



Sistema de Detecção de Incêndios Florestais

Uma abordagem que integra visão computacional,
redes neurais convolucionais e sistemas embarcados

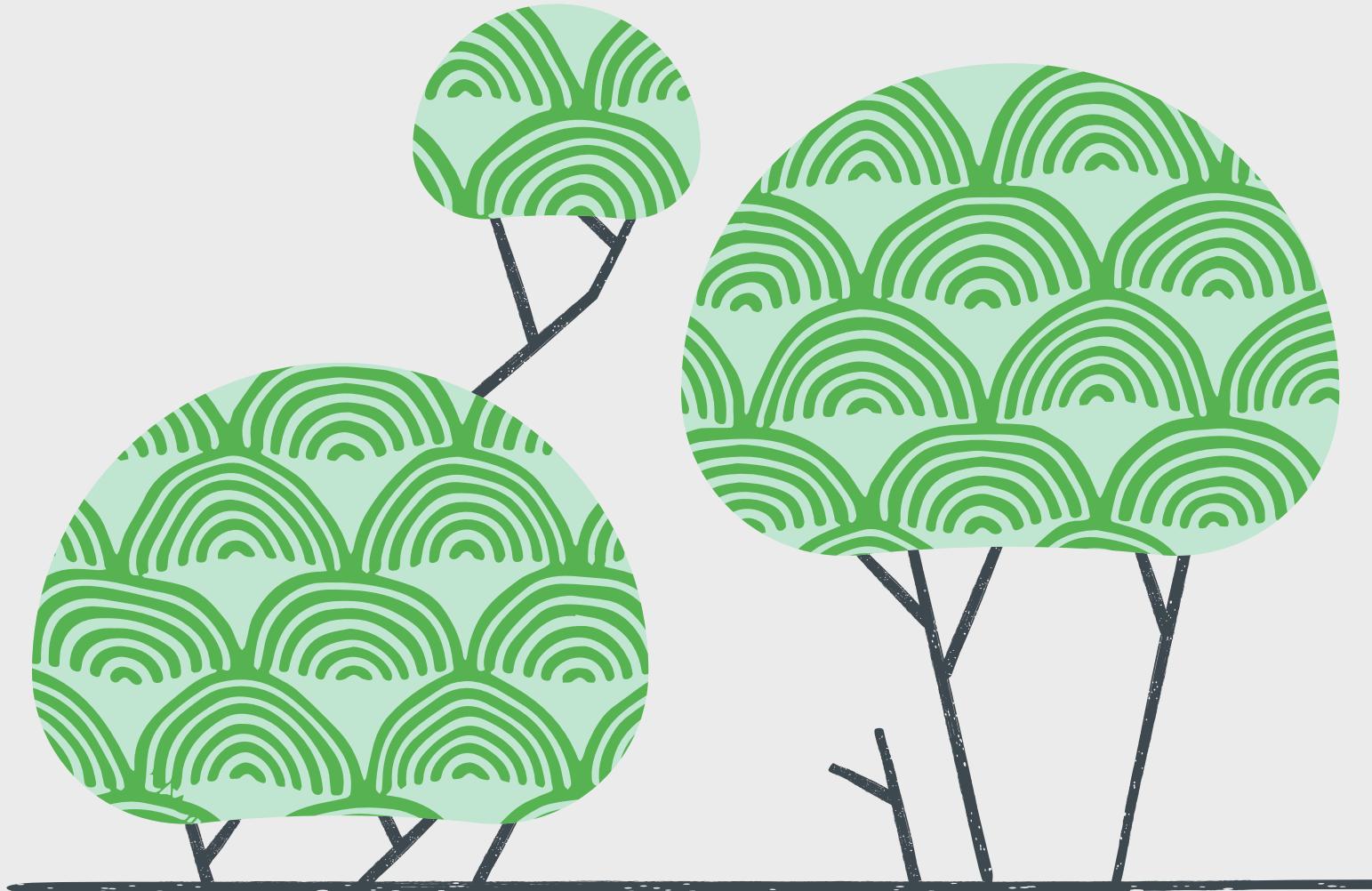
Acadêmicos: Lucas A. de Souza Batista, Erick Felipe Borges, Felipe Rodrigues Benatto

Orientador: André L. M. Santana

Curso: Ciências da Computação

Campus: Vila Olímpia - Noturno

Sumário



- Introdução
- Fundamentação teórica
 - Técnicas de Visão Computacional
 - Estratégias de Validação
 - Redes Neurais Convolucionais
 - Validação de Classificadores
 - Arquitetura de Sistemas Embarcados
 - Transmissão de Dados Sem Fio
- Metodologia
- Análise e Resultados
 - Resultados dos Modelos
 - Cenários
 - Resultados gerais
- Conclusões
- Trabalhos Futuros

Introdução

O planeta está cada vez mais quente e seco marcado pelos impactos do aquecimento global. Tais impactos são, cada vez mais, intensificados pela destruição ocasionada pelos incêndios florestais [DENNISON et al. 2014]. Estes eventos causam diversos problemas, não só ambientais, mas também econômicos e, acima de tudo, representam um grande perigo para a vida. Um incêndio florestal de larga escala pode avançar por dias ou até mesmo semanas, destruindo tudo que encontra no seu caminho.

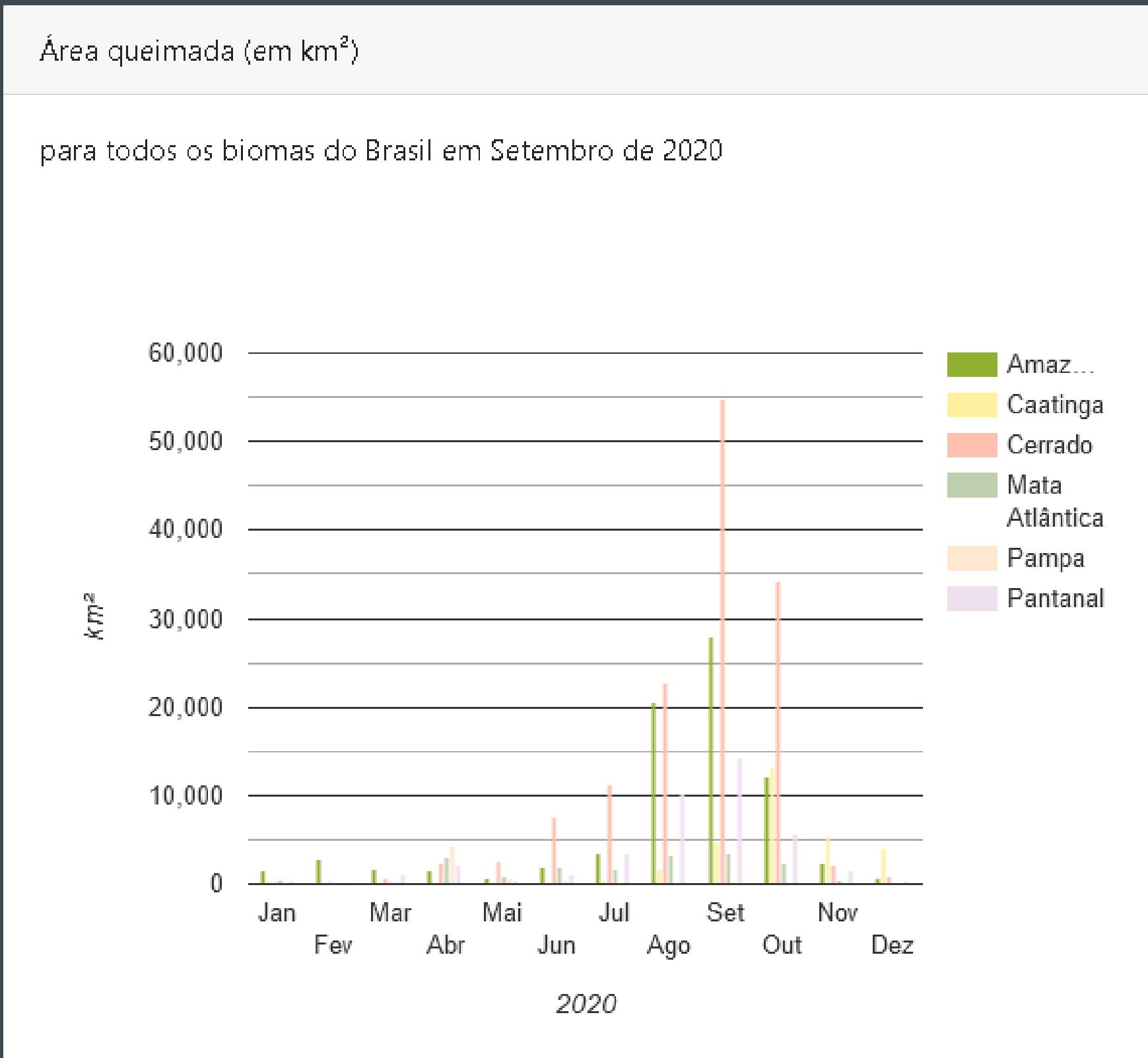
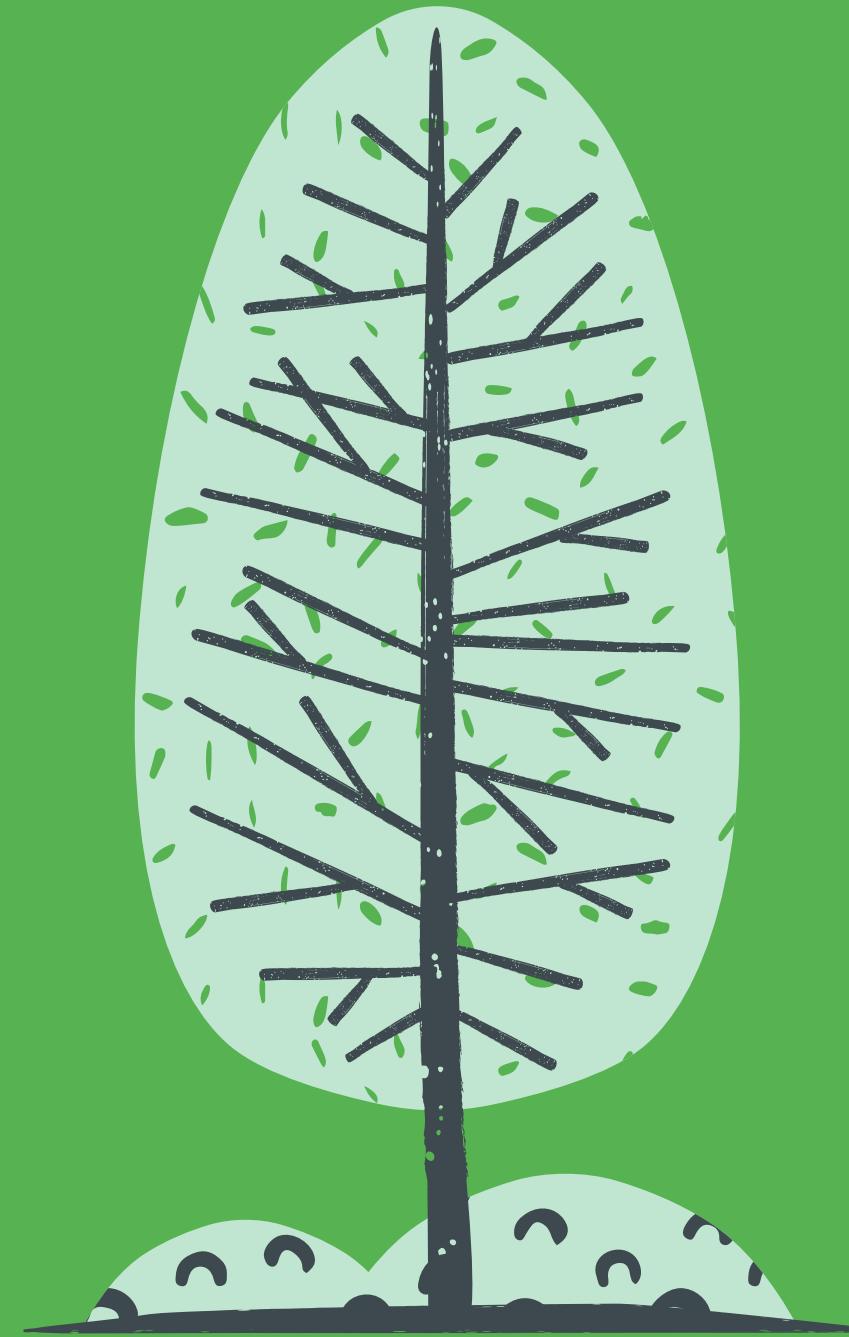


Figura 1 - Gráfico de colunas que relaciona o mês com a área queimada (em km²) para todos os biomas do Brasil em Setembro de 2020.

Fonte: Instituto Nacional de Pesquisas Espaciais (INPE), 2021.





Objetivo

O objetivo deste trabalho é desenvolver e avaliar a eficiência de um protótipo de sistema de detecção de incêndios florestais. O protótipo de sistema, que é composto por um nó sensor, equipado com um sensor de temperatura, um sensor de gás, uma câmera e um transmissor-receptor, e por um gateway experimental, que possui um transmissor-receptor, irá realizar a predição da existência de fogo, ou não, na área monitorada, isto é, por meio da união de conceitos de computação em nuvem, visão computacional, IoT e Inteligência Artificial. A principal ideia é, além de reunir alguns dados de uma determinada área por meio do uso dos sensores descritos acima, transportar estes dados por meio de uma tecnologia de transmissão sem fio de longa distância até a nuvem para que, dessa forma, estes dados sejam analisados e, posteriormente, distribuídos.

Visão Computacional



De acordo com Backes a área de estudo de visão computacional tem o objetivo de atribuir as máquinas a capacidade da visão, o objetivo é desenvolver a percepção visual humana em uma máquina. A capacidade visual humana é capaz de filtrar objetos ou regiões de interesse, texturas, cores e formas, além da capacidade de relacionar essas informações com outras imagens vistas anteriormente.

A visão computacional surgiu na mesma época dos primeiros computadores e a área de pesquisa evoluiu a medida que mais recursos computacionais se tornaram disponíveis. Nos dias de hoje, a pesquisa em visão computacional possuí uma grande variedade de algoritmos de alto desempenho em problemas específicos, como o reconhecimento facial, análise de formas, segmentação em imagens médicas, além de ser uma área sempre aberta a novas ideias.

No entanto ainda é muito a se evoluir para alcançar a capacidade de se emular a visão e percepção humana.

Técnicas de Visão Computacional



No processo de desenvolvimento de um sistema de visão computacional é necessário seguir uma trilha com várias etapas. São elas:

- Aquisição: Aqui a tentativa é simular a função dos olhos através de captura de imagens. Utilizamos normalmente câmeras, filmadoras etc.
- Processamento de imagens: O objetivo é melhorar a qualidade da imagem, removendo ruídos, suavizando a imagem.
- Segmentação: Determinar as regiões de interesse da imagem.
- Extração de características: Definimos aqui um conjunto de características do objeto de interesse, funciona como uma identificação da imagem.
- Reconhecimento de padrões: É responsável por classificar as imagens com base em suas características.

Transmissão de Dados Sem Fio

No desenvolvimento de um projeto IoT, a transmissão de dados e informações é recorrente, devemos considerar que na maior parte das vezes os sistemas estão embarcados e o racionamento de energia é crucial para o funcionamento do dispositivo por semanas e até anos sem a necessidade recarga ou manutenção de baterias. Considerando isso a utilização das low-power wide-area network (LPWAN) ganhou uma grande popularidade na área, devido a permissão de comunicação sem fio de longas distâncias por um baixo custo de energia. O objetivo deste trabalho é implementar um sistema de monitoramento em real em áreas de florestas para a prevenção de incêndios, utilizaremos uma adaptação do sistema LoRa para o envio de informações de sensores para um gateway. Nosso gateway, irá alimentar um banco de dados que será consumido por um aplicativo para exibição em tempo real de nossas áreas de monitoramento.

Long Range (LoRa)

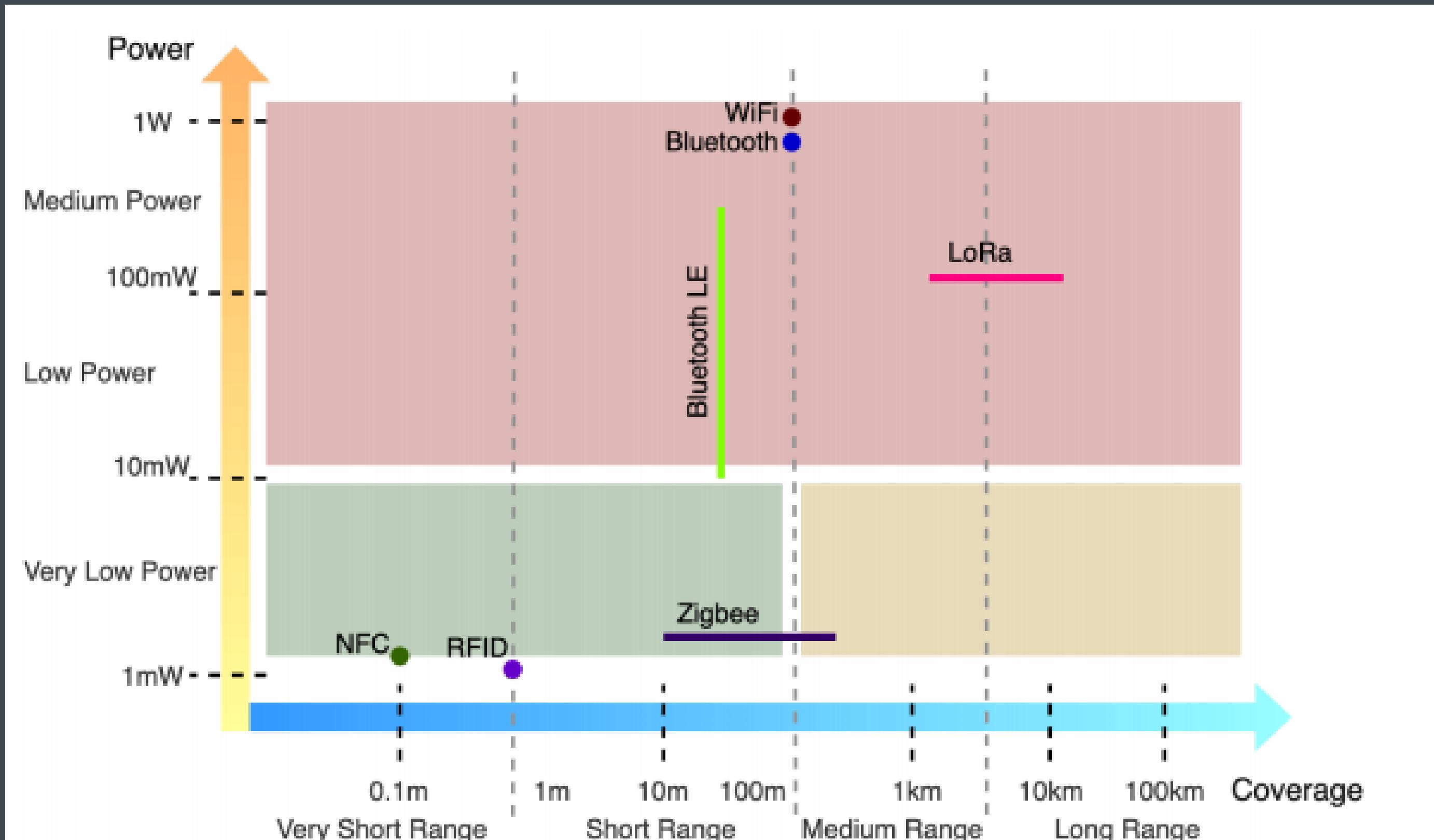


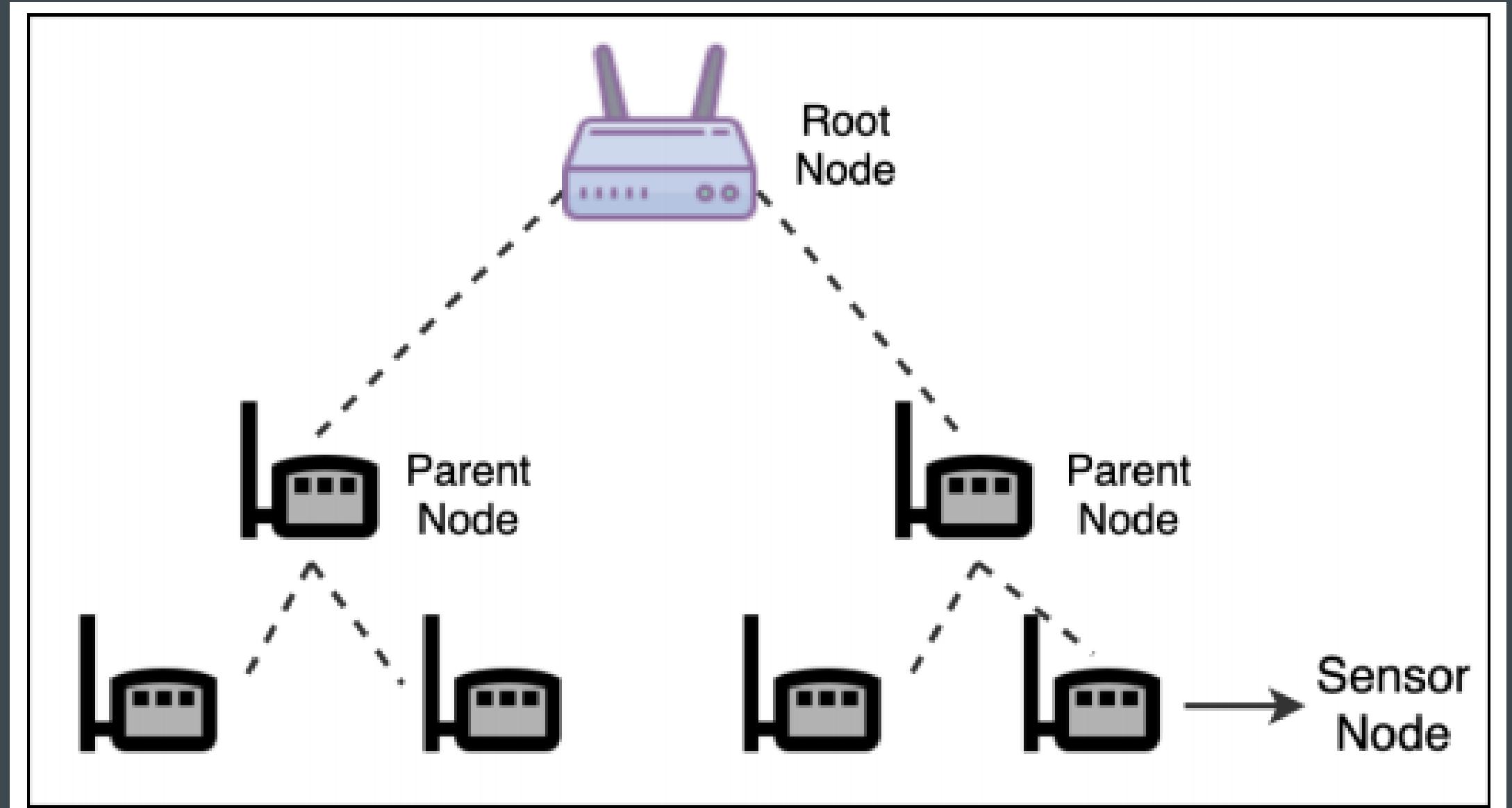
Figura 2 - Consumo de energia e alcance de cobertura para diferentes tecnologias sem fio.

Fonte: Tonghao Chen, 2019

Long Range (LoRa)

Ao utilizar uma RSSF baseada em LoRa podemos reduzir significativamente o número de dispositivos além de trazer benefícios adicionais, como por exemplo nos custos de construção e manutenção que são proporcionais ao número de dispositivos. Se o número de dispositivos diminuir, os custos de construção e manutenção também serão reduzidos. Outro ponto relevante seria o multi-hop com uma rede com um pequeno número de dispositivos resultando em um número de saltos máximo ou médio inferior, enquanto o número de saltos diminui teremos uma queda no consumo de energia, pois teremos uma menor quantidade de dispositivos envolvidos em cada caminho. Com a redução no consumo de energia, podemos estender a vida útil da rede, também reduzindo a capacidade humana e recursos materiais para substituição da bateria, impactando também na taxa de erro e interferência entre os fluxos de rede que vão diminuir a medida que o número de dispositivos diminui.

Topologia da Rede



Uma topologia de rede é um mapeamento de uma rede de comunicações que define como os dispositivos de rede são organizados e como a dados fluem pela rede. Em nosso projeto utilizamos o modelo de topologia em árvore / estrela a estrela

Figura 2 - Consumo de energia e alcance de cobertura para diferentes tecnologias sem fio.

Fonte: Tonghao Chen, 2019

Arquitetura de Sistemas Embarcados

Um sistema embarcado pode ser definido como um dispositivo com um único propósito de aplicação. São sistemas microprocessados e normalmente fazem parte de outro sistema maior, agindo como controlador de eventos com sensores, atuadores e interfaces para o acesso de seus usuários. Exemplos de equipamentos onde se utilizam sistemas embarcados: veículos, robótica, casas inteligentes, sistemas de aeronaves, equipamentos agrícolas entre outros. No sistema proposto nesta tese são utilizados dois chips microcontroladores, um localizado na placa do dispositivo final (ATMega2560) e outro localizado na placa do gateway (ATMega328P).

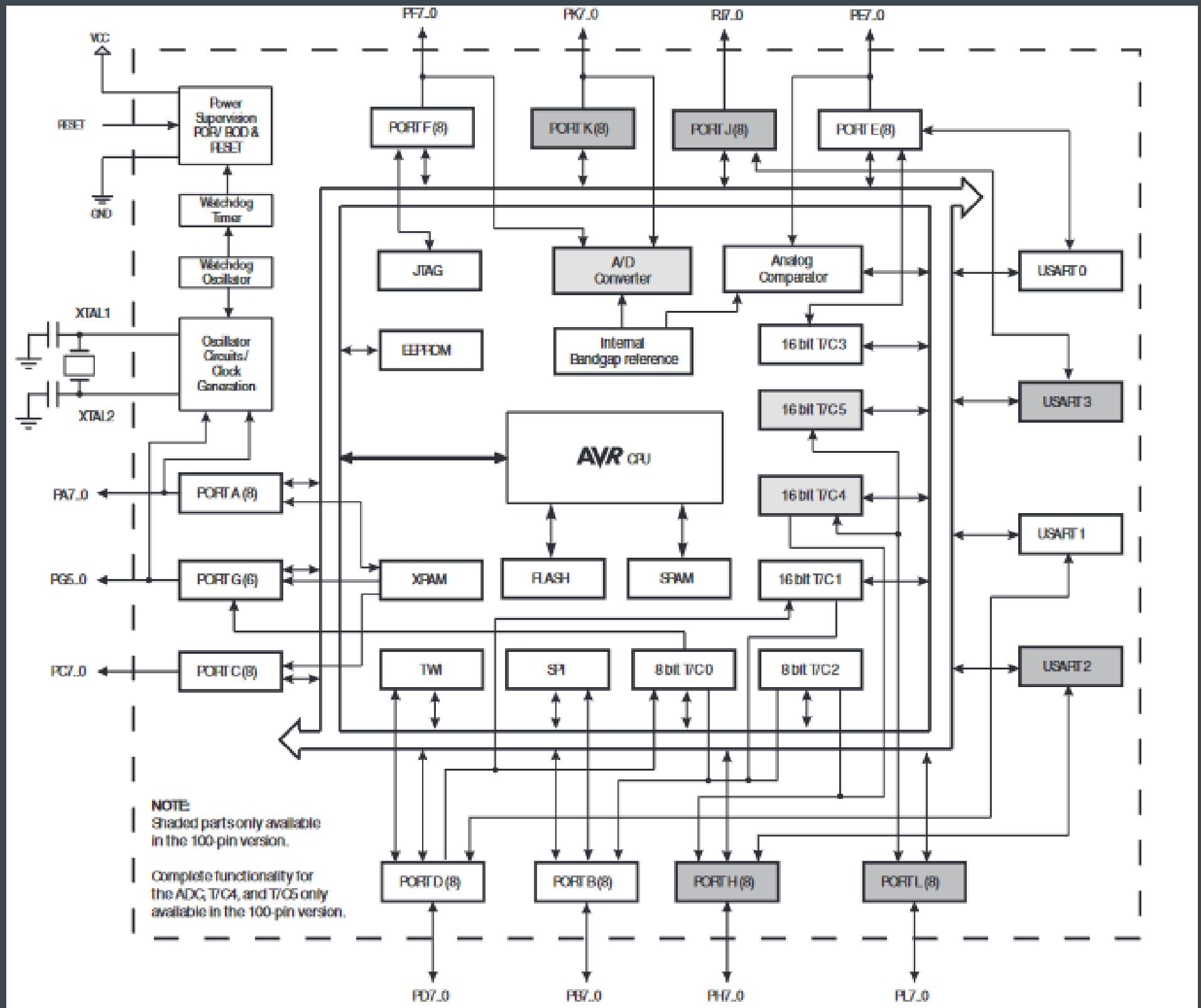


Figura 2 - Diagrama de blocos do chip microcontrolador ATMega2560.

Fonte: Datasheet Atmel ATmega640/V-1280/V-1281/V-2560/V-2561/V.



Figura 1 – Figura contendo o conjunto de imagens MobileNet

Fonte: ImageNet Large Scale Visual Recognition Challenge, 2015

Transfer Learning - ImageNet

Transfer Learning (aprendizado por transferência) é um método de aprendizado de máquina em que um modelo desenvolvido para uma tarefa é reutilizado como ponto de partida para um modelo em uma segunda tarefa. É muito popular em Deep Learning, onde modelos pré-treinados são usados como o ponto de partida em tarefas de visão computacional e processamento de linguagem natural, devido aos vastos recursos de computação e tempo necessários para desenvolver modelos de rede neural e dos grandes saltos em habilidades que eles fornecem acelerando o processo de treinamento melhorando o desempenho do nosso modelo

A base de imagens ImageNet possui mais de dez milhões de URLs com imagens rotuladas. A base de imagens foi utilizada para o treinamento dos modelos que serão descritos posteriormente.

Deep Learning - Redes Neurais Convolucionais (CNN)

CNNs são um tipo específico de rede neural artificial bastante eficaz para classificação de imagens pois são capazes de levar em consideração a coerência espacial da imagem, ou seja, que pixels próximos uns dos outros estão frequentemente relacionados.

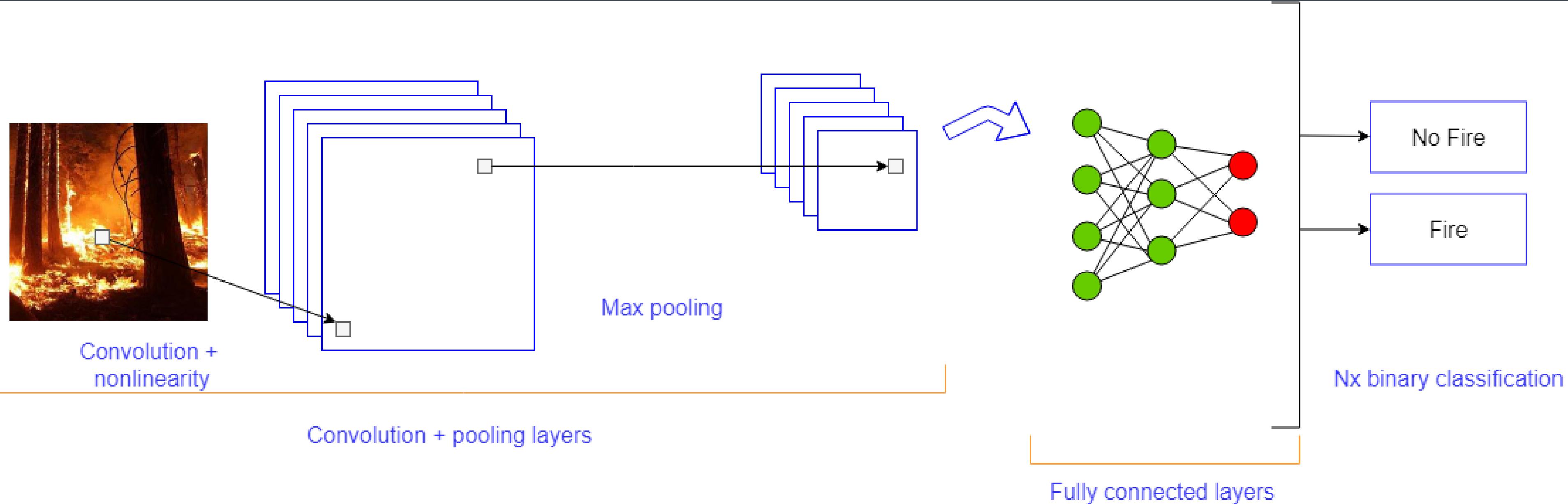


Figura 2 – Figura que ilustra o funcionamento da Rede Neural Convolucional

Fonte: Autor, 2021

Modelos utilizados - InceptionV3

Proposto por Szegedy et al. (2015), é uma arquitetura de CNN que busca a resolução de diversos problemas de reconhecimento de imagens em grande escala e também podem ser utilizados com técnicas de transfer learning (descrito anteriormente). O seu diferencial é a presença de módulos extractores de características convolucionais. Estes módulos tem como funcionalidade aprender com menos parâmetros que contém uma maior gama de informação.

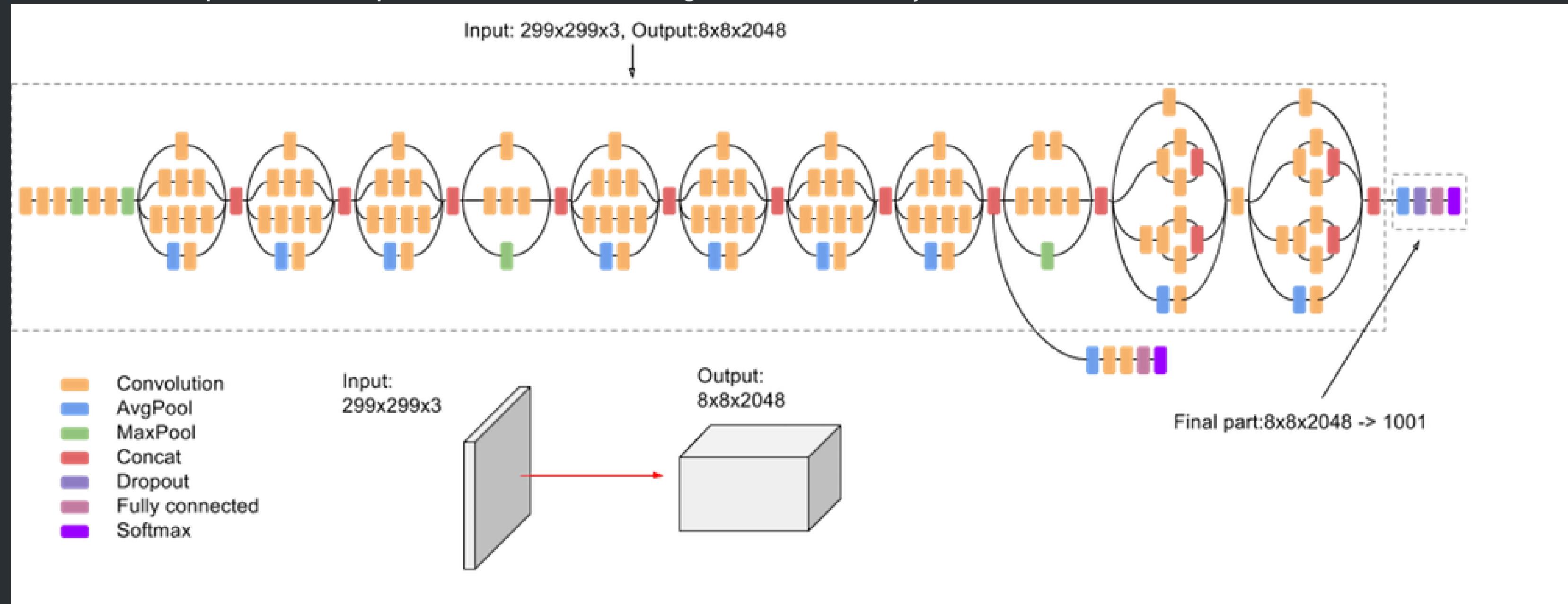


Figura 3 – Figura que ilustra o funcionamento do algoritmo InceptionV3

Fonte: Google Cloud, 2021

Modelos utilizados - Xception

Proposto por Chollet et al.(2016), é uma arquitetura de CNN similar ao Inception descrito anteriormente e, também podem ser utilizados com técnicas de transfer learning (descrito anteriormente). O diferencial do modelo é que os módulos de iniciação foram substituídos por convoluções separáveis em profundidade. A Xception possui a mesma quantidade de parâmetros que o InceptionV3 com o total de 36 camadas convolucionais. Assim, tendo um uso mais eficiente dos parâmetros.

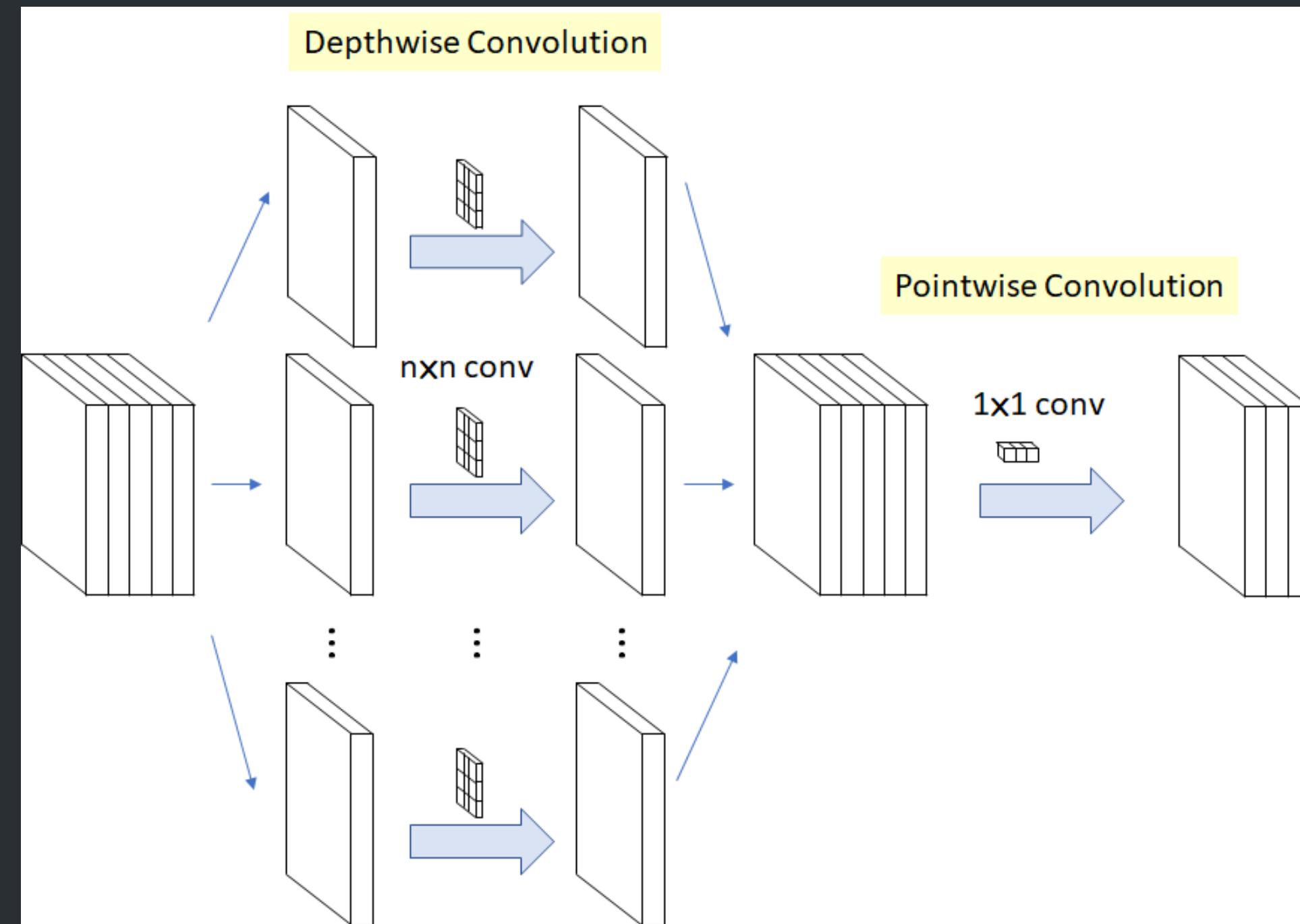


Figura 4 – Figura que ilustra o funcionamento do algoritmo Xception

Modelos utilizados - VGG16

Proposto por K. Simonyan et al. (2014), é uma arquitetura de CNN que apresenta uma menor quantidade de camadas e uma distribuição de pesos que é capaz de extrair diversas características. Também são utilizadas com técnicas de transfer learning (descrito anteriormente).

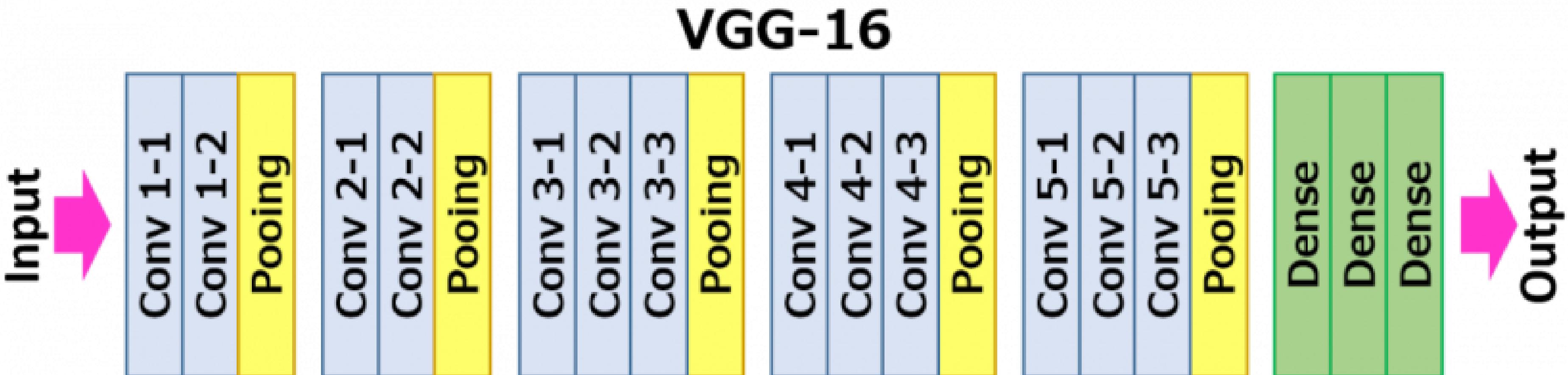


Figura 5 – Figura que ilustra o funcionamento do algoritmo VGG16.

Fonte: NeuroHive, 2018

Modelos utilizados - MobileNet

Proposto por Howard et al. (2017), é uma arquitetura de CNN que foram criados para a execução de tarefas de visão computacional em dispositivos móveis e sistemas embarcados. Também podem ser utilizados com técnicas de transfer learning (descrito anteriormente). Eles são baseados em operações de convolução separáveis em profundidade, que diminui o ônus das operações nas primeiras camadas.

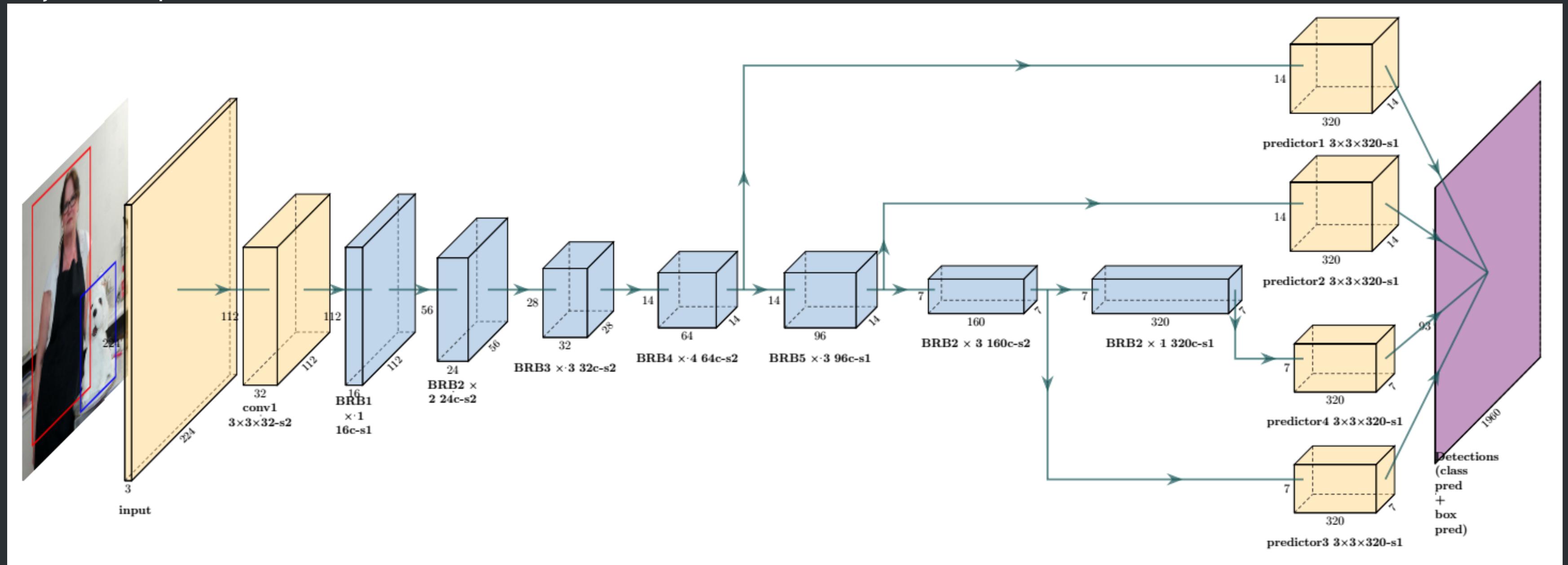


Figura 6 – Figura que ilustra o funcionamento do algoritmo MobileNet.

Fonte: MobileNet-Tiny, 2021



Estratégias de validação

Matriz de Confusão

A matriz de confusão é uma métrica de avaliação que ajuda a compreender o desempenho dos modelos onde temos os valores reais e os valores preditos pelo nosso modelo. Como podemos observar na figura abaixo:

		Valor predito	
		Com fogo	Sem fogo
Real	Com fogo	Verdadeiro Positivo (TP)	Falso Negativo (FN)
	Sem fogo	Falso Positivo (FP)	Verdadeiro Negativo (TN)

Figura 7 – Diagrama que ilustra a matriz de confusão.

Fonte: Autor, 2021

Acurácia, Precisão, Recall e F1-Score

Acurácia (AC)

indica uma performance geral do modelo. Dentre todas as classificações, quantas o modelo classificou corretamente.

$$AC = \frac{(TP + TN)}{(TN + TP + FN + FP)}$$

Figura 8 –Figura que indica o cálculo da Acurácia.

Fonte: Autor, 2021

Recall (RC)

Proporção dos incêndios encontrados no algoritmo.

$$RC = \frac{TP}{(TP + FN)}$$

Figura 9 –Figura que indica o cálculo do Recall.

Fonte: Autor, 2021

Precisão (PR)

Número de observações classificadas como fogo ou sem fogo que são relevantes.

$$PR = \frac{TP}{(TP + FP)}$$

Figura 10 –Figura que indica o cálculo da precisão.

Fonte: Autor, 2021

F1-Score

Média harmônica de recall e precisão que indica a qualidade geral dos modelos.

$$F_1 = \frac{(2 \times PR \times RC)}{(PR + RC)}$$

Figura 11 –Figura que indica o cálculo do F1-Score.

Fonte: Autor, 2021

Curva ROC e AUC

Curva ROC/AUC

A curva ROC é uma das formas de analisar classificadores em problemas binários, o algoritmo é considerado perfeito quando ele possui maior protuberância em direção ao canto superior esquerdo que é mostrado na figura ao lado.

A curva ROC varia entre 0 e 1.

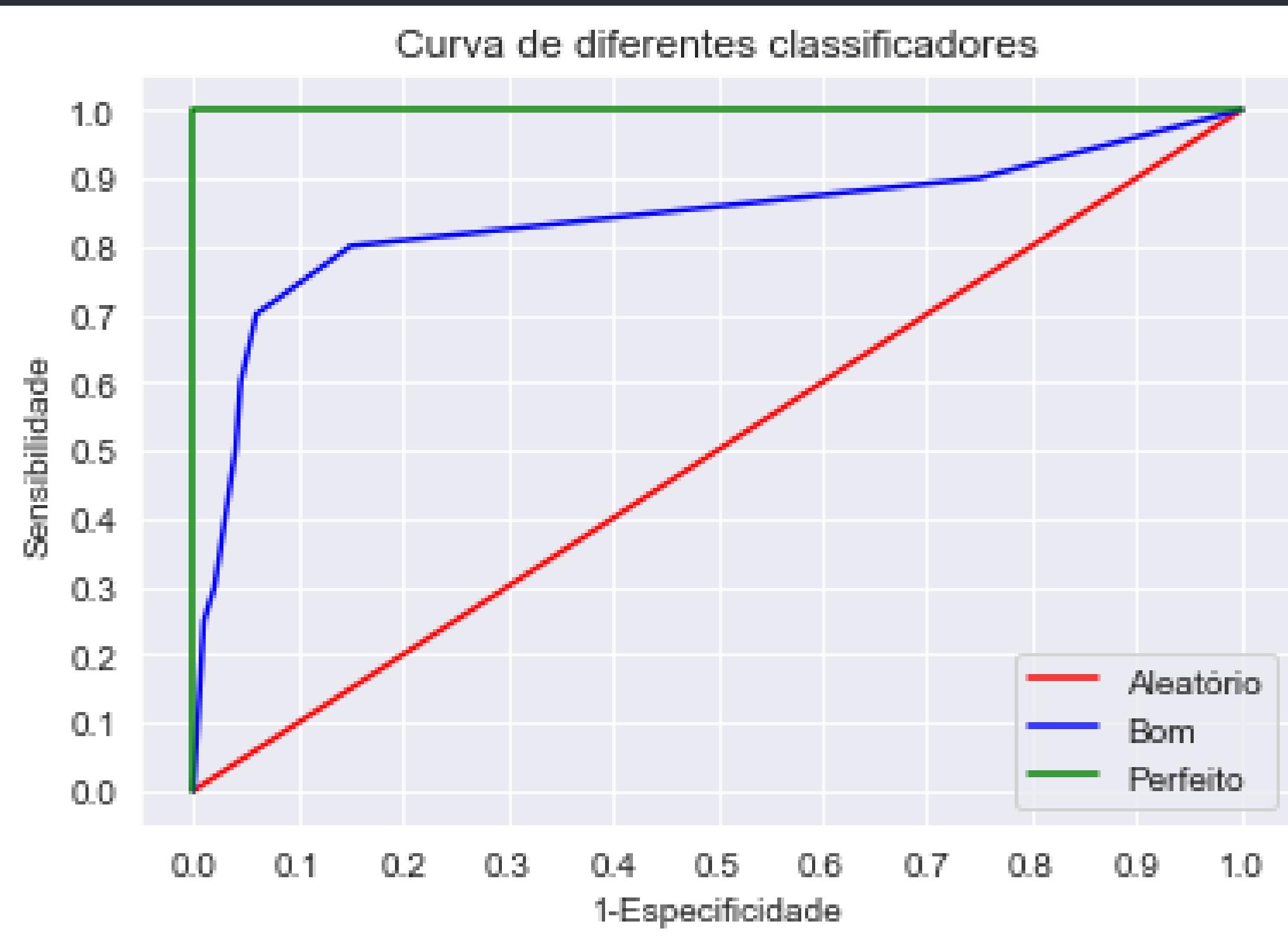


Figura 11 – Diagrama que ilustra a Curva de Receiver Operating Characteristic (ROC).



Metodologia

Metodologia

Este tópico tem como objetivo apresentar os aspectos metodológicos deste trabalho. A metodologia aplicada nesta tese segue uma natureza prática, isto é, aplicada, a partir de uma abordagem quantitativa e cujo objetivo é tratado como sendo explicativo e exploratório. No diagrama abaixo é possível observar as etapas que foram seguidas para obtenção do produto final deste estudo.

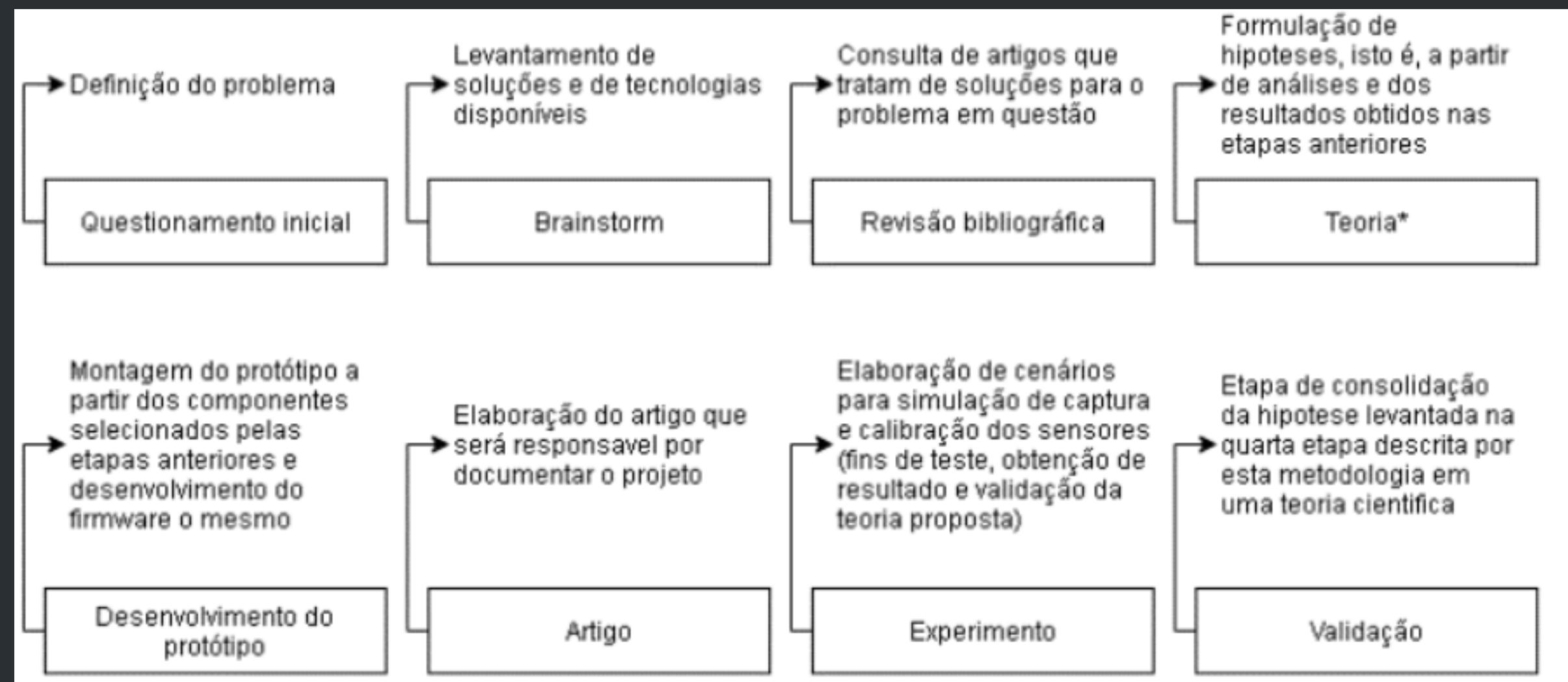
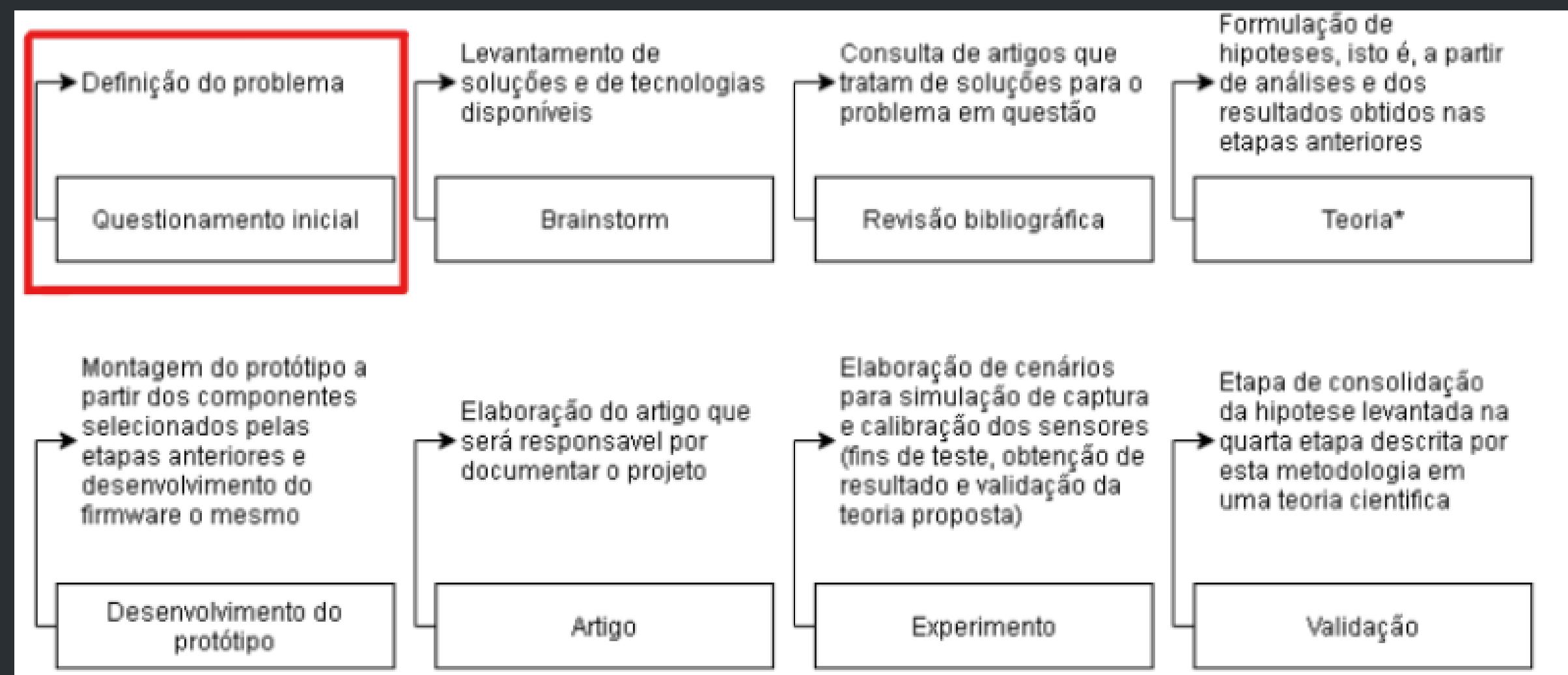


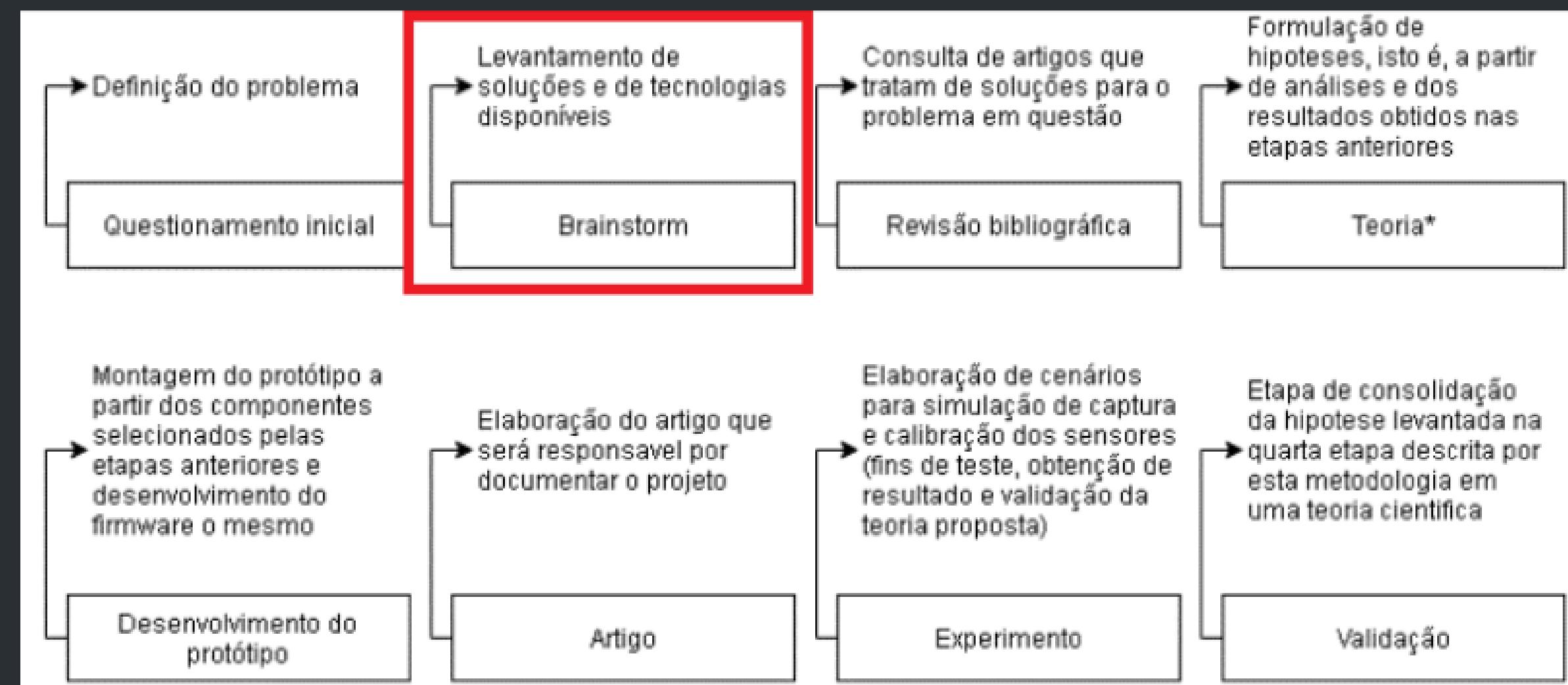
Figura 12 – Diagrama de representação dos procedimentos metodológicos.

Fonte – Autor, 2021

A primeira etapa, nomeada como questionamento inicial conforme observado na figura acima, abrange o processo de definição do problema. Nesta fase, os envolvidos discutiram acerca de múltiplas adversidades que estavam em destaque naquele momento e que poderiam ser solucionadas, isto é, a partir de conhecimentos prévios dos integrantes.

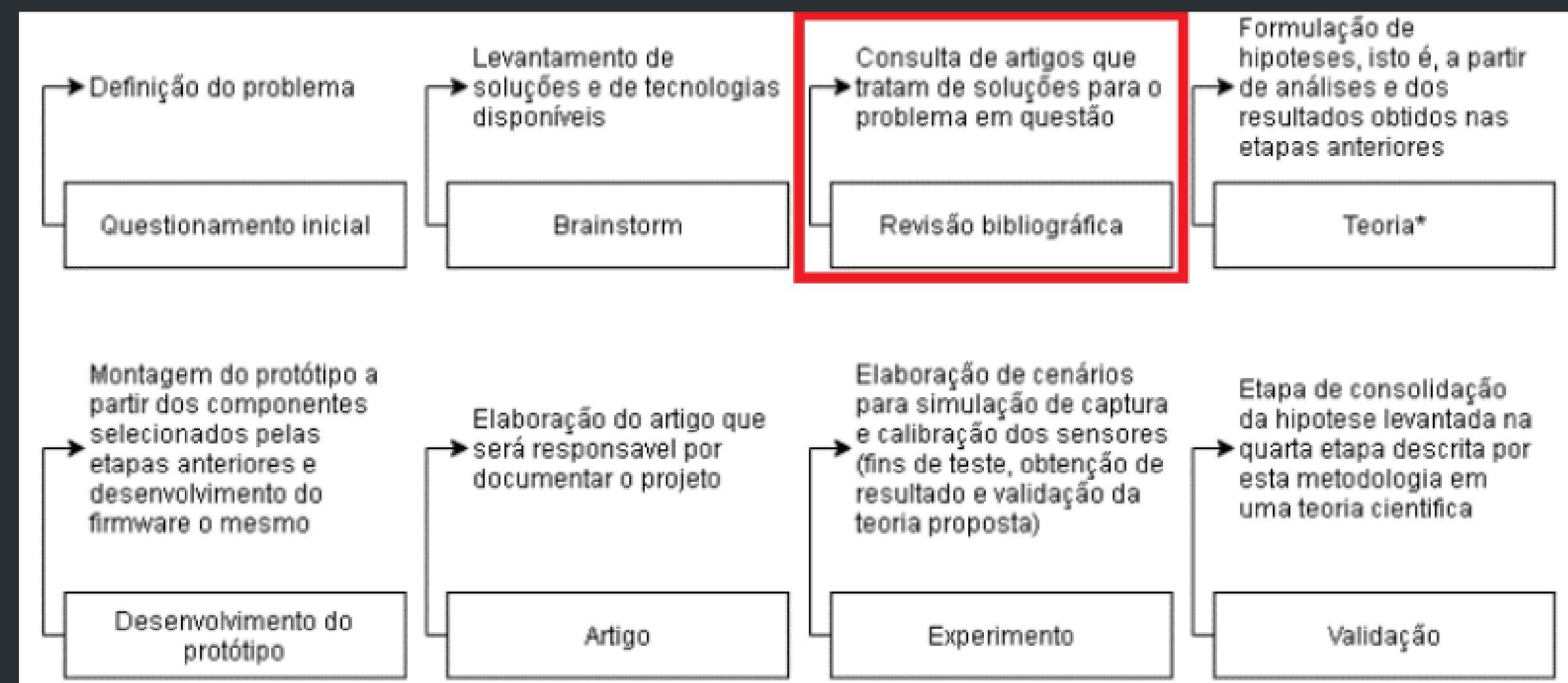


Após este processo, caracterizado pela definição da ideia que norteará as fases subsequentes, foi realizado um brainstorm que teve como objetivo obter soluções para o dado problema inicial e, além disso, elencar as tecnologias e protocolos que seriam úteis para a implementação do projeto de maneira eficiente, inovadora e de forma a considerar o máximo de ramificações, tanto em termos de soluções, quanto em termos de tecnologias que poderiam ser incorporadas a estas. Evidentemente que algumas das ramificações propostas pelo brainstorm foram alteradas ou eliminadas nas fases subsequentes.

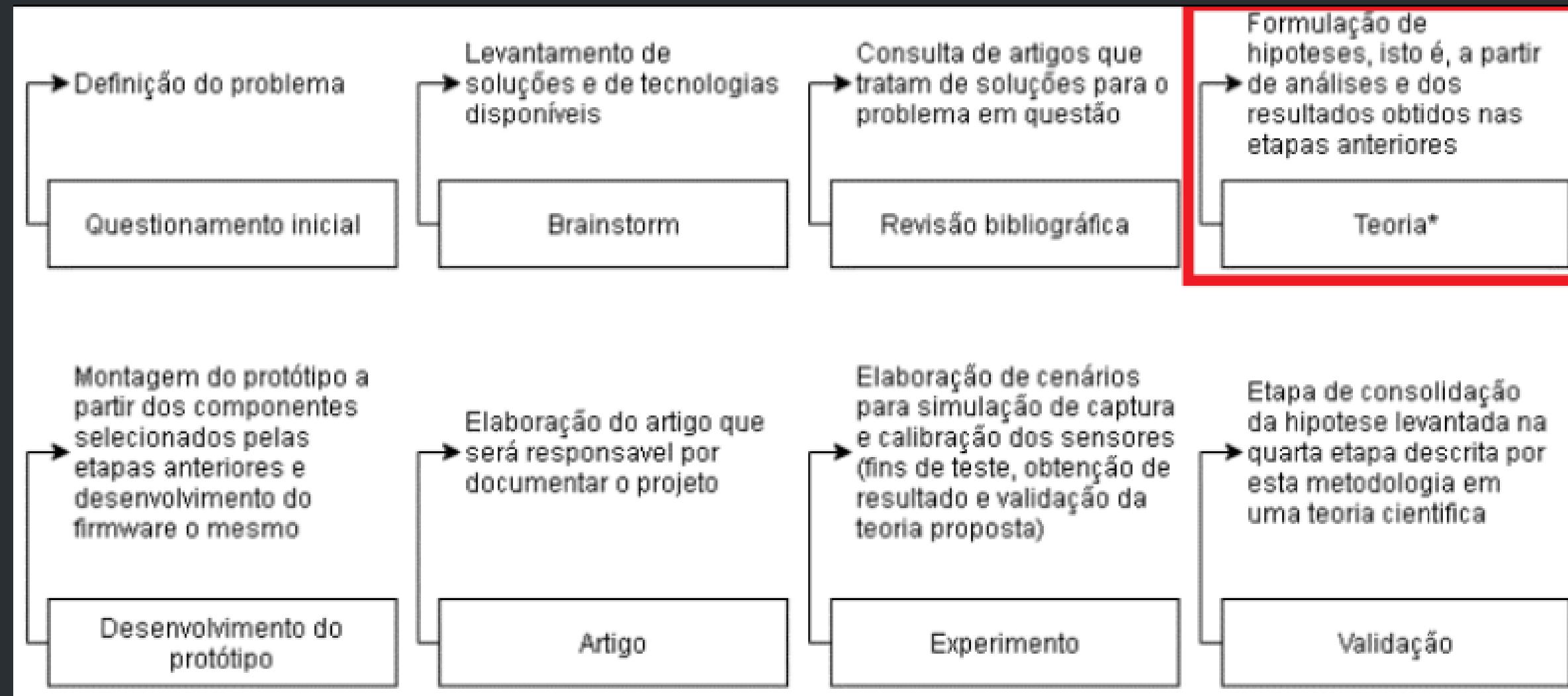


Após a etapa de brainstorming foi realizada a etapa de consulta de artigos científicos que se relacionam com o problema abordado neste projeto. Este processo visou a obtenção de dados relacionados a maneira como são ou foram conduzidas pesquisas nesta área, isto é, afim de formular uma espécie de base para o projeto.

É importante ressaltar que esta etapa também foi importante para a obtenção de dados relacionados ao progresso dos estudos nesta área, isto é, afim de otimizar o processo de desenvolvimento do projeto, e foi importante para a averiguação dos métodos de validação já previamente utilizados por outros pesquisadores.



A formulação de hipóteses para elaboração de uma teoria científica é uma etapa importante que envolve a compreensão do que será tratado exatamente no trabalho. Esta fase foi realizada através do levantamento de novas indagações que estão relacionadas ao questionamento inicial.

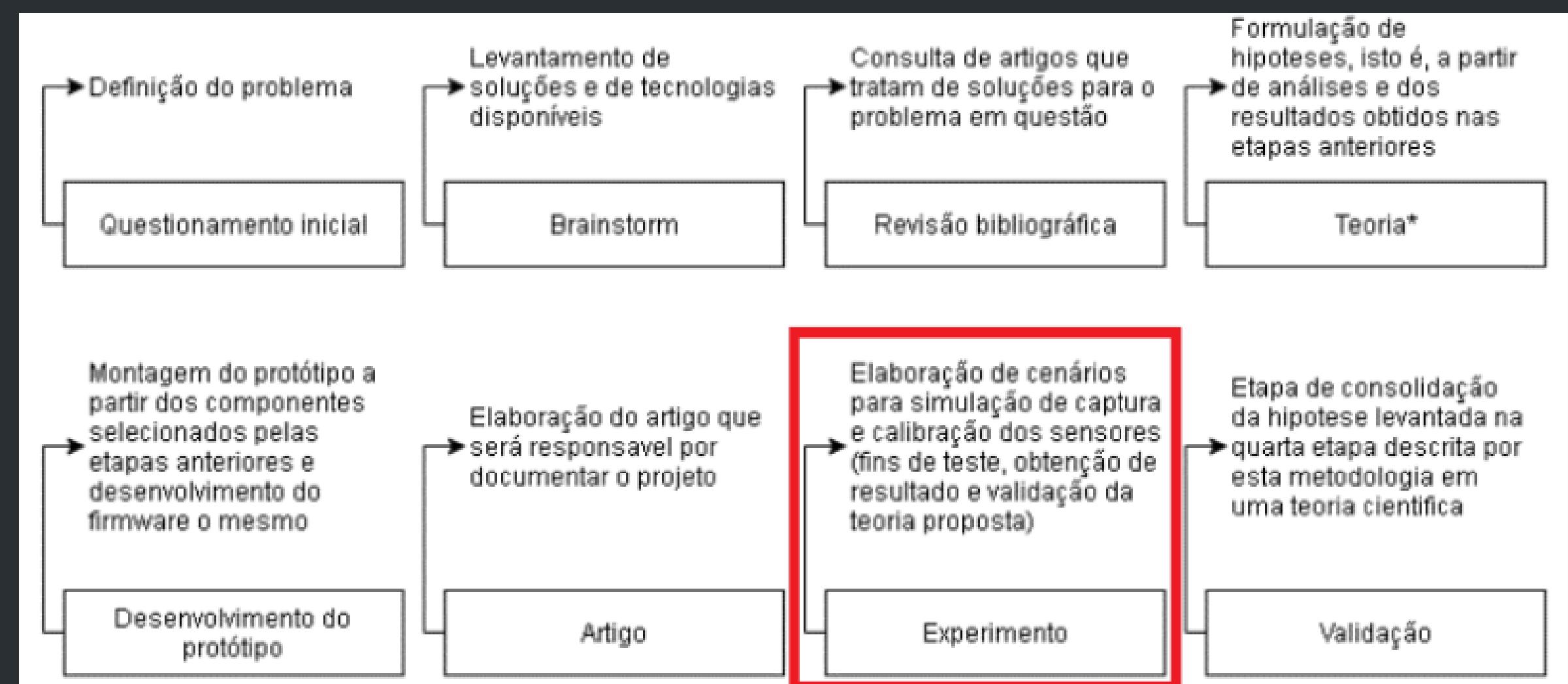


Metodologia para Formulação de Hipóteses

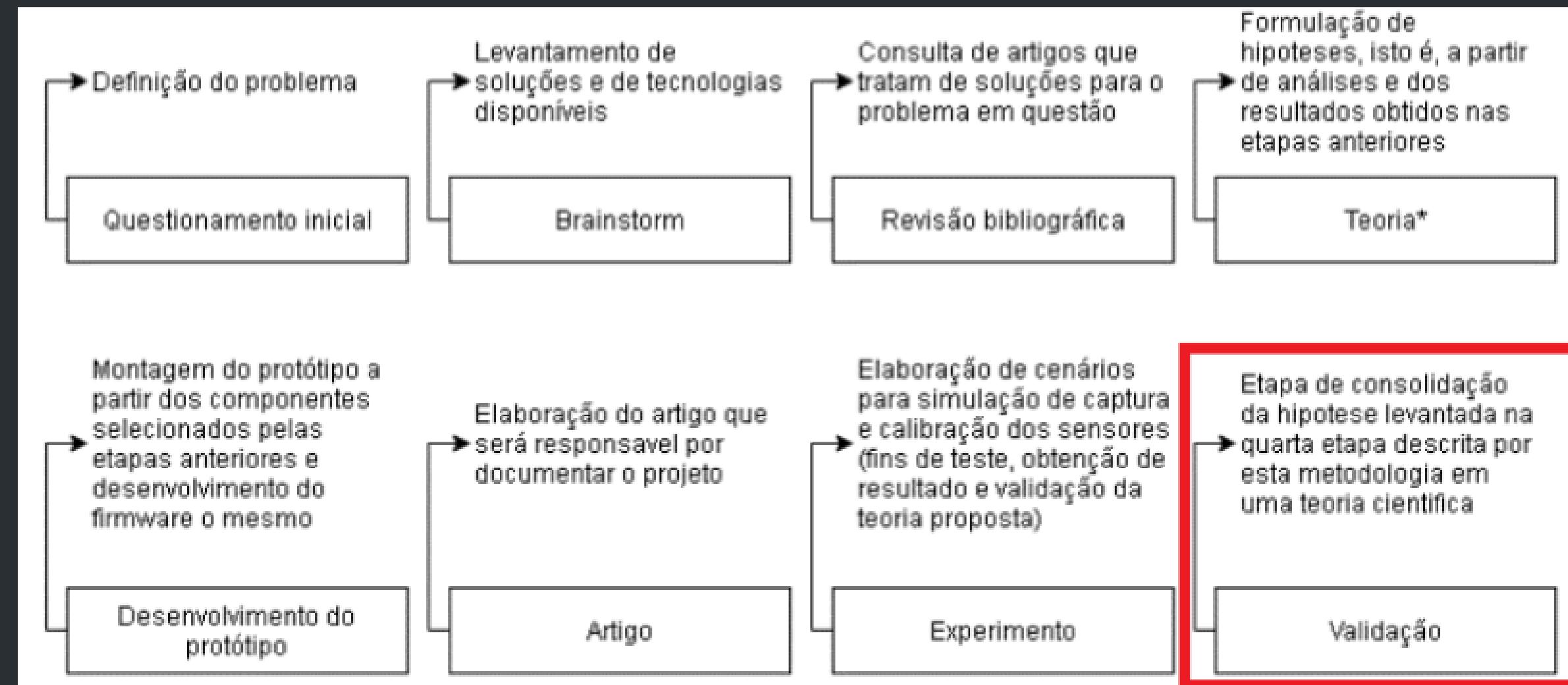
A metodologia para formulação de hipóteses utilizada neste projeto é dada pelas seguintes etapas:

- Definição do tópico baseado no questionamento inicial
- Realização de novas indagações
- Elaboração de hipóteses
- Explicação e previsões
- Análise de possíveis riscos

No processo de elaboração e realização do experimento são tratados aspectos como uso de cenários para testes e calibração dos sensores. A metodologia adotada para esta fase é a de testar e calibrar individualmente os sensores, isto é, submetendo estes a variações daquilo que está sendo medido (variáveis como, por exemplo, temperatura, gás, localização). Ainda nesta etapa, posteriormente a este teste e calibração e, evidentemente, após treinamento da rede neural convolucional, o sistema é submetido a testes em campo afim de progredir para a próxima etapa (validação).



A fase de validação deste projeto tem como objetivo apresentar os resultados obtidos nas etapas anteriores, isto é, no processo de elaboração e realização do experimento. A validação em si ocorre a partir da análise das métricas de avaliação das técnicas abordadas neste projeto.





Desenvolvimento

Base de imagens utilizadas – Conjunto 1

Os dados foram disponibilizados por Khan et al. (2020) para o treinamento dos modelos para distinguir entre as imagens que contêm fogo (imagens de fogo) e imagens regulares (imagens que não são de fogo), então todo o problema é de classificação binária. Na figura abaixo é mostrado alguns exemplos de imagens no conjunto de dados 1.



Figura 13 – Figura que ilustra o conjunto de imagens 1

Fonte: Autor, 2021

Base de imagens utilizadas – Treinamento, Validação e Teste 1

O conjunto de dados 1 foi dividido em 60:20:20, então ficamos com estas quantidades de imagens: 1216 para treinamento, 304 para validação, 380 para teste. Na figura a seguir podemos observar que ambos os datasets estão平衡ados.

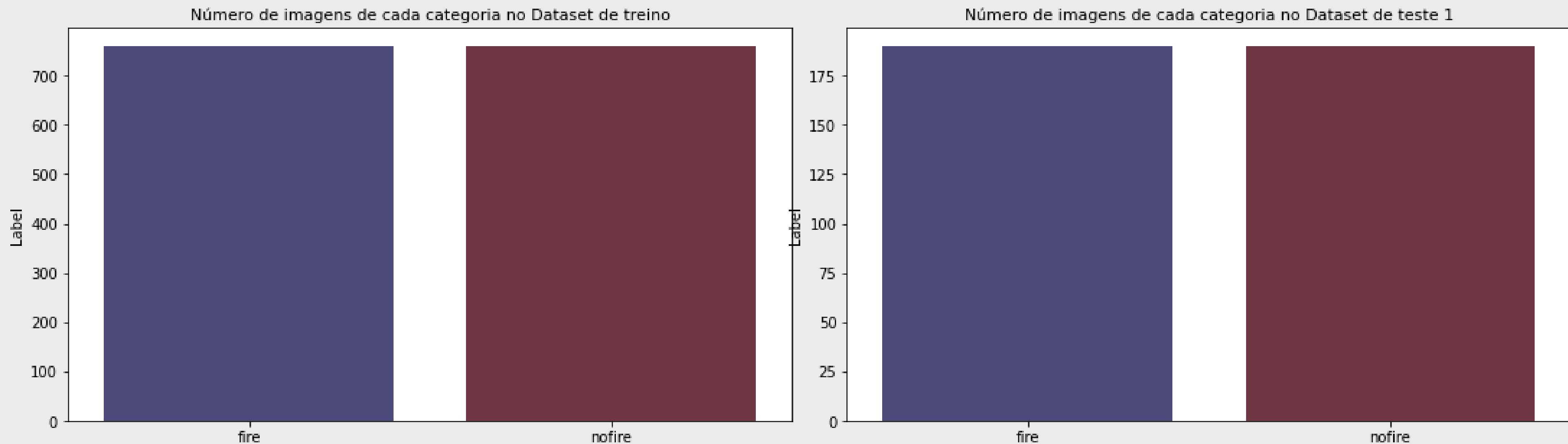


Figura 14 e 15 – Gráficos que indicam a quantidade de imagens em cada conjunto de dados.

Fonte: Autor, 2021

Base de imagens utilizadas – Conjunto 2

Os dados foram disponibilizados pela plataforma Kaggle e o utilizamos para o teste dos modelos para distinguir entre as imagens que contêm fogo (imagens de fogo) e imagens regulares (imagens que não são de fogo), então todo o problema é de classificação binária. Na figura abaixo é mostrado alguns exemplos de imagens no conjunto de dados 2.



Figura 16 – Figura que ilustra o conjunto de imagens 2

Fonte: Autor, 2021

Base de imagens utilizadas - Teste 2

O conjunto de dados 2 contém o total de 998 imagens ao total, sendo 755 imagens de fogo 244 imagens da natureza.

Na figura a seguir podemos observar que ambos os datasets estão desbalanceados e por este motivo decidimos utiliza-lo apenas para o teste dos modelos.

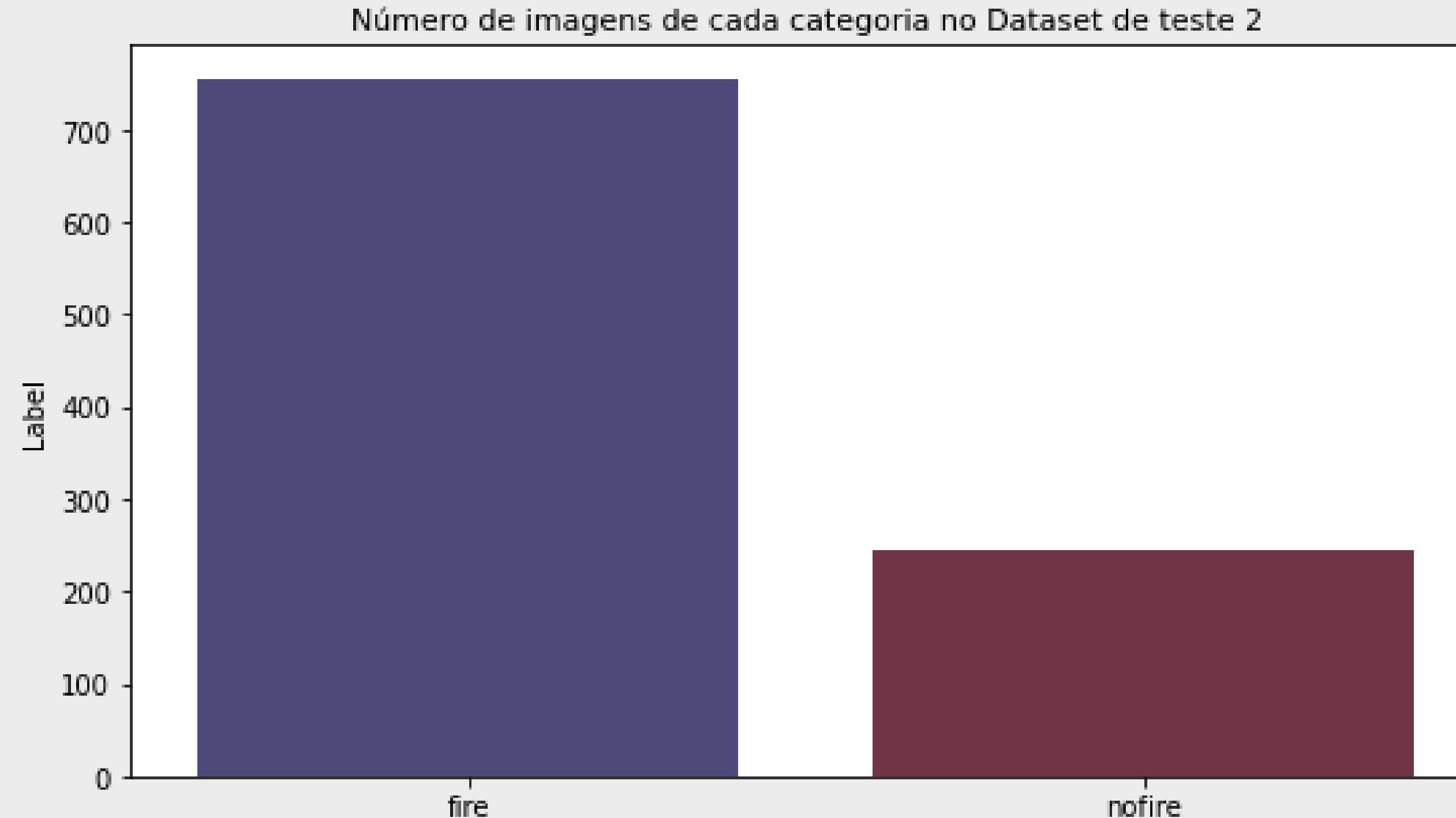


Figura 17 – Gráfico que indicam a quantidade de imagens no conjunto de teste 2.

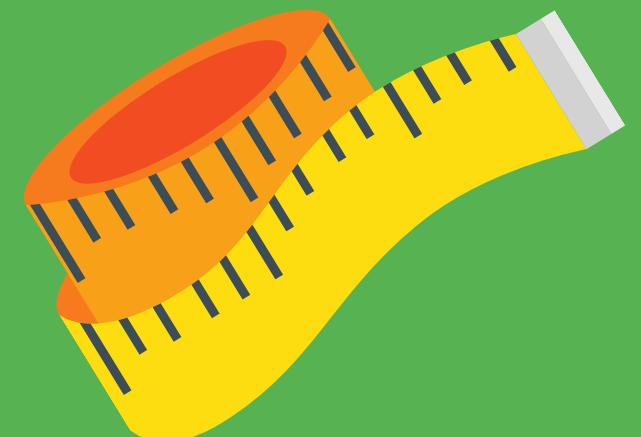
Fonte: Autor, 2021



Resultados

Cenários

Ao longo do desenvolvimento deste projeto foram criados diversos cenários afim de validar o funcionamento adequado dos componentes do sistema. De modo geral é possível dizer que a execução dos cenários para testes estão dividida em duas partes. Na primeira parte, cada todos os componentes (sensores, transceptores e protocolos) foram testados individualmente. Na segunda parte da execução dos cenários de testes, foram testados todos os componentes simultaneamente, isto é, todos atuando em um mesmo cenário.



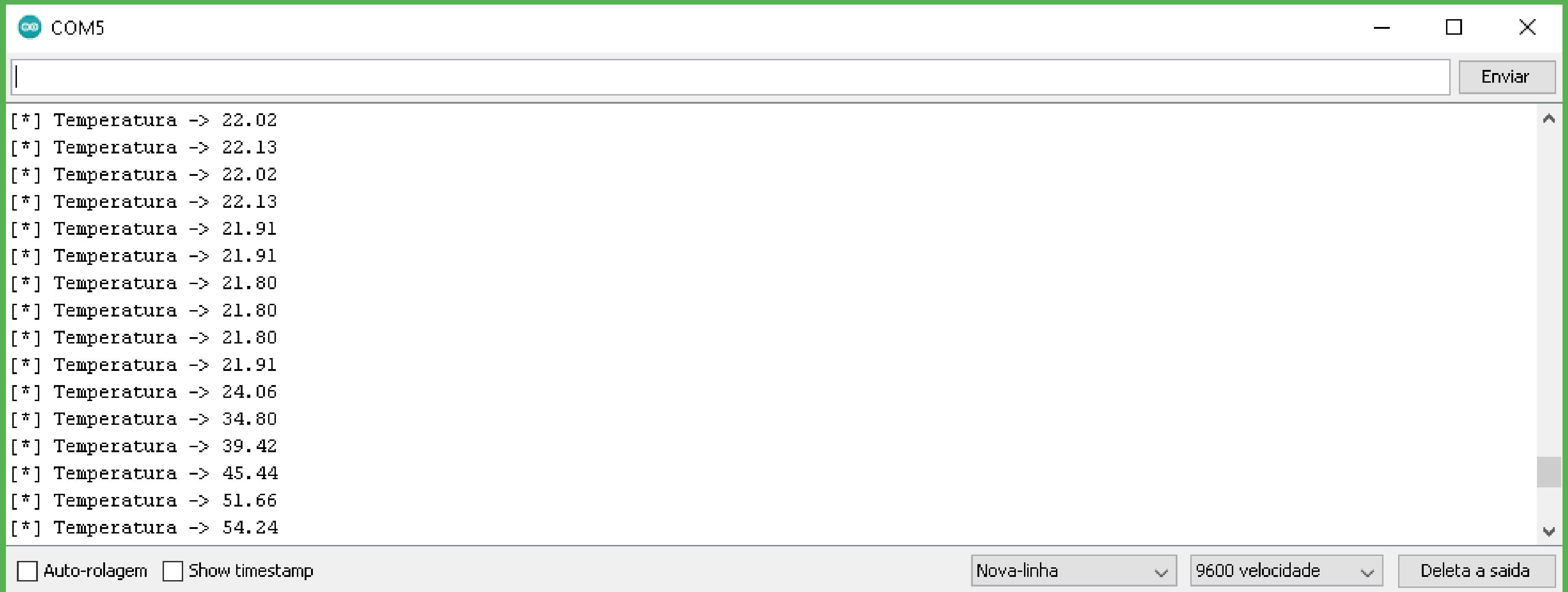
Cenários



O primeiro cenário foi criado para avaliar o funcionamento individual do sensor de temperatura. Para realização dos testes foi utilizado um ferro de solda. O ferro foi aproximado do sensor de temperatura lentamente até encostar nele. Os resultados deste teste confirmaram o bom funcionamento do sensor e a corretude do algoritmo responsável por calcular a temperatura a partir da voltagem de saída do sensor. Na figura abaixo é mostrado o a execução do teste do sensor de temperatura, a temperatura no dia do teste, a temperatura do sensor e a temperatura do sensor utilizando o ferro de solda.



Cenários



The screenshot shows a terminal window titled "COM5". The window contains a list of temperature readings, each preceded by a "[" symbol and followed by an arrow and the value. The values fluctuate between 21.80 and 54.24. The window has standard operating system controls (minimize, maximize, close) at the top right. At the bottom, there are checkboxes for "Auto-rolagem" and "Show timestamp", and dropdown menus for "Nova-linha" (New Line), "9600 velocidade" (9600 speed), and "Delete a saída" (Delete output).

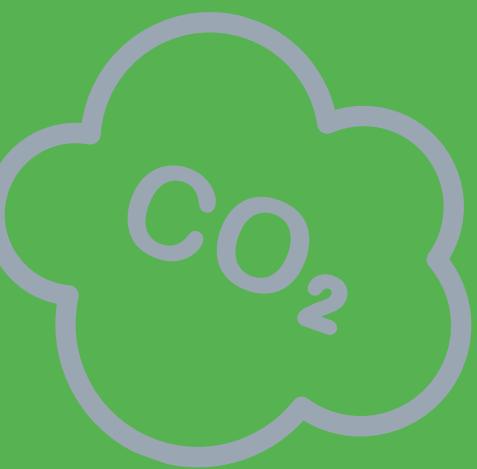
```
[*] Temperatura -> 22.02
[*] Temperatura -> 22.13
[*] Temperatura -> 22.02
[*] Temperatura -> 22.13
[*] Temperatura -> 21.91
[*] Temperatura -> 21.91
[*] Temperatura -> 21.80
[*] Temperatura -> 21.80
[*] Temperatura -> 21.80
[*] Temperatura -> 21.91
[*] Temperatura -> 24.06
[*] Temperatura -> 34.80
[*] Temperatura -> 39.42
[*] Temperatura -> 45.44
[*] Temperatura -> 51.66
[*] Temperatura -> 54.24
```

Auto-rolagem Show timestamp

Nova-linha ▾ 9600 velocidade ▾ Delete a saída

Figura 18 - Exibição das temperaturas capturadas durante a execução do primeiro cenário de teste.

Fonte: Autor, 2021.



Cenários

O segundo cenário foi criado para avaliar o funcionamento individual do sensor de concentração de gases. Para determinar o funcionamento adequado do sensor ele foi exposto ao resultado da combustão de um pedaço de papel. A reação oriunda da queima do papel é conhecida por produzir uma determinada quantidade de dióxido de carbono que, como visto na fundamentação teórica desta tese, é um componente primário da fumaça proveniente de incêndios florestais.



Cenários

A screenshot of a terminal window titled "COM5". The window contains a list of messages indicating the capture of CO2 levels. The messages are as follows:

- [*] CO2 -> 410.91
- [*] CO2 -> 408.83
- [*] CO2 -> 407.68
- [*] CO2 -> 406.44
- [*] CO2 -> 405.88
- [*] CO2 -> 405.35
- [*] CO2 -> 405.18
- [*] CO2 -> 405.02
- [*] CO2 -> 404.70
- [*] CO2 -> 404.39
- [*] CO2 -> 404.24
- [*] CO2 -> 404.70
- [*] CO2 -> 571.85
- [*] CO2 -> 772.58
- [*] CO2 -> 1055.02

The terminal window includes standard controls like minimize, maximize, and close buttons at the top right. At the bottom, there are checkboxes for "Auto-rolagem" and "Show timestamp", and dropdown menus for "Nova-linha", "9600 velocidade", and "Delete a saída".

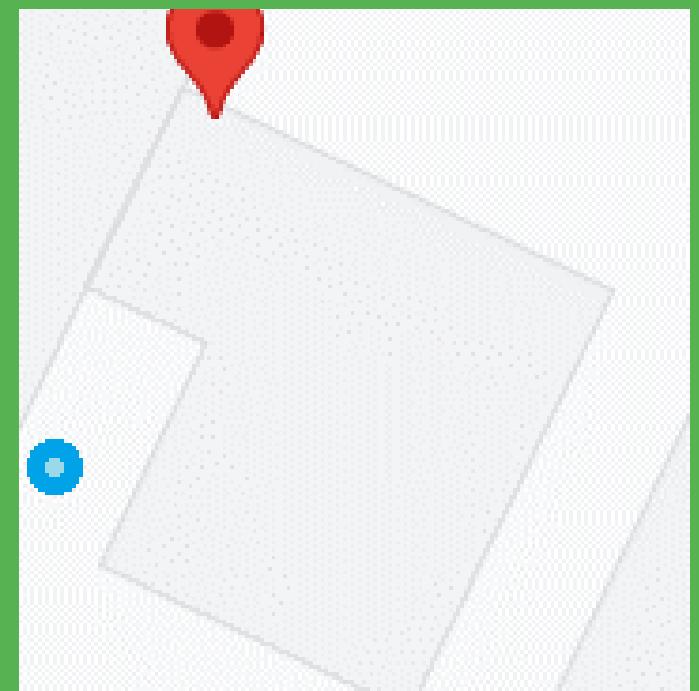
Figura 19 - Exibição da quantidade de CO2 capturadas durante a execução do segundo cenário de teste.

Fonte: Autor, 2021.

Cenários



O terceiro cenário foi criado para avaliar o funcionamento individual do dispositivo de localização GPS. Para realização dos testes foram simplesmente trocada as posições do nó que possuía o dispositivo. Nas figuras abaixo é mostrado o funcionamento do módulo de geolocalização, a imagem referente ao teste e a comparação entre localização real (ponto azul) e localização obtida pelo módulo de geolocalização.



Cenários

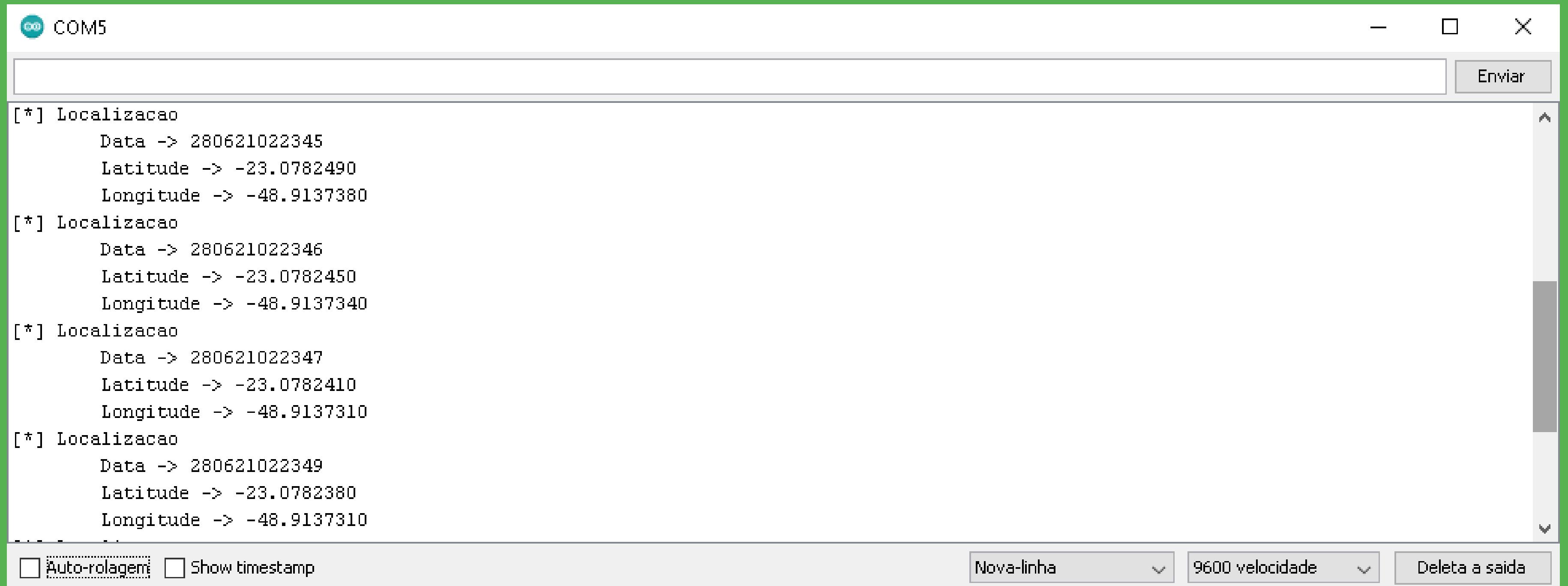
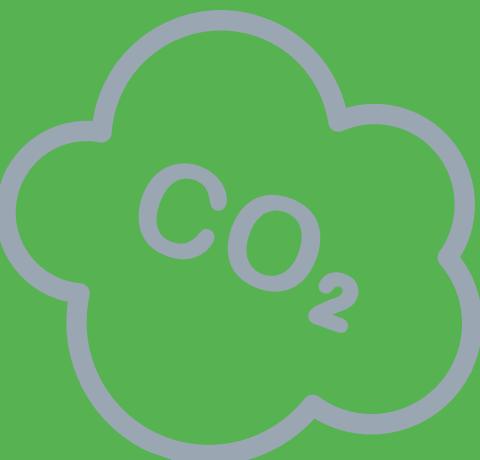


Figura 20 - Exibição da latitude e longitude durante a execução do terceiro cenário de teste.

Fonte: Autor, 2021.



Cenários



O quarto cenário foi criado para avaliar o funcionamento do dispositivo de comunicação LoRa. Foram constatados nesta etapa que a melhor taxa de dados para o dispositivo testado é a taxa de 19200bps, isto é, com um spreading factor de 7 e uma bandwidth de 500kHz.



Resultados - Performance no treinamento e validação

	Modelo	Tempo	Acurácia Treinamento (%)	Acurácia Validação (%)
1	ConvNet	34 minutos	0.968191	0.968684
2	Inception	19 minutos	0.988553	0.977763
3	Xception	28 minutos	0.991020	0.982763
4	VGG16	06 horas e 29 minutos	0.965674	0.954803
5	MobileNet	22 minutos	0.980674	0.993289

Tabela 1 – Tabela indicando os resultados dos modelos utilizados na base de treinamento e validação.

Fonte: Autor, 2021

Resultados - Performance no Teste 1

	Modelo	Acurácia Teste (%)	Recall (%)	F1 (%)	AUC
1	ConvNet	0.939474	0.905263	0.927224	0.928947
2	Inception	0.973684	0.957895	0.973262	0.973684
3	Xception	0.973684	0.947368	0.972973	0.973684
4	VGG16	0.913158	0.921053	0.923483	0.923684
5	MobileNet	0.984211	0.968421	0.983957	0.984211

Tabela 2 – Tabela indicando os resultados dos modelos utilizados na base de teste 1.

Fonte: Autor, 2021

Resultados - Curva ROC Teste 1

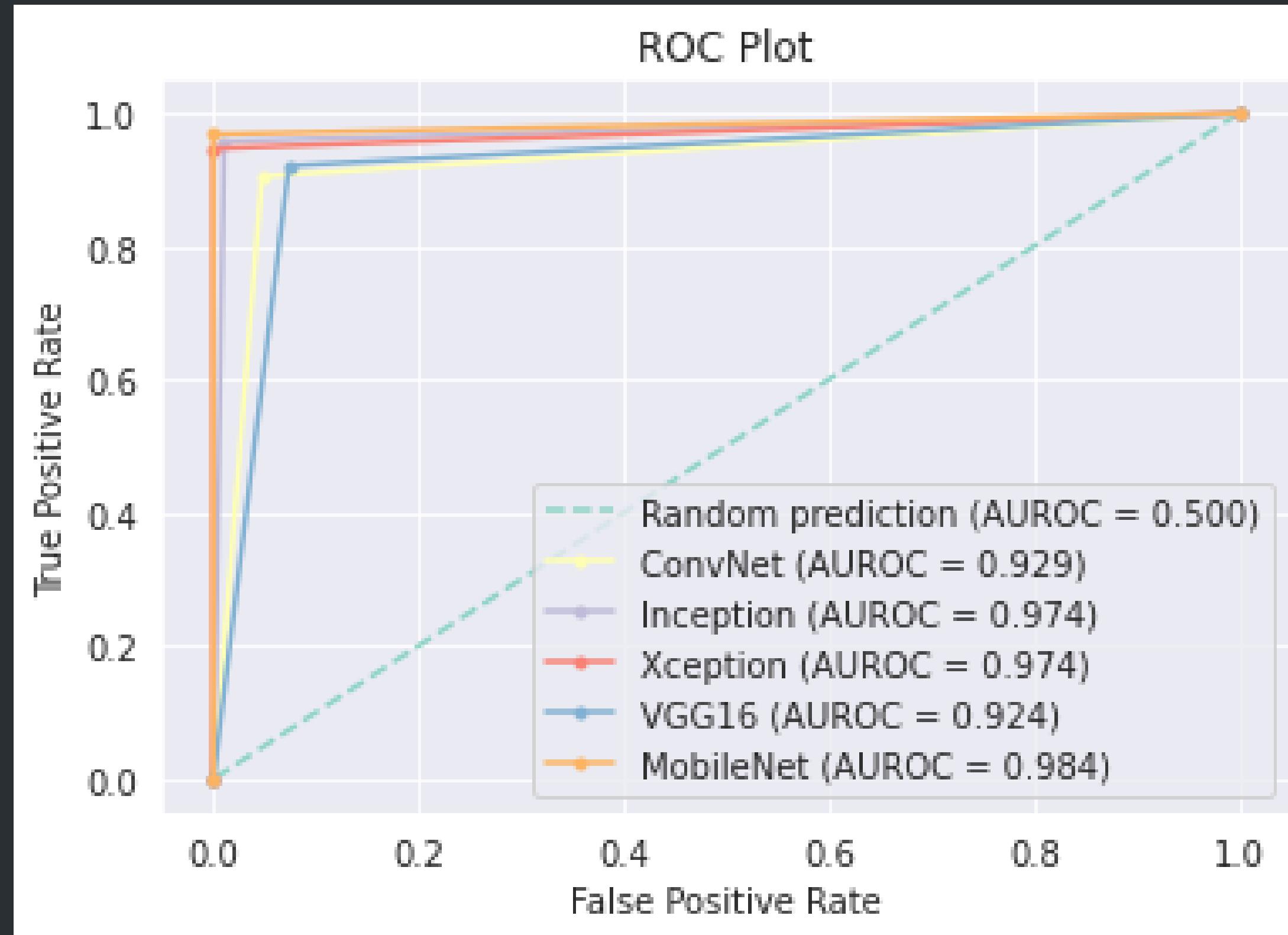


Figura 18 – Figura que ilustra a Curva de Receiver Operating Characteristic (ROC) dos modelos utilizados na base de teste 1.

Resultados - Performance no Teste 2

	Modelo	Acurácia Teste (%)	Recall (%)	F1 (%)	AUC
1	ConvNet	0.937876	0.946502	0.876190	0.938814
2	Inception	0.954910	0.843621	0.901099	0.917175
3	Xception	0.891784	0.588477	0.725888	0.788941
4	VGG16	0.824649	0.481481	0.579208	0.711602
5	MobileNet	0.949900	0.810700	0.887387	0.902701

Tabela 3 – Tabela indicando os resultados dos modelos utilizados na base de teste 2.

Fonte: Autor, 2021

Resultados - Curva ROC Teste 2

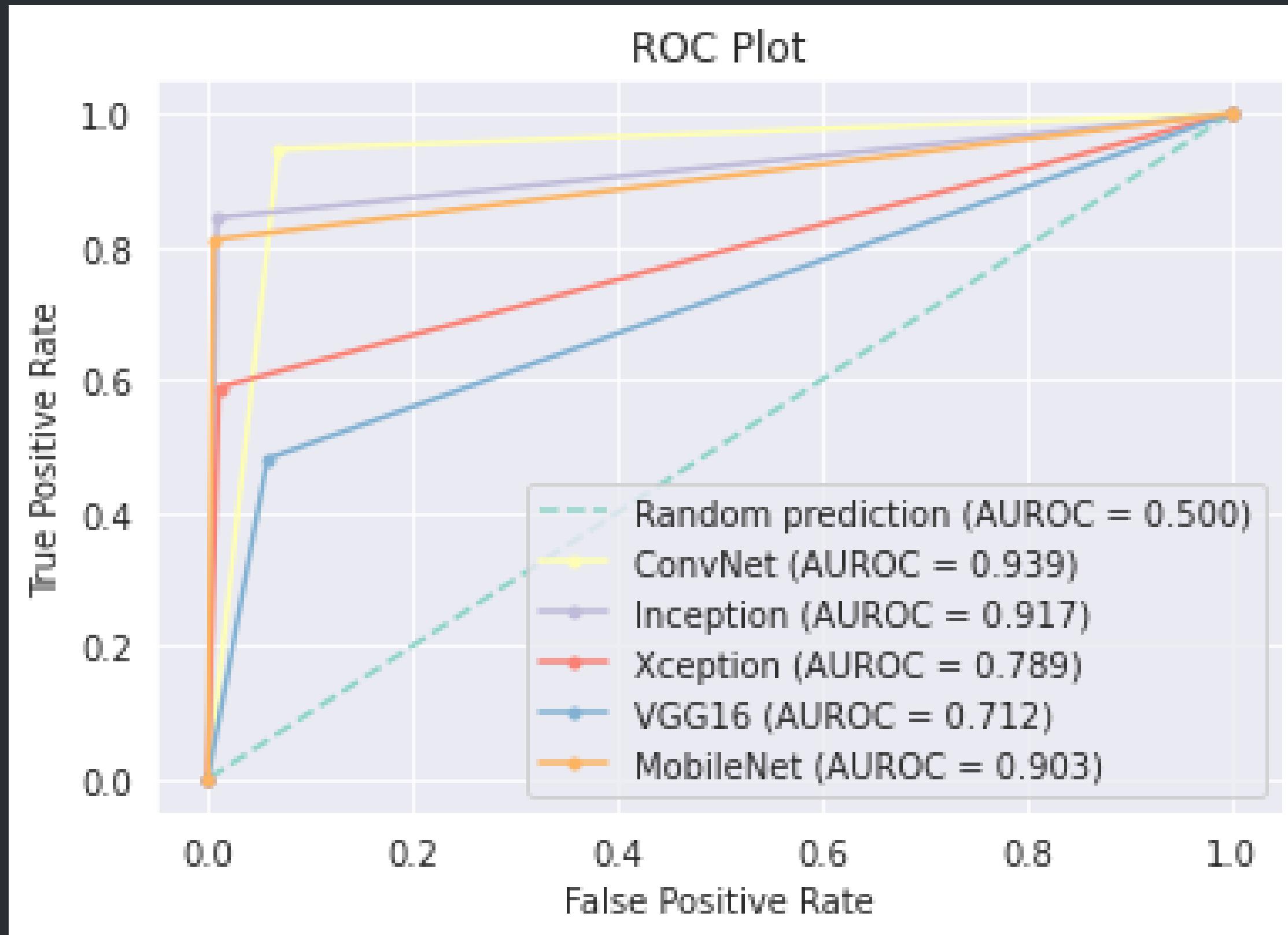


Figura 19 – Figura que ilustra a Curva de Receiver Operating Characteristic (ROC) dos modelos utilizados na base de teste 2.

Resultados - Imagens previstas conjunto de teste 1

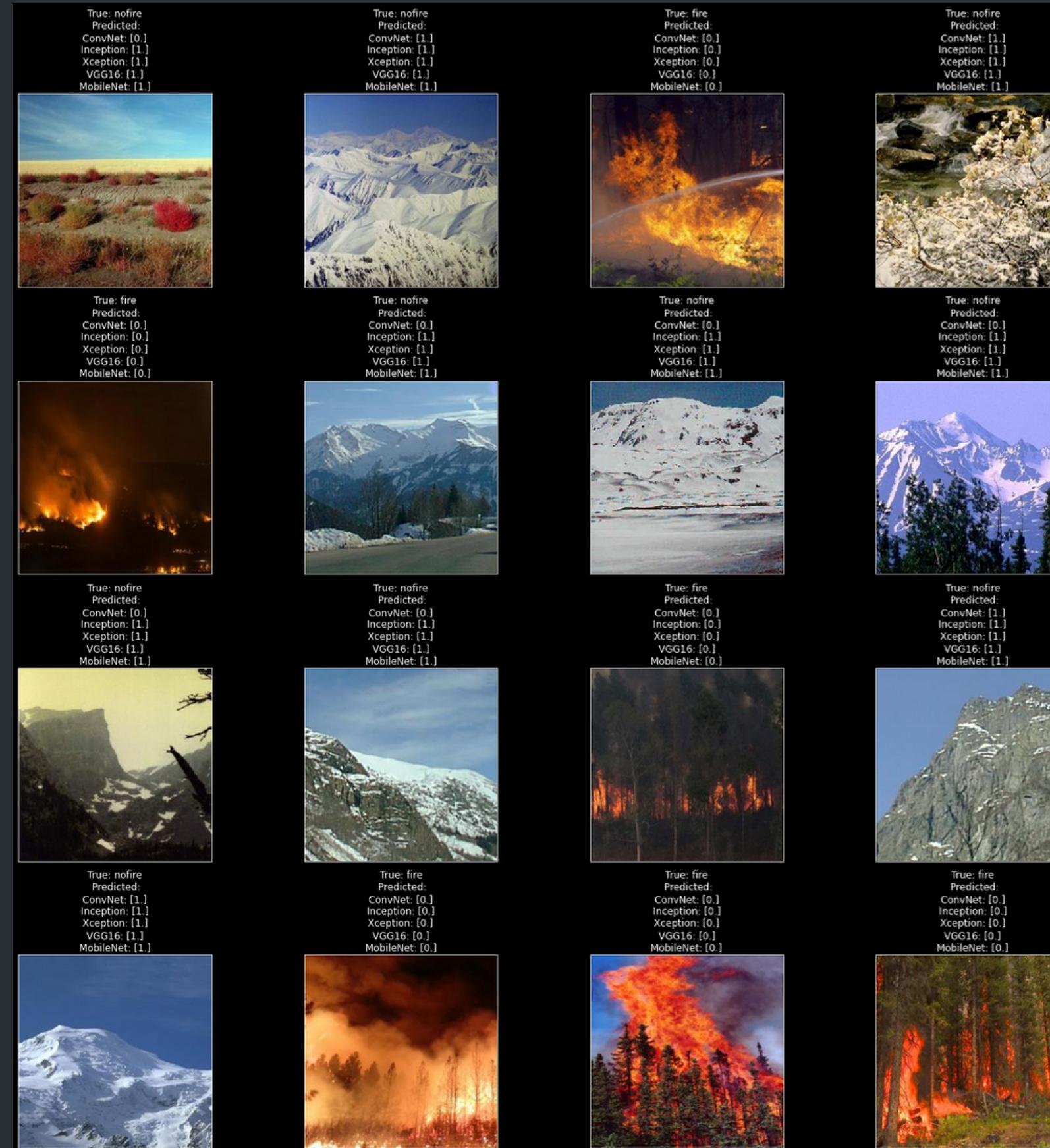


Figura 21- Figura que ilustra as imagens previstas de cada modelo no conjunto de teste 1.

Resultados - Imagens previstas conjunto de teste 2



Figura 22 – Figura que ilustra as imagens previstas de cada modelo no conjunto de teste 2.



Conclusão

Conclusão

- Neste estudo investigamos o desempenho de algoritmos de Visão Computacional tais sendo: ConvNet, Inception, Xception, VGG16 e MobileNet. Cada algoritmo foi avaliado com base nos valores das métricas de classificação das tabelas anteriores. Fizemos experiências com diferentes otimizações e técnicas que aumentaram consideravelmente o desempenho dos modelos, produzindo precisão de previsão em AUC de [0.93, 0.91], [0.97, 0.88] [0.97, 0.88], [0.93, 0.79] e [0.98, 0.90] para os modelos ConvNet, Inception, Xception, VGG16 e MobileNet no conjunto de dados testset1 e testset2 respectivamente. Com isso, concluimos que o modelo que obteve o melhor resultado foi o MobileNet dentre todas as métricas utilizadas. Também foi levado em consideração que, a criação do algoritmo já foi pensada para sistemas de pequeno porte, sendo sistemas embarcados.
- Também foram analisados aspectos referentes a utilização de sistemas embarcados em uma aplicação voltada a detecção de incêndios florestais. Os resultados obtidos a partir dos cenários de testes puderam confirmar a possibilidade de utilização destes sistemas em conjunto com modelos treinados e um sistema baseado em nuvem.



Trabalhos futuros

Trabalhos futuros

- Para trabalhos futuros devem ser considerados os seguintes aspectos:
- Teste dos métodos de conversão e armazenamento de energia alternativos em cenários reais e determinação da autonomia do sistema;
- Teste com uma rede maior de sensores, isto é, realizando a checagem das melhores distâncias entre os nós para melhoria da qualidade de transmissão;
- Implementação de criptografia ponta-a-ponta entre nó sensor e gateway;
- Implantação de mecanismos de troca de canais para evitar colisões de pacotes;



Referências Bibliográficas

Dennison et al. (2014), "Wildfires and Climate Change", <https://www.c2es.org/content/wildfires-and-climate-change/>, Abril

C. Altman, (2020) "Inteligência Artificial está em alta em 2020", https://www.em.com.br/app/noticia/ciencia/2020/02/06/interna_ciencia,1119794/inteligencia-artificial-esta-em-alta-em-2020.shtml, Abril

Stefanini, (2019) "Inteligência Artificial nos setores da economia: entenda as funcionalidades" <https://stefanini.com/pt-br/trends/artigos/inteligencia-artificial-nos-setores-da-economia>, Abril

R.Metz, (2020)"How AI is helping spot wildfires faster" <https://edition.cnn.com/2019/12/05/tech/ai-wildfires/index.html> , Abril

Jia Deng et al (2012), "ImageNet: A large-scale hierarchical image database", <https://paperswithcode.com/dataset/imagenet>, Junho

Souza, E. G, (2019) "Entendendo o que é Matriz de Confusão com Python", <https://medium.com/data-hackers/entendendo-o-que-%C3%A9-matriz-de-confus%C3%A3o-com-python-114e683ec509>, Junho

Rodrigues, (2019), "Métricas de Avaliação: acurácia, precisão, recall... quais as diferenças?" <https://vitorborbarodrigues.medium.com/m%C3%A9tricas-de-avalia%C3%A7%C3%A3o-acur%C3%A1cia-precis%C3%A3o-recall-quais-as-diferen%C3%A7as-c8f05e0a513c>, Maio

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016) Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 28-18-2826)

François Chollet et al (2016), "Xception: Deep learning with depthwise separable convolutions", CoRR, [abs/1610.02357](https://arxiv.org/abs/1610.02357)

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. [http://arxiv.org/abs/1704.04861](https://arxiv.org/abs/1704.04861).

Tsang, S. K (2018), "Review: Xception — With Depthwise Separable Convolution, Better Than Inception-v3 (Image Classification)",
Hassan, M. U, (2018), "VGG16 – Convolutional Network for Classification and Detection"

Tonghao Chen, (2019). "Implementing Efficient and Multi-Hop Image Acquisition In Remote Monitoring IoT systems using LoRa Technology" In harvest.