente a cada canal de entrada. Em comparação com as convoluções padrão, que operam em todos os canais simultaneamente, essa abordagem reduz o custo computacional, pois cada canal é processado individualmente. Esse processo favorece a extração de características específicas de cada canal, otimizando o aprendizado do modelo. A etapa subsequente, a convolução pontual, desempenha um papel crucial na integração das informações obtidas pela convolução

em profundidade. Nessa fase, uma convolução  $1 \times 1$  é aplicada para combinar os mapas de características gerados pela etapa anterior. O termo “pontual” refere-se ao fato de que essa convolução atua em cada pixel individualmente, sem considerar relações espaciais complexas. Essa abordagem simplificada permite a combinação eficiente de informações, reduzindo ainda mais a complexidade computacional e o número de parâmetros.

Além das convoluções, a arquitetura incorpora técnicas como normalização em lote (*batch normalization*), um procedimento que normaliza as entradas de cada camada, acelerando o treinamento e melhorando a generalização do modelo. A função de ativação ReLU é utilizada para introduzir não-linearidades, aumentando a capacidade do modelo de aprender padrões complexos nos dados. Outro elemento importante são as camadas de *Pooling*, que realizam uma redução nos mapas de características, diminuindo a resolução espacial e, assim, reduzindo a carga computacional. Essa técnica é crucial para preservar as informações mais relevantes enquanto descarta detalhes menos importantes. Entretanto, é importante ressaltar que, devido às restrições de hardware no contexto deste projeto, o modelo completo do MobileNet, com suas 28 camadas, não é viável. Em vez disso, optou-se por empregar apenas 25% de sua profundidade, mantendo a eficácia na extração de características essenciais, enquanto se adequa às limitações de recursos.

### C. Arquitetura do Classificador

O modelo final adota uma arquitetura composta por diversas camadas assim como ilustrado na Figura 5. Após o processamento, a camada de entrada é configurada para receber imagens no formato  $(96, 96, 3)$ , indicando uma resolução de  $96 \times 96$  pixels e três canais de cor RGB. Essa imagem de entrada, então, é processada pela rede MobileNet, resultando em uma saída com dimensões  $(3, 3, 1280)$ . Nesse contexto, as dimensões  $(3, 3)$  representam as informações espaciais da imagem, enquanto a dimensão 1280 denota as características extraídas pela MobileNet. Posteriormente, a saída da MobileNet passa por uma operação de reformulação (*reshape*), preservando a dimensão mais externa e agrupando as dimensões internas, resultando em um formato de  $(9, 1280)$ . Nesse ponto, ocorre um achatamento dos dados, mantendo a dimensão mais externa e agrupando as demais. Em seguida, uma camada densa é aplicada com 16 neurônios, atuando apenas na dimensão mais externa, resultando em uma saída de  $(9, 16)$ . Uma camada de dropout com uma taxa de 10% é então introduzida, promovendo a regularização e prevenindo o sobreajuste do modelo. A camada seguinte, de achatamento (*flatten*), transforma a saída para uma representação unidimensional de 144. Essa saída é direcionada para outra camada densa com dois neurônios, cuja finalidade é agregar informações e permitir a diferenciação entre as duas classes desejadas (presença ou ausência de incêndio florestal).

A Tabela II apresentada a arquitetura do modelo, detalhando as dimensões de saída de cada camada e o número de parâmetros envolvidos. Em suma, essa configuração re-



Figura 5. Arquitetura do Classificador

flete uma estratégia eficaz para extrair e processar características relevantes na detecção de incêndios florestais, enquanto mantém uma eficiência computacional notável, para atender as requisições de hardware do projeto.

Tabela II  
ARQUITETURA DO MODELO.

Camada	Formato de Saída	# Parâmetros
Entrada	(96, 96, 3)	0
MobileNet 25%	(3, 3, 1280)	410208
reshape	(9, 1280)	0
densa	(9, 16)	20496
dropout	(9, 16)	0
flatten	(144)	0
densa	(2)	290
Total:		430994

#### D. Treinamento e Aprendizado por Transferência

O processo de treinamento do modelo de detecção incorpora a técnica de aprendizado por transferência. Essa abordagem aproveita o conhecimento adquirido por um modelo pré-treinado em uma tarefa mais geral e aplica-o a uma tarefa específica relacionada. No caso deste projeto, o MobileNet, treinado anteriormente no conjunto de dados ImageNet, serve como o modelo base. A etapa inicial do treinamento envolve a aplicação de uma política de aumento de dados, que introduz variações nas imagens de treinamento, como espelhamento, ajuste de brilho e redimensionamento aleatório. Essa estratégia visa enriquecer o conjunto de dados de treinamento, proporcionando ao modelo uma exposição diversificada a padrões e contextos, fortalecendo sua capacidade de generalização.

Durante o treinamento, faz-se uma nova subdivisão do conjunto de treinamento, reservando-se 20% para a validação. Inicialmente, a camada que implementa o MobileNet é congelada, portanto, não tem seus pesos atualizados. É utilizado o otimizador Adam, uma taxa de aprendizagem de 0,0005 e a função de perda de entropia cruzada categórica, enquanto monitora a métrica de acurácia. Esse estágio é conduzido ao longo de 20 épocas, com lotes contendo 32 imagens. A utilização de *callbacks* é essencial para monitorar o progresso do treinamento e salvar o modelo que apresenta o melhor desempenho. Após essa fase inicial, o ajuste fino é implementado para adaptar o modelo às características específicas do conjunto de dados de detecção de incêndios florestais. O modelo é recarregado a partir do ponto de melhor desempenho

durante o treinamento inicial. Nesse estágio, ocorre um ajuste na porcentagem de camadas treináveis, descongelando 65% das camadas mais superiores. Esse processo permite ao modelo adaptar-se mais especificamente às nuances da detecção de incêndios florestais, enquanto as camadas anteriores permanecem congeladas para preservar o conhecimento prévio do MobileNet. O modelo é, então, recompilado e treinado por mais 10 épocas com um passo de aprendizagem de 0,000045.

#### E. Quantização Pós-treinamento

Uma vez treinado, o modelo ainda não está pronto para a implementação, isto porque foi treinado em operações de ponto flutuante de 16 bits (float 16). Essas operações são muito custosas para microcontroladores e acarretam grande consumo de energia, além de serem lentas. Assim, foi realizada a quantização pós-treinamento no modelo. Optou-se por uma quantização completa, ou seja, tanto os pesos, quanto as funções de ativação são convertidos para inteiros de 8 bits (int 8), para evitar o uso de operações de ponto flutuante no microcontrolador [13].

A quantização completa de inteiros requer calibração, que envolve estimar o intervalo de valores para cada tensor de ponto flutuante no modelo. Este passo de calibração garante que os valores quantizados representem com precisão os valores originais de ponto flutuante. Para calibrar o modelo, é utilizado um conjunto de dados representativo, que é um pequeno subconjunto dos dados de treinamento escolhido aleatoriamente. Esse processo é conduzido utilizando o TensorFlow Lite, uma extensão da biblioteca TensorFlow otimizada para dispositivos com recursos limitados. Após a calibração, o modelo é convertido para um formato totalmente quantizado, que utiliza apenas operações inteiras. Isso resulta em reduções significativas no tamanho do modelo e na complexidade computacional, reduzindo latência e o uso de memória, tornando-o adequado para implementação em dispositivos de borda. A representação de tensores quantizados em ponto flutuante é dada pela Equação (1)

$$r = (q - z) * s, \quad (1)$$

em que  $r$  é o valor real,  $q$  é o valor quantizado,  $z$  é o ponto de zero e  $s$  é o fator de escala. Para o modelo em questão, o fator de escala é  $s = 0.00390625$  e o ponto de zero é  $z = -128$ .

#### F. Implementação

Após o processo de quantização, por meio da plataforma EI, é possível implementar o modelo. A opção escolhida é a



Biblioteca Arduino, adequada para placas de desenvolvimento Arduino baseadas em ARM. O funcionamento da aplicação é brevemente descrito pelo Algoritmo 1.

#### Algoritmo 1: MODELO DE DETECÇÃO DE INCÊNDIOS

**Requer:** Câmera OV7675, modelo quantizado, biblioteca Edge Impulse

```

1 início
2   Inicializar a câmera OV7675;
3   Configurar os pinos dos LEDs para indicar a
    detecção de incêndio;
4   enquanto Interrupção == Falso faça
5     Esperar 2 segundos;
6     Capturar imagem da câmera OV7675;
7     Redimensionar a imagem capturada para as
        dimensões de entrada do modelo;
8     Normalizar a imagem;
9     Executar o modelo de detecção de incêndios na
        imagem processada;
10    Obter resultados de classificação do modelo;
11    se Probabilidade da classe Incêndio > Limiar
        então
12      Acender o LED vermelho para indicar
          detecção de incêndio;
13      Apagar o LED azul;
14    fim
15    senão
16      Acender o LED azul;
17      Apagar o LED vermelho;
18    fim
19  fim
20  Desinicializar a câmera OV7675;
21 fim

```

O algoritmo de detecção de incêndios florestais inicia inicializando a câmera OV7675, os pinos do LED para indicar detecção de incêndio e o limiar de probabilidade para detecção. O loop principal captura continuamente uma imagem da câmera, a redimensiona para as dimensões de entrada do modelo, realiza a normalização, executa o modelo de detecção de incêndios na imagem redimensionada e obtém os resultados de classificação. Se a probabilidade da classe Incêndio for maior que o limiar escolhido, o LED vermelho é aceso para indicar a detecção de incêndio, enquanto o LED azul é apagado. Caso contrário, o LED azul é aceso e o LED vermelho é apagado. O loop continua até que seja interrompido. O loop termina quando o usuário insere um sinal para interromper a inferência. O algoritmo então desinicializa a câmera e sai.

#### G. Métricas de Desempenho

Ao lidar com a complexidade da detecção de incêndios, a avaliação efetiva do desempenho dos modelos desempenha um papel central na confiabilidade e utilidade do sistema. Neste contexto, diversas métricas são empregadas, cada uma

proporcionando análises específicas sobre o comportamento do modelo.

A matriz de confusão é uma ferramenta que delinea os acertos e erros do modelo em comparação com os resultados esperados. Ela se baseia em 4 valores cruciais: Os Verdadeiros Positivos (VP) que representam as classificações corretas de incêndios; Os Falsos Negativos (FN) que denotam casos em que o modelo falhou ao prever incêndio quando este estava presente; Os Falsos Positivos (FP) que indicam situações em que o modelo erroneamente previu incêndio; Verdadeiros Negativos que refletem as classificações corretas da ausência de incêndios. A Tabela III resume como a matriz de confusão é encontrada e a partir dela é possível obter as demais métricas.

Tabela III  
MATRIZ DE CONFUSÃO

		Classificação	
		Sim	Não
Incêndio	Sim	VP	FN
	Não	FP	VN

A Acurácia é uma métrica geral que revela a desempenho global do modelo, indicando a proporção de classificações corretas. Essa métrica, embora valiosa, pode ser enganosa em cenários com desequilíbrio nas classes, o que naturalmente ocorre na detecção de incêndios, onde as ocorrências de incêndios podem ser significativamente menores em comparação com situações normais. A Acurácia é calculada como o número de previsões corretas dividido pelo total de previsões, de forma definida pela Equação (2)

$$\text{Acurácia} = \frac{VP + VN}{VP + VN + FP + FN} \quad (2)$$

A Precisão emerge como uma métrica vital, especialmente quando falsos positivos são considerados mais prejudiciais que falsos negativos. Em contextos como a identificação de incêndios, a precisão destaca a importância de evitar alarmes falsos, garantindo que as classificações positivas sejam confiáveis. A precisão mede a proporção de previsões positivas que são realmente corretas. A Precisão é calculada pela Equação (3)

$$\text{Precisão} = \frac{VP}{VP + FP} \quad (3)$$

Por outro lado, o *Recall*, também chamado de evocação ou sensibilidade, torna-se crucial quando a consequência de falsos negativos é mais impactante. Na detecção de incêndios, é fundamental garantir que todos os incêndios reais sejam identificados, mesmo que isso resulte em alguns casos de falsos positivos. Definido como o número de verdadeiros positivos dividido pelo total de casos positivos reais, assim, é encontrado pela Equação (4)

$$\text{Recall} = \frac{VP}{VP + FN} \quad (4)$$

O *F1-Score* oferece uma abordagem equilibrada entre a Precisão e o *Recall*. Essa métrica é valiosa quando ambos

desempenham papéis essenciais na avaliação do modelo. É calculado como a média ponderada de precisão e revocação, dada pela Equação (5).

$$F1-Score = \frac{2 \times \text{Precisão} \times \text{Recall}}{\text{Precisão} + \text{Recall}}. \quad (5)$$

#### IV. RESULTADOS

Nesta seção, apresenta-se os resultados obtidos durante a avaliação do modelo proposto para detecção de incêndios florestais.

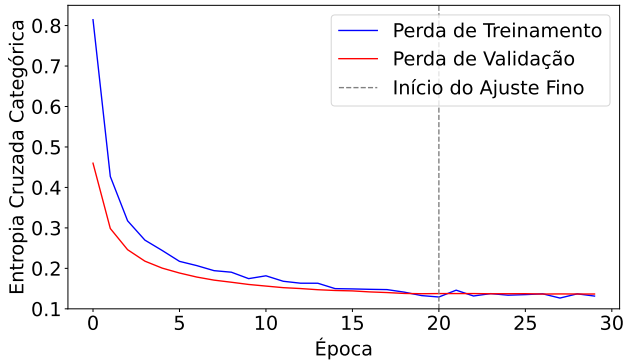


Figura 6. Perda do modelo durante as épocas de treinamento.

As métricas de perda para os conjuntos de treinamento e validação são apresentadas na Figura 6 e indicam que o modelo está aprendendo de maneira eficaz, apresentando desempenho consistente. A perda de treinamento é consistentemente superior à perda de validação, mesmo que a discrepância entre as duas curvas seja relativamente pequena, sugerindo que o modelo não está sobreajustando significativamente os dados de treinamento. A perda de treinamento diminui consistentemente, atingindo um mínimo de 0,1266 na época 27. A perda de validação também apresenta uma trajetória decrescente, embora a uma taxa mais moderada, atingindo um mínimo de 0,1365 na mesma época. A pequena diferença entre as curvas de perda de treinamento e validação reflete a capacidade do modelo de generalizar eficazmente para dados não vistos. Uma diferença substancial indicaria sobreajuste aos dados de treinamento, prejudicando a capacidade do modelo de generalização.

A evolução da acurácia ao longo das épocas é mostrado na Figura 7, os resultados para os conjuntos de treinamento e validação corroboram o bom desempenho do modelo. Novamente, não é verificado sobreajuste, uma vez que as curvas são próximas. A acurácia de treinamento atinge um pico de 95,5% na época 27, enquanto a acurácia de validação alcança 95,2% na mesma época. Esses resultados destacam a capacidade do modelo de distinguir com alta precisão entre imagens de incêndios florestais e não incêndios.

A Tabela IV apresenta uma comparação entre os resultados de dois modelos distintos para a detecção de incêndios: o Modelo Float32 e o Modelo Quantizado sobre o conjunto de testes. Essa comparação é baseada no relatório de

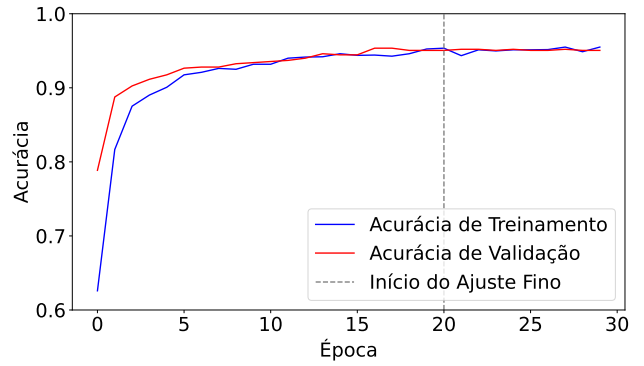


Figura 7. Acurácia do modelo durante as épocas de treinamento.

classificação, que abrange métricas essenciais como precisão, *Recall* e *F1-score* para cada classe, bem como as métricas agregadas de Acurácia, Média Macro e Média Ponderada. Os resultados revelam que o modelo quantizado apresenta

Tabela IV  
ANÁLISE DAS MÉTRICAS DE CLASSIFICAÇÃO PARA OS MODELOS EM PONTO FLUTUANTE E INTEIRO.

Modelo Float32					
	Classe	Precisão	Recall	<i>F1-score</i>	Suporte
	Fogo	0.953	0.958	0.955	424
	Normal	0.957	0.952	0.954	418
Acurácia				0.955	842
Média Macro		0.955	0.955	0.955	842
Média Ponderada		0.955	0.955	0.955	842
Modelo Quantizado					
	Classe	Precisão	Recall	<i>F1-score</i>	Suporte
	Fogo	0.976	0.950	0.963	424
	Normal	0.951	0.976	0.963	418
Acurácia				0.963	842
Média Macro		0.963	0.963	0.963	842
Média Ponderada		0.964	0.963	0.963	842

melhor desempenho em termos de precisão para a classe Fogo, atingindo uma precisão de 0.976 em comparação com a precisão de 0.953 do Modelo float32. No entanto, o Modelo Float32 supera ligeiramente em *recall* para a mesma classe, marcando 0.958 contra 0.950 do modelo quantizado. Essa diferença é refletida no *F1-score*, que é praticamente igual para ambas as classes em ambos os modelos. A acurácia, média macro e média ponderada são consistentemente elevadas em ambos os modelos, indicando uma capacidade geral robusta de classificação. A diferença no desempenho é pequena, mas é significativa visto que o modelo quantizado é muito menor e mais rápido.

Prosseguindo a análise com o modelo quantizado, a Figura 8 detalha o desempenho do modelo para o conjunto de testes por meio da matriz de confusão. É possível notar que 408 casos normais corretamente classificados como normais, 403 casos de incêndio corretamente classificados como incêndio,

21 casos normais incorretamente classificados como incêndio e 10 casos de incêndio incorretamente classificados como normais.

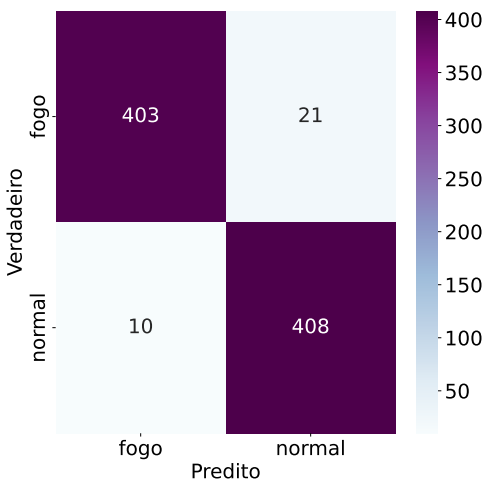


Figura 8. Matriz de Confusão para o conjunto de Testes.

A Figura 9 mostra os resultados da execução do classificador quantizado em 10 imagens aleatórias do conjunto de testes. O classificador identificou corretamente 9 figuras, das quais 8 foram com altos índices de confiança. Isso indica que o classificador é capaz de generalizar bem para dados invisíveis. Em uma análise mais subjetiva, a imagem da segunda coluna e segunda linha foi classificada corretamente, porém com baixa confiabilidade. É possível dizer que a fonte de luz na figura causou confusão no modelo, contribuindo para que a confiança fosse de 58,86%. Da mesma forma que a imagem classificada incorretamente na quarta coluna e primeira fileira possui reflexos intensos na parte inferior da figura. Ainda é possível argumentar que as fumaças localizadas na parte superior podem ter sido confundidas com nuvens ou até mesmo com o próprio céu. No geral, os resultados deste experimento sugerem que o classificador quantizado é um modelo de aprendizado de máquina robusto e confiável. É capaz de classificar com precisão uma ampla variedade de figuras com altos índices de confiança, por outro lado, é necessário aprimorar seu desempenho quando há a presença de fontes luminosas.

Devido à dificuldade em reproduzir os cenários de incêndios para verificar o desempenho do sistema em um caso real, para realizar os testes reais utilizou-se um vídeo de um início de incêndio em uma árvore em uma tela de 11 polegadas (0,28 m) posicionada a uma distância de 0,35 m. O fato de o sensor não ser equipado com uma lente que possibilite um maior ângulo de abertura dificulta a classificação de imagens em uma maior distância. A Figura 11 demonstra um caso em que não há focos de incêndios, já a Figura 10 mostra o caso em que o sistema foi capaz de identificar corretamente o caso de incêndio, assim

o LED vermelho foi acionado, em ambos os casos o limiar escolhido foi de 0,6.

## V. CONCLUSÃO E PROPOSTAS FUTURAS

Este estudo apresentou um modelo de detecção de incêndios florestais baseado em aprendizado profundo, utilizando a arquitetura MobileNet e empregando técnicas de aprendizado por transferência e quantização pós-treinamento por meio da plataforma EI. O modelo foi treinado e avaliado em um conjunto de dados diversificado, demonstrando um desempenho robusto na distinção entre imagens de incêndios florestais e situações normais. Os resultados obtidos durante a fase de treinamento e validação demonstraram que o modelo quantizado atingiu altos níveis de precisão, *recall* e *F1-score* para ambas as classes, Fogo e Normal. Além disso, a comparação entre o modelo quantizado e o modelo em ponto flutuante mostrou que, apesar da redução na complexidade do modelo quantizado, houve uma melhoria marginal na precisão, indicando a viabilidade dessa abordagem para aplicações em dispositivos embarcados.

A implementação prática, com testes utilizando um vídeo simulando incêndio, corroborou a eficácia do sistema, acionando corretamente o LED vermelho em casos de detecção de incêndio. Embora os resultados deste estudo sejam promissores, algumas limitações foram identificadas, incluindo restrições no ângulo de visão da câmera e desafios na realização de testes em ambientes reais.

Uma proposta futura para mitigar essas limitações envolve a exploração de tecnologias de câmeras com ângulos de visão mais amplos, permitindo uma cobertura mais abrangente e melhorando a capacidade do sistema de detecção. Além disso, considerar o uso de múltiplas câmeras ou uma abordagem de visão multiocular pode proporcionar uma visão mais holística do ambiente. Para contornar as dificuldades na realização de testes em ambientes reais, a utilização de ambientes controlados pode ser uma estratégia viável. Esses ambientes simulados podem reproduzir cenários realistas de incêndios florestais, permitindo testes extensivos em condições controladas.

Outro ponto a ser pesquisado é a utilização de sensores de temperatura, umidade e gases. A implementação de um modelo que utilize dados diversificados permite criar um sistema mais robusto e preciso e ainda, sim, ficar comprometida a um sistema de baixo poder computacional e menor custo.

Além disso, a tecnologia de detecção de padrões desenvolvida neste projeto pode ser utilizada em cenários diversos, como monitoramento de áreas urbanas e instalações industriais. A flexibilidade do modelo de aprendizado de máquina, aliada à sua capacidade de adaptação a diferentes domínios, amplia o alcance de aplicação dessa tecnologia para além da detecção de incêndios florestais.

## REFERÊNCIAS

- [1] U. Nations, "As Wildfires Increase, Integrated Strategies for Forests, Climate, and Sustainability Are Ever More Urgent," <https://www.un.org/en/un-chronicle/wildfires-increase-integrated-strategies-forests-climate-and-sustainability-are-ever-0>, 2023, [Online; acessado em 20-Novembro-2023].



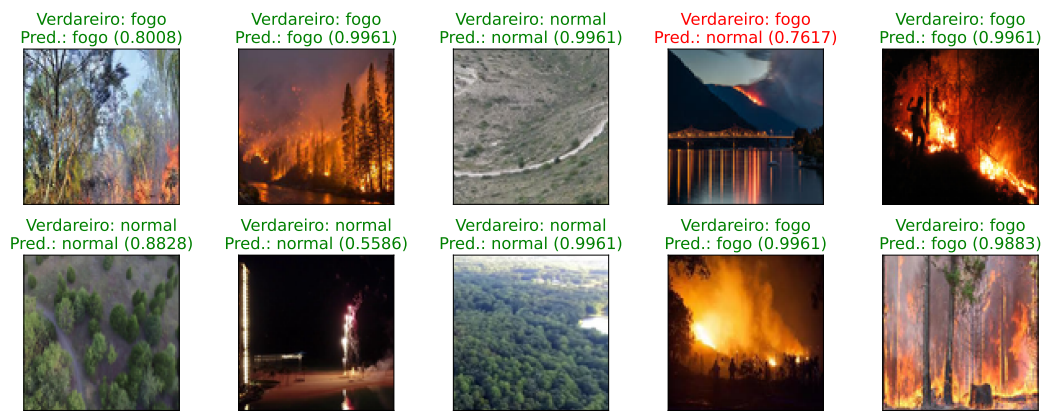


Figura 9. Exemplo de classificações para o modelo quantizado

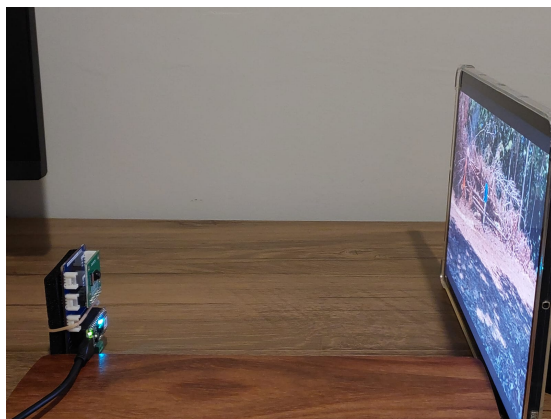


Figura 10. Teste para o caso negativo, portanto o LED azul está aceso.

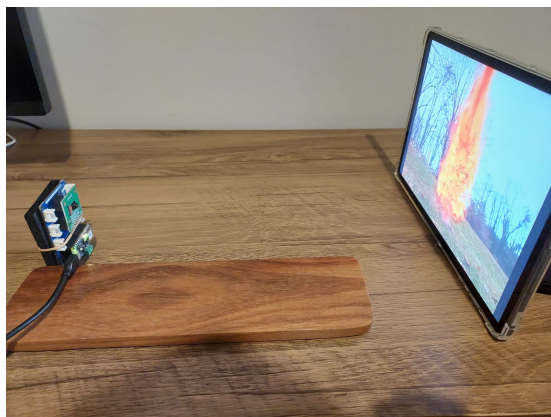


Figura 11. Teste para o caso positivo, portanto o LED azul está aceso.

- [2] U. N. E. Programme, "Number of Wildfires to Rise by 50% by 2100 and Governments Are Not Prepared," <https://www.unep.org/news-and-stories/press-release/number-wildfires-rise-50-2100-and-governments-are-not-prepared>, 2022, [Online; acessado em 20-Novembro-2023].
- [3] G. of the Northwest Territories, "Detecting wildfire," <https://www.gov.nt.ca/ecc/en/services/wildfire-operations/detecting-wildfire>, [Online; acessado em 20-Novembro-2023].

- [4] N. T. Toan, P. Thanh Cong, N. Q. Viet Hung, and J. Jo, "A deep learning approach for early wildfire detection from hyperspectral satellite images," in *2019 7th International Conference on Robot Intelligence Technology and Applications (RiTA)*, 2019, pp. 38–45.
- [5] R. Ghali, M. A. Akhloufi, and W. S. Mseddi, "Deep learning and transformer approaches for uav-based wildfire detection and segmentation," *Sensors*, vol. 22, no. 5, 2022. [Online]. Available: <https://www.mdpi.com/1424-8220/22/5/1977>
- [6] K. Govil, M. L. Welch, J. T. Ball, and C. R. Pennypacker, "Preliminary results from a wildfire detection system using deep learning on remote camera images," *Remote Sensing*, vol. 12, no. 1, 2020. [Online]. Available: <https://www.mdpi.com/2072-4292/12/1/166>
- [7] A. Bouguettaya, H. Zarzour, A. M. Taberkit, and A. Kechida, "A review on early wildfire detection from unmanned aerial vehicles using deep learning-based computer vision algorithms," *Signal Processing*, vol. 190, p. 108309, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0165168421003467>
- [8] Arduino, "Arduino Nano 33 BLE Sense," <https://docs.arduino.cc/hardware/nano-33-ble-sense>, 2023, [Online; acessado em 20-Novembro-2023].
- [9] OmniVision Technologies, "OV7675 A23A Datasheet," <https://www.ovt.com/products/ov07675-a23a>, 2023, [Online; acessado em 21-Novembro-2023].
- [10] EdgeImpulse, "EdgeImpulse: A Cloud-Based Machine Learning Platform for Edge Devices," <https://www.edgeimpulse.com/>, 2023, [Online; acessado em 21-Novembro-2023].
- [11] Mohnish Sai Prasad, "Forest Fire Images," <https://www.kaggle.com/datasets/mohnishsaiprasad/forest-fire-images>, 2021, [Online].
- [12] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017.
- [13] TensorFlow, "Post-training quantization," [https://www.tensorflow.org/lite/convert/post\\_training\\_quantization](https://www.tensorflow.org/lite/convert/post_training_quantization), 2023, [Online; acessado em 22-Novembro-2023].