

UNIVERSIDADE FEDERAL DE ITAJUBÁ  
PROGRAMA DE PÓS GRADUAÇÃO EM  
CIÊNCIA E TECNOLOGIA DA COMPUTAÇÃO

Algoritmos de Processamento Digital de Imagens para  
Rastreamento de Linhas de Transmissão de Energia

Antonio Josivaldo Dantas Filho

Itajubá, 06 de Dezembro de 2021

UNIVERSIDADE FEDERAL DE ITAJUBÁ

PROGRAMA DE PÓS GRADUAÇÃO EM  
CIÊNCIA E TECNOLOGIA DA COMPUTAÇÃO

Antonio Josivaldo Dantas Filho  
Algoritmos de Processamento Digital de Imagens para  
Rastreamento de Linhas de Transmissão de Energia

Dissertação submetida ao Programa de Pós-Graduação em  
Ciência e Tecnologia da Computação (POSCOMP) como  
parte dos requisitos para obtenção do Título de Mestre em  
Ciência e Tecnologia da Computação.

**Área de Concentração:** Matemática Computacional

**Linha de Pesquisa:** Inteligência Artificial

**Orientador:** Prof. Dr. Alexandre Carlos Brandão Ramos

Itajubá, 06 de Dezembro de 2021

# Agradecimentos

Gostaria de agradecer primeiramente a Deus por iluminar e guiar esta caminhada com saúde e força para superar todas as dificuldades e desafios encontrados até o momento. Agradeço a todas as pessoas que me ajudaram durante minha pesquisa. Em particular, aos meus filhos Maria Cecília e Samuel Dantas por me compreenderem na minha ausência e darem forças para continuar minha caminhada. À minha maravilhosa esposa Eduarda Dantas pelo apoio incondicional nos meus estudos e carreira. Aos meus pais, Maria e Antonio Dantas, por serem sempre minha base e inspiração para superar os desafios. À minha irmã, Anne Dantas, pela parceria e carinho. Aos meus colegas de estudo, principalmente Leandro Diniz e Wander Martins, pela assistência desde o início do mestrado. Ao meu orientador Alexandre Carlos Brandão Ramos, por estar sempre disponível, me ajudar e guiar meus estudos da maneira certa, além de, oferecer diversas oportunidades únicas para minha trajetória acadêmica. Ao meu orientador externo no estágio de curto prazo no exterior, Peter Funk, por seu grande amparo pessoal e orientação profissional na Suécia. À Unifei e toda equipe que a integra, em especial a Pró-Reitoria do Programa de Pós-Graduação e o Instituto de Matemática e Computação. E, por fim, a todas as pessoas que me auxiliaram e acompanharam do início ao fim desta jornada direta ou indiretamente. Este estudo foi parcialmente financiado pelo Código Financeiro 001 da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior Brasil (CAPES).

*Se a educação sozinha não transforma a sociedade, sem ela tampouco a sociedade muda.*

Paulo Freire

# Resumo

As linhas de transmissão de energia de alta potência são de grande importância para o funcionamento de todos os setores da sociedade. Para garantir a confiabilidade e disponibilidade do fornecimento de energia, são realizadas inspeções regulares e ocasionais, em geral, de forma manual. Essas inspeções procuram identificar anomalias na rede de transmissão e sinais de perigo às estruturas. Veículos aéreos não tripulados, popularmente conhecidos como drones, têm sido cada vez mais empregados em atividades repetitivas e tediosas, devido ao baixo custo e aumento na segurança, e podem ser aplicados ao rastreamento dessas estruturas. Contudo, a variedade de ambientes e torres, erro nas medições da localização, e alto custo são obstáculos difíceis de superar para realização com exatidão e qualidade da inspeção. Este trabalho tem como principal objetivo estudar, desenvolver e comparar técnicas de processamento de imagens em tempo real para determinar movimentos a serem executadas pelo drone na correção da trajetória, e assim, executar um rastreamento das linhas de transmissão com alta precisão e baixo custo. Diversas soluções são encontradas na literatura, e estão localizadas e revisadas de forma sistemática neste estudo. A respeito das análises na linha de hardware, foram comparadas placas controladoras e *frameworks*, além dos softwares de estações de solo e simuladores, com principal enfoque no processamento de imagens usando detector de bordas, aprendizado profundo e aprendizado por reforço profundo. Ademais, em relação ao treinamento das redes, foi confeccionado um *dataset* para identificação de linhas, e outro para ações a serem executadas. Outrossim, com o intuito de avaliar o desempenho da solução, um cenário virtual realista baseado em um percurso real foi produzido. Em suma, um sistema baseado em um protótipo aparelhado com dois drones quadricóptero equipados apenas com uma câmera e um computador, trabalhando em conjunto com um algoritmo cooperativo e processamento de imagem por aprendizado profundo, resultou em um breve rastreamento em linha de transmissão de energia, simulando uma inspeção em ambiente real.

# Abstract

High power power transmission lines are of great importance for the functioning of all sectors of society. To guarantee the reliability and availability of the energy supply, regular and occasional inspections are carried out, always manually. These inspections seek to identify anomalies in the transmission network and signs of danger to structures. Unmanned aerial vehicles, popularly known as drones, have been increasingly employed in repetitive and tedious activities, due to their low cost and increased safety, and can be applied to tracking these structures. However, the variety of environments and towers, error in location measurements, and high cost are difficult obstacles to overcome for accurate and quality inspection. The main objective of this work is to study, develop and compare real-time image processing techniques to determine movements to be performed by the drone in trajectory correction, and thus perform a tracking of transmission lines with high precision and low cost. Several solutions are found in the literature, and are located and systematically reviewed in this study. Regarding the analysis of the hardware line, controller boards and frameworks were compared, as well as ground station and simulator software, with the main focus on image processing using edge detector, deep learning and reinforcement learning. Furthermore, in relation to the training of networks, a dataset was made for identifying lines, and another for actions to be performed. Also, in order to evaluate the solution's performance, a realistic virtual scenario based on a real route was produced. In short, a system based on a prototype equipped with two quadcopter drones equipped with only a camera and a computer, working in conjunction with a cooperative algorithm and deep learning image processing, resulted in a brief tracking in a power transmission line, simulating an inspection in a real environment.

# Sumário

	<b>Lista de Figuras . . . . .</b>	<b>IX</b>
	<b>Lista de Tabelas . . . . .</b>	<b>XII</b>
<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>1</b>
1.1	Motivação . . . . .	4
1.2	Escopo da pesquisa . . . . .	6
1.3	Trabalhos relacionados . . . . .	6
1.4	Objetivos . . . . .	6
1.4.1	Objetivos Específicos . . . . .	7
1.5	Estrutura do Trabalho . . . . .	7
<b>2</b>	<b>MATERIAIS E MÉTODOS . . . . .</b>	<b>9</b>
<b>2.1</b>	<b>ARP . . . . .</b>	<b>9</b>
2.1.1	Dispositivos de Controle de Voo . . . . .	11
2.1.2	ArduPilot . . . . .	12
2.1.3	DJI Naza . . . . .	13
2.1.4	Pixhawk . . . . .	13
<b>2.2</b>	<b>Frameworks . . . . .</b>	<b>13</b>
2.2.1	ROS . . . . .	14
2.2.1.1	MAVROS . . . . .	14
2.2.2	MAVSDK . . . . .	15
<b>2.3</b>	<b>Interface de usuário para Voo . . . . .</b>	<b>15</b>
2.3.1	MAVProxy . . . . .	16

2.3.2	Mission Planner . . . . .	16
2.3.3	QGroundControl . . . . .	17
2.4	Simulação . . . . .	17
2.4.1	JMAVSim . . . . .	18
2.4.2	Gazebo . . . . .	18
2.4.3	Airsim . . . . .	19
2.5	RSL . . . . .	20
2.6	Legislação . . . . .	22
2.7	Visão Computacional . . . . .	23
2.7.1	Pré-Processamento . . . . .	23
2.7.1.1	Redimensionamento . . . . .	23
2.7.1.2	Componentes . . . . .	24
2.7.1.3	Ruído . . . . .	24
2.7.2	Detecção de bordas . . . . .	24
2.7.2.1	Detectores de borda . . . . .	25
2.7.3	Aprendizado de máquina . . . . .	27
2.7.4	Aprendizado Profundo . . . . .	29
2.7.4.1	Rede Neural Convolutacional . . . . .	29
2.7.5	Detector de múltiplas caixas de passagem única . . . . .	30
2.7.6	Aprendizado por Reforço profundo . . . . .	31
2.7.6.1	Algoritmos de Aprendizado por Reforço . . . . .	33
2.7.7	<i>Frameworks</i> para Visão Computacional . . . . .	34
2.7.7.1	OpenCV . . . . .	34
2.7.7.2	CNTK . . . . .	35
2.7.7.3	PyTorch . . . . .	35
2.7.7.4	TensorFlow . . . . .	36

2.7.7.5	Keras . . . . .	36
2.7.7.6	TensorBoard . . . . .	36
2.7.8	Estágio no Exterior . . . . .	37
<b>3</b>	<b>DESENVOLVIMENTO . . . . .</b>	<b>39</b>
<b>3.1</b>	<b>Modelagem . . . . .</b>	<b>39</b>
<b>3.2</b>	<b>Arquitetura do Sistema . . . . .</b>	<b>41</b>
<b>3.3</b>	<b>Detecção de Bordas . . . . .</b>	<b>42</b>
<b>3.4</b>	<b>Aprendizagem Profunda . . . . .</b>	<b>45</b>
<b>3.5</b>	<b>Aprendizado por reforço . . . . .</b>	<b>46</b>
<b>4</b>	<b>EXPERIMENTOS E RESULTADOS . . . . .</b>	<b>49</b>
<b>4.1</b>	<b>RSL . . . . .</b>	<b>49</b>
<b>4.2</b>	<b>Legislação . . . . .</b>	<b>51</b>
<b>4.3</b>	<b>Ambiente de simulação . . . . .</b>	<b>53</b>
<b>4.3.1</b>	<b>Torres . . . . .</b>	<b>53</b>
<b>4.3.2</b>	<b>Mundo . . . . .</b>	<b>53</b>
<b>4.4</b>	<b>Preparação do <i>Dataset</i> . . . . .</b>	<b>55</b>
<b>4.5</b>	<b>Análise das Soluções . . . . .</b>	<b>56</b>
<b>4.6</b>	<b>Detecção de bordas . . . . .</b>	<b>57</b>
<b>4.7</b>	<b>Aprendizado Profundo . . . . .</b>	<b>58</b>
<b>4.7.1</b>	<b>Monoclasse . . . . .</b>	<b>59</b>
<b>4.7.2</b>	<b>Multiclasse . . . . .</b>	<b>60</b>
<b>4.8</b>	<b>Aprendizado por reforço . . . . .</b>	<b>62</b>
<b>4.8.1</b>	<b>Recompensa . . . . .</b>	<b>62</b>
<b>4.8.2</b>	<b>Algoritmos . . . . .</b>	<b>64</b>
<b>4.9</b>	<b>Experimentos no Percurso . . . . .</b>	<b>65</b>

4.9.1	Variáveis de Ambiente . . . . .	66
4.9.2	Algoritmo de Controle em Pares de <i>buffer</i> . . . . .	67
4.9.3	Comparação dos Métodos . . . . .	68
4.9.4	Experimento Externo . . . . .	70
4.9.4.1	Drone Real . . . . .	70
4.9.4.2	Execução do Algoritmo . . . . .	72
5	<b>CONCLUSÃO . . . . .</b>	<b>74</b>
5.0.1	Limitações do estudo . . . . .	74
5.0.2	Principais Dificuldades . . . . .	75
5.0.3	Principais Contribuições . . . . .	75
5.1	Trabalhos Futuros . . . . .	76
5.2	Considerações Finais . . . . .	77
6	<b>APÊNDICE A - REVISÃO SISTEMÁTICA DA LI- TERATURA . . . . .</b>	<b>78</b>
6.1	Definição da pesquisa . . . . .	78
6.2	Termos de pesquisa . . . . .	78
6.3	Critérios de seleção e qualidade . . . . .	79
6.4	Condução . . . . .	81
6.5	Análises . . . . .	82
6.6	Discussão . . . . .	83
7	<b>APÊNDICE B - ESPECIFICAÇÕES DO ARP UTI- LIZADO NO ESTUDO . . . . .</b>	<b>85</b>
8	<b>APÊNDICE C - RESUMO DO RELATÓRIO DE ESTÁGIO NO EXTERIOR . . . . .</b>	<b>87</b>

<b>8.1</b>	<b>Introduction . . . . .</b>	<b>87</b>
<b>8.2</b>	<b>Methodology . . . . .</b>	<b>87</b>
<b>8.3</b>	<b>Provisional Results . . . . .</b>	<b>88</b>
	<b>Anexo A . . . . .</b>	<b>92</b>
	<b>Anexo B . . . . .</b>	<b>93</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>94</b>

# Lista de Figuras

Figura 1 – Defeitos comuns em estruturas de linhas de transmissão de energia. Fonte: (SILVA, 2011) Adaptado . . . . .	2
Figura 2 – Ocorrências comuns no ambiente próximo a linha de transmissão de energia. Fonte: (SILVA, 2011; GLOBO, 2019) Adaptado . . .	3
Figura 3 – Modos comuns de inspeção em linhas de transmissão de energia. Fonte: (RESENDE, 2017; RANGEL; KIENITZ; BRANDÃO, 2009) . . . . .	4
Figura 4 – Tipos diferentes de estruturas e ambientes de linhas de transmissão de energia. Fonte: (ELIAS, 2015; RANGEL; KIENITZ; BRANDÃO, 2009) . . . . .	5
Figura 5 – Orientação de uma Aeronave. Fonte: Próprio Autor . . . . .	10
Figura 6 – Arquitetura do hardware e comunicação. Fonte: Próprio Autor .	11
Figura 7 – Controladoras de Voo de Baixo Custo. Fonte: Próprio Autor . .	12
Figura 8 – Estação de controle de solo Mission Planner à esquerda, MAV-Proxy no centro e QGroundControl à direita. Fonte: (WILLEE, 2019) . . . . .	16
Figura 9 – Simuladores JMAVSIM à esquerda e Gazebo à direita. Fonte: (TEAM, 2019) . . . . .	18
Figura 10 – Mundo 'Vizinhança' do Airsim, com os dados dos sensores (direita) e opções de alteração de clima (esquerda). Fonte: (MICROSOFT, 2019) . . . . .	20
Figura 11 – Etapas para o desenvolvimento de uma revisão sistemática da literatura. Fonte: Próprio Autor . . . . .	21
Figura 12 – Filtro gradiente Canny. Fonte: (FILHO et al., 2020) . . . . .	25
Figura 13 – Exemplificação da Transformada da linha Hough Padrão, Probabilística e <i>Line Segment Detector</i> . Fonte: (HUAMÁN, 2019) Adaptada . . . . .	26
Figura 14 – Estrutura de neurônio artificial com a função de ativação. Fonte: Próprio Autor . . . . .	28

Figura 15 – Exemplo de arquitetura de uma CNN. Fonte: (BRITO et al., 2019) . . . . .	30
Figura 16 – Exemplo de arquitetura de aprendizado DRL. Fonte: Próprio autor . . . . .	32
Figura 17 – Interface do TensorBoard. Fonte: (KUDLUR et al., 2020) . . . . .	37
Figura 18 – Solução de rastreamento proposto. Fonte: Próprio Autor . . . . .	39
Figura 19 – Fluxograma da Arquitetura do Sistema. Fonte: Próprio Autor . . . . .	42
Figura 20 – Fluxograma do funcionamento do algoritmo de ED. Fonte: Próprio Autor . . . . .	43
Figura 21 – Função arco tangente hiperbólica. Fonte: (PINTO et al., 2019) . . . . .	44
Figura 22 – Fluxograma do funcionamento do algoritmo SSD. Fonte: Próprio Autor . . . . .	46
Figura 23 – Fluxograma do funcionamento do algoritmo de Aprendizado por reforço. Fonte: Próprio Autor . . . . .	47
Figura 24 – Fluxograma para solicitação de voo seguindo a legislação atual. Fonte: Próprio Autor . . . . .	52
Figura 25 – Área de atuação na torre de alta tensão. Fonte: Próprio Autor . . . . .	53
Figura 26 – Torres de transmissão de alta tensão utilizadas no simulador Airsim. Fonte: Próprio Autor . . . . .	54
Figura 27 – Ambientes de testes utilizadas no simulador Airsim. Fonte: Próprio Autor . . . . .	54
Figura 28 – Imagens do <i>Dataset</i> rotuladas, rótulos em linhas à esquerda, visão de cima no centro, e visão lateral à direita. Fonte: Próprio Autor . . . . .	55
Figura 29 – Função de Perda dos Modelos SSDs Monoclasse. Fonte: Próprio Autor . . . . .	59
Figura 30 – Média da Precisão dos Modelos SSDs Monoclasse. Fonte: Próprio Autor . . . . .	60
Figura 31 – Média da Revocação dos Modelos SSDs Monoclasse. Fonte: Próprio Autor . . . . .	60
Figura 32 – Função de Perda dos Modelos SSDs Multiclasse. Fonte: Próprio Autor . . . . .	61

Figura 33 – Média ponderada da Precisão dos Modelos SSDs Multiclasse. Fonte: Próprio Autor . . . . .	62
Figura 34 – Média da Revocação dos Modelos SSDs Multiclasse. Fonte: Próprio Autor . . . . .	63
Figura 35 – Resultados do Treinamento de Aprendizado por Reforço Profundo. Fonte: Próprio Autor . . . . .	65
Figura 36 – Percurso Real à esquerda e no Simulador Airsim à direita. Fonte: Próprio Autor . . . . .	66
Figura 37 – Gráficos das variáveis analisadas. Fonte: Próprio Autor . . . . .	70
Figura 38 – Eficiência mecânica do motor e da hélice baseado no diâmetro da hélice. Fonte: (DANTAS et al., 2021) . . . . .	71
Figura 39 – Tela do QGroundstation para ajuste de PID . . . . .	72
Figura 40 – Imagens da execução em campo. baixo Fonte: Próprio Autor . . . . .	73
Figura 41 – Processo de seleção de estudos primários (esquerda) e Filtros em cada mecanismo de busca (direita). Fonte Próprio Autor . . . . .	82
Figura 42 – Imagem real do ARP utilizado no estudo. Fonte: Próprio Autor . . . . .	85
Figura 43 – Process outline of project (adapted from presentation by Peter Funk) . . . . .	89
Figura 44 – Communication and relative positioning and internal pose estimation (BROWN et al., 2021) . . . . .	90
Figura 45 – SISANT da aeronave utilizada no estudo. Fonte: (ANAC, 2021) . . . . .	92
Figura 46 – Solicitações de voo usadas no estudo. Fonte: (VARGAS, 2019) . . . . .	93

# Lista de Tabelas

Tabela 1 – Comparação de estruturas de ARPs . . . . .	9
Tabela 3 – Características identificadas nos artigos (MARTINS et al., 2020) . . . . .	50
Tabela 4 – Experimento utilizando ED em imagens testes com diferentes parâmetros . . . . .	57
Tabela 5 – Experimento utilizando os algoritmos e determinando as ações .	58
Tabela 6 – Comparação de alguns modelos de SSD populares . . . . .	59
Tabela 7 – Variáveis de Ambiente para teste dos modelos . . . . .	67
Tabela 8 – Comparação do Tempo médio de treinamento e execução . . . .	69
Tabela 9 – Parâmetros finais obtidos para ajuste de PID . . . . .	72
Tabela 10 – Critérios de Exclusão . . . . .	80
Tabela 11 – Critérios de Inclusão . . . . .	80
Tabela 12 – Critérios de Qualidade . . . . .	80
Tabela 13 – Componente para operação do ARP . . . . .	86

# Lista de abreviaturas e siglas

A2C	<i>Advantage Actor Critic</i>	34
ACM	<i>Association for Computing Machinery</i>	21
Adam	<i>Adaptive Momentum Estimation</i>	28
AMASS	Anomalia Magnética do Atlântico Sul	5
ANAC	Agência Nacional de Aviação Civil	5
Anatel	Agência nacional de telecomunicação	5
ANEEL	Agência Nacional de Energia Elétrica	1
API	<i>Application Programming Interface</i>	15
ARP	Aeronaves Remotamente Pilotadas	3
CE	Critérios de Exclusão	79
CEMIG	Companhia Energética de Minas Gerais	1
CI	Critérios de Inclusão	79
CNN	<i>Convolutional Neural Network</i>	29
CNTK	<i>Microsoft Cognitive Toolkit</i>	35
CPU	<i>Central Processing Unit</i>	19
CQ	Critérios de Qualidade	79
DECEA	Departamento de Controle do Espaço Aéreo	5
DL	<i>Deep Learning</i>	29
DQN	<i>Deep Q Learning</i>	33
DRL	<i>Deep Reinforcement Learning</i>	33
ED	<i>Edge Detector</i>	24
EPE	Empresa de Pesquisa Energética	1
FC	<i>Flight Control</i>	12
FN	Falsos Negativos	56
FP	Falsos Positivos	56
FPN	<i>Feature Pyramid Network</i>	31
FPS	<i>Frames Per Second</i>	35
GCS	<i>Ground Control Station</i>	15
GPS	<i>Global Positioning System</i>	4

GPU	<i>Graphics Processing Unit</i>	19
GUI	<i>Graphical User Interface</i>	16
HITL	<i>Hardware in the Loop Simulation</i>	19
IEEE	<i>Institute of Electrical and Electronics Engineers</i>	21
IMU	<i>Inertial Measurement Unit</i>	4
LiDAR	<i>Light Detection and Ranging</i>	82
LSD	<i>Line Segment Detector</i>	26
MAVLink	<i>Micro Air Vehicle Link</i>	14
MBC	Manutenção Baseada em Condição	2
MBT	Manutenção Baseada em Tempo	2
MDH	<i>Mälardalens högskola</i>	37
MDP	<i>Markov Decision Process</i>	31
ML	<i>Machine Learning</i>	27
MLP	<i>MultiLayer Perceptron</i>	27
NA	Não se Aplica	58
NED	<i>North, East and Down</i>	10
OGW	<i>Over Ground Wire</i>	83
ONNX	<i>Open Neural Network Exchange</i>	35
ONS	Operador Nacional do Sistema Elétrico	1
OpenCV	<i>Open Source Computer Vision Library</i>	34
PD	<i>Pitch Diameter</i>	71
PID	Proporcional Integral Derivativo	11
PPO	<i>Proximal Policy Optimization</i>	34
QE	Questões Específicas	78
QP	Questão Principal	78
RAM	<i>Random Access Memory</i>	20
RCNN	<i>Region Based Convolutional Neural Networks</i>	31
ReLU	<i>Rectified Linear Unit</i>	29
ResNet	<i>Residual Neural Network</i>	89
RL	<i>Reinforcement Learning</i>	31
RMS	<i>Root Mean Square</i>	28
RNA	Redes Neurais Artificiais	27
ROI	Regiões de Interesse	76

ROS	<i>Robot Operating System</i>	14
RSL	Revisão Sistemática da Literatura	6
SAR	<i>Search And Rescue</i>	37
SARPAS	Solicitação de Acesso de Aeronaves Remotamente Pilotadas	23
SARUAV	<i>Search And Rescue Unmanned Aerial Vehicles</i>	87
SDF	<i>Simulation Description Format</i>	18
SDK	<i>Software Development Kit</i>	14
SDRAM	<i>Synchronous Dynamic Random-Access Memory</i>	57
SISANT	Sistema de Aeronaves Não Tripuladas	22
SITL	<i>Software In The Loop</i>	16
SS	<i>Search String</i>	20
SSD	<i>Single Shot MultiBox Detector</i>	30
TH	Transformada de Hough	25
THP	Transformada de Hough Probabilístico	26
UAV	<i>Unmanned Aerial Vehicle</i>	3
UDP	<i>User Datagram Protocol</i>	15
Unifei	Universidade Federal de Itajubá	22
VANT	Veículos Aéreos Não Tripulados	3
VGG	<i>Visual Geometry Group</i>	30
Viscap	Visão, Sistemas de Computação e Aplicações	51
VLOS	<i>Visual Line-Of-Sight</i>	22
VP	Verdadeiros Positivos	56
XML	<i>eXtensible Markup Language</i>	18

# 1 Introdução

O conceito de infraestrutura é vital para o desenvolvimento de qualquer sociedade. Além de ser parte integrante da existência humana, desempenha um papel crucial na promoção do crescimento econômico e, assim, contribui para a redução da disparidade econômica (SRINIVASU; RAO, 2013).

Torres de transmissão de energia elétrica são enormes infraestruturas com grande importância para o abastecimento elétrico de um país, pois são elas que suportam os cabos que transmitem a energia do local de geração até o consumidor final. A confiabilidade da distribuição continua sendo um dos tópicos mais importantes no setor de energia elétrica. Devido a correlação direta com a satisfação do cliente. A ruptura ou mesmo a danificação dessas estruturas pode ter consequências significativas para a economia (KIRTLEY, 2020).

De acordo com o ONS (Operador Nacional do Sistema Elétrico), em 2025 a extensão do comprimento da transmissão de eletricidade, utilizando linhas de alta tensão, no Brasil atingirá 184.054 km de extensão, sendo o número atual 145.600 km. O consumo de energia elétrica no Brasil somou 474.231GWh em 2020, um aumento de 38% em relação aos 10 anos anteriores, segundo o EPE (Empresa de Pesquisa Energética) (MACHADO, 2021). Além disso, a CEMIG (Companhia Energética de Minas Gerais) contará com um investimento governamental do programa Mais Energia no valor de 5 bilhões de reais, criando 200 novas subestações de distribuição de energia, aumentando para 615 unidades e 3.100 quilômetros de novas linhas de alta tensão até 2027 no estado de Minas Gerais (BIANCHETTI, 2021).

A fim de garantir a continuidade e qualidade desse sistema complexo, as atividades de manutenção e inspeção são regulamentadas pela ONS e fiscalizadas pela ANEEL (Agência Nacional de Energia Elétrica), que determina um plano mínimo de ação e periodicidade (ONS, 2021). Essas medidas são necessárias pois as estruturas de transmissão que possuem longa extensão estão sujeitas a danos causados por ações da natureza e do envelhecimento, necessitando, portanto, de verificações minuciosas de forma periódica ou após grandes catástrofes ambientais. Ademais, é

de competência das concessionárias montar um plano de inspeções para esses casos, para que as manutenções ocorram regularmente de forma especializada preventiva e preditiva, e de acordo com as características de suas estruturas. As estratégias de preservação podem ser de MBT (Manutenção Baseada em Tempo), definido por intervalos fixos de calendário, horas de operação ou ciclos de operação, ou MBC (Manutenção Baseada em Condição), que é uma extensão da estratégia MBT e utilizar técnicas de teste não invasivas para avaliar a condição do equipamento (KIRTLEY, 2020).

Essas inspeções procuram identificar os problemas na estrutura e cabos de alta tensão de classe 2, que são condutores de energia encordoados. Esses problemas podem interferir no funcionamento do trajeto desde a fonte de alimentação ao consumidor final, ou que reduza a vida útil do sistema, tais como: ruptura parcial ou total de fios, problemas do isolador e suas conexões, sobrecarga de energia no fio, falta de peças, objetos prendidos que não são pertencentes a estrutura, entre outras situações que possam comprometer a disponibilidade. A figura 1 apresenta exemplos reais da ocorrência de alguns desses problemas.



Figura 1 – Defeitos comuns em estruturas de linhas de transmissão de energia.  
Fonte: (SILVA, 2011) Adaptado

Problemas alheios à estrutura podem ser identificados nas inspeções regula-

res e de manutenção. Consistem em ocorrências no ambiente que oferecem riscos futuros as pessoas próximas ou ao funcionamento regular da transmissão de energia, problemas que podem ser apresentados por: árvores próximas à estrutura ou condutores, supressão irregular em meia encosta, invasões indevidas na faixa de servidão, erosões naturais ou de retirada para aterro, entre outros. A figura 2 apresenta as situações mais comuns que podem comprometer as estruturas ou segurança das pessoas.



Figura 2 – Ocorrências comuns no ambiente próximo a linha de transmissão de energia. Fonte: (SILVA, 2011; GLOBO, 2019) Adaptado

Em geral, as inspeções simples de rotina podem ser realizadas com uma patrulha pedestre ou automóvel terrestre, composto por um conjunto de eletricitas, localizados em plataformas no local com o auxílio de caminhões elevadores, ou em plataformas isoladas com auxílio de binóculos. As mais específicas, por sua vez, são feitas por um helicóptero tripulado contendo sensores para aquisição de dados, realizando um voo paralelo à linha de transmissão e um operador realiza a filmagem da linha (SILVA; MUNARO, 2008).

Esses métodos, embora funcionais, apresentam muitas desvantagens: a primeira se faz restrita por conta da limitação do ponto de observação e equipamento, enquanto a segunda pela disponibilidade dos equipamentos e veículos utilizados. Ambas soluções exigem alto investimento por curso e, também, são de alto risco de acidentes aos operadores (SILVA; MUNARO, 2008; RESENDE, 2017).

A ARP (Aeronaves Remotamente Pilotadas), em inglês UAV (*Unmanned Aerial Vehicle*), também chamada de VANT (Veículos Aéreos Não Tripulados), e comumente conhecidos como drones, já foram utilizados em muitas aplicações civis e militares tradicionais com sucesso. Inclusive, recentemente, os ARPs têm

sido empregados nas inspeções de linhas de transmissão de forma manual e reduzida, em detrimento à sua limitação quanto ao campo de visão restrito, escassez de operadores especializados e tempo de atuação (BRITO et al., 2019).

Os métodos tradicionais de inspeção são ilustrados na figura 3 abaixo.



Figura 3 – Modos comuns de inspeção em linhas de transmissão de energia Fonte: (RESENDE, 2017; RANGEL; KIENITZ; BRANDÃO, 2009)

Uma aplicação emergente é a substituição dos meios tradicionais de inspeção de linhas de transmissão de energia por ARPs autônomos, porém novas questões são levantadas, dentre elas a forma de atuar com menor custo, melhor desempenho e maior segurança, atendendo à legislação vigente. Dessarte, o uso de ARP para rastreamento do trajeto de uma linha de transmissão de energia atualmente é realizado através de um piloto altamente treinado ou por missões simples pré-programadas, utilizando *waypoints* baseadas em IMU (*Inertial Measurement Unit*)<sup>1</sup> acoplados na aeronave, além de coordenadas GPS (*Global Positioning System*) com conhecimento prévio do ambiente para a realização da missão (PAUL, 2006).

## 1.1 Motivação

Identificar os elementos para auxiliar na inspeção utilizando um ambiente de navegação e controle de altitude de forma precisa é um grande desafio para essa aplicação, em especial em situações onde o sinal de GPS e dados IMU pode não ser confiáveis, ou a condução do operador possa ser afetada. Falhas de precisão em instrumentos de medição, interferências externas, como nível de absorção da

---

<sup>1</sup> Dispositivo eletrônico que mede a força específica, taxa angular e orientação do corpo, utilizando um conjunto de sensores como: acelerômetros, giroscópios e/ou bússolas

energia eletromagnética, AMASS (Anomalia Magnética do Atlântico Sul) e condições climáticas adversas são comuns. A limitação da perspectiva da visão do operador, informações limitadas ou incorretas de coordenadas das torres de linhas de transmissão, particularidade de solo e estrutura, também contribuem para impossibilitar quase por completo o uso dessa técnica. Conseguir uma solução de baixo custo, ampla, eficiente e segura, principalmente em um país com grandes extensões e recursos financeiros limitados em infraestrutura, é um grande desafio (TRIVEDI et al., 2005).

Além das questões levantadas, as estruturas das linhas de transmissão são instaladas em ambientes diferentes, podendo existir corredores em centros urbanos, regiões montanhosas, pontes, rios, entre outros. Similarmente, é comum essas estruturas se cruzarem com outros corredores de linhas de energia, que podem ser de diferentes categorias e formatos. Na figura 4 são apresentados diferentes ambientes e tipos dessas estruturas.



Figura 4 – Tipos diferentes de estruturas e ambientes de linhas de transmissão de energia. Fonte: (ELIAS, 2015; RANGEL; KIENITZ; BRANDÃO, 2009)

Recentemente estão sendo aprovadas pela ANEEL, ANAC (Agência Nacional de Aviação Civil), DECEA (Departamento de Controle do Espaço Aéreo) e Anatel (Agência nacional de telecomunicação), autorizações de projeto para exploração comercial da inspeção de linhas de transmissão de energia elétrica utilizando ARP, dessa forma, os limites de atuação do equipamento e o seu modo de operação estão sendo flexibilizados (ANAC, 2021).

## 1.2 Escopo da pesquisa

Neste trabalho é proposto o desenvolvimento de um sistema autônomo para realizar o rastreamento de linhas de transmissão de energia utilizando ARP da forma mais eficiente, e com baixo custo de implementação e operação, fundamentado, exclusivamente, no tratamento de imagens em tempo real que identifica a estrutura durante o voo e controla os movimentos da aeronave, orientando-a pelo caminho mais correto.

## 1.3 Trabalhos relacionados

Com a intenção de identificar no estado da arte em publicações formas de realizar o rastreamento de linhas de transmissão de energia utilizando ARP, e subsidiar a tomada de decisão em projetos futuros, foi realizado o processo de RSL (Revisão Sistemática da Literatura), verificando na academia como esse problema está sendo tratado e fornecendo uma análise técnica para a solução.

A RSL aplica um protocolo de revisão bem elaborado para buscar artigos em bases de dados científicas e, com o resultado da análise, responder às questões de pesquisa (KITCHENHAM, 2004). Diversos meios técnicos de rastreamento de linhas de transmissão de energia foram analisados, revelando-se um campo que pode ser bem explorado e com grande potencial de desenvolvimento.

Foram analisados 15 estudos, observou-se que cinco pesquisas utilizaram algum tipo de laser, que aumenta a precisão, mas encarece o material. Nove pesquisas utilizam visão computacional, sendo seis algoritmos de detecção de borda, mostrando uma preferência por métodos mais consolidados, contudo novas alternativas na visão computacional tem surgido. Os detalhes dos trabalhos analisados podem ser encontrados no Apêndice 6.

## 1.4 Objetivos

Esta pesquisa de mestrado tem como objetivo principal estudar e implementar um algoritmo capaz de identificar e seguir estruturas de linhas de transmissão de energia de alta tensão utilizando um ARP, em ambientes simulado e real, sem o

uso de GPS ou qualquer outro sensor IMU ativo, empregando apenas uma câmera embarcada e técnicas de visão por computador, com o menor custo possível.

O algoritmo é baseado em: captura de imagens aéreas pela aeronave durante o voo, processamento dessas imagens em tempo real, identificação da estrutura, tomada de decisão quanto a melhor opção de caminho, e comando dos movimentos da aeronave de forma a conduzi-la pelo caminho definido.

### 1.4.1 Objetivos Específicos

Os objetivos específicos desta pesquisa podem ser enumerados como:

- (a) Estudar as dificuldades, possibilidades e atuação da inspeção de linhas de transmissão de forma autônoma.
- (b) Identificar os métodos e algoritmos utilizados para rastreamento em linhas de transmissão de forma autônoma.
- (c) Construir um ARP que possua segurança e eficiência necessária para realização de rastreamento em linhas de transmissão, com o menor custo possível.
- (d) Criar um ambiente simulado que possa ser posteriormente replicado para um ARP em voos reais e controlados, de forma generalizada.
- (e) Produzir um *dataset* com imagens reais de rastreamento de torres de transmissão de energia.
- (f) Desenvolver um algoritmo eficiente capaz de comandar o ARP de forma eficiente e generalizada para rastreamento em linhas de transmissão.
- (g) Realizar comparações dos resultados obtidos com as técnicas em relação as características de um ambiente real.

## 1.5 Estrutura do Trabalho

O restante desta dissertação de mestrado está dividido em cinco outros capítulos, três apêndices e dois anexos, organizados como segue:

- O Capítulo 2 são descritos os materiais e métodos considerados e aplicados nesta pesquisa, juntamente com suas justificativas.
- No Capítulo 3 é apresentada a modelagem de identificação e rastreamento das estruturas de linhas de transmissão de energia para as técnicas citadas.
- Já o Capítulo 4, descreve os experimentos realizados e análises dos resultados obtidos em ambientes simulados, bem como os comparativos com o ambiente de acordo com cada método empregado.
- A Conclusão, no Capítulo 5, apresenta as contribuições do trabalho, dificuldades e limitações, finalizando com as propostas para trabalhos futuros e considerações da pesquisa.
- Ao final dos capítulos, os Apêndices A, B e C detalham a RSL realizada, a montagem do ARP e o resumo do estágio no exterior, respectivamente, e os Anexos A e B apresentam documentos emitidos durante o estudo.

## 2 Materiais e Métodos

Este capítulo está focado em explicar os conceitos sobre as técnicas, ferramentas e operações utilizadas na concepção do sistema apresentado. Além disso, é importante para apresentar as características, benefícios e défices das plataformas definidas e descartadas durante o processo de desenvolvimento das soluções desta pesquisa. Inicialmente são abordados os materiais e em seguida os métodos.

### 2.1 ARP

Com o rápido desenvolvimento da tecnologia, projetos de veículos autônomos estão em destaque no cenário e pesquisa acadêmica computacional, e tiveram progressos rápidos na última década (RIEDMAIER et al., 2020). Para os ARPs, as possibilidades são maiores considerando sua dinâmica, dependendo da aplicação é possível utilizar multi rotores ou asas fixas de forma autônoma e segura (MAANYU; RAJ; CHOUBEY, 2020).

Uma das propostas é construir um pequeno ARP com hardware suficiente para completar a proposta de voo, e com o menor custo possível (FILHO et al., 2020). Com a utilização de dados dos fabricantes e cálculos baseados em (BICZYSKI et al., 2020) informações sobre motores, baterias, velocidade máxima de subida, o tempo da bateria em modo estacionário e as unidades de preço médio (considerando US\$ 1000) foram organizadas na tabela 1, comparando com outro tipos e custos de ARPs de diferentes estruturas (DANTAS et al., 2021).

Tabela 1 – Comparação de estruturas de ARPs

Estrutura	Carga Máxima	Máx- Velocidade	Subida	Tempo de Bateria	Preço
Q250	1000g	5 m/s		720s	0.2
F450	1600g	4 m/s		880s	0.2
Airship	300g	2 m/s		3110s	1.0
Mavic	900g	4 m/s		1440s	1.0

O ARP utilizado na pesquisa foi um quadricóptero modelo F450 (45 cm), que engloba a categoria de ARP mini, com peso entre 2 e 25Kg (GUPTA et al., 2013). Sua escolha foi devido ao baixo custo de material para construção, capacidade de carga de até 1600g livres, baixo consumo de bateria, facilidade de manutenção, total liberdade de movimentação, boa estabilidade, velocidade e autonomia de tempo de voo quando comparado a outros ARPs similares (DANTAS et al., 2021). A figura 5 mostra os ângulos de Euler, os movimentos em relação ao espaço e orientações das coordenadas, em inglês NED (*North, East and Down*) em relação ao ARP (MAANYU; RAJ; CHOUBEY, 2020).

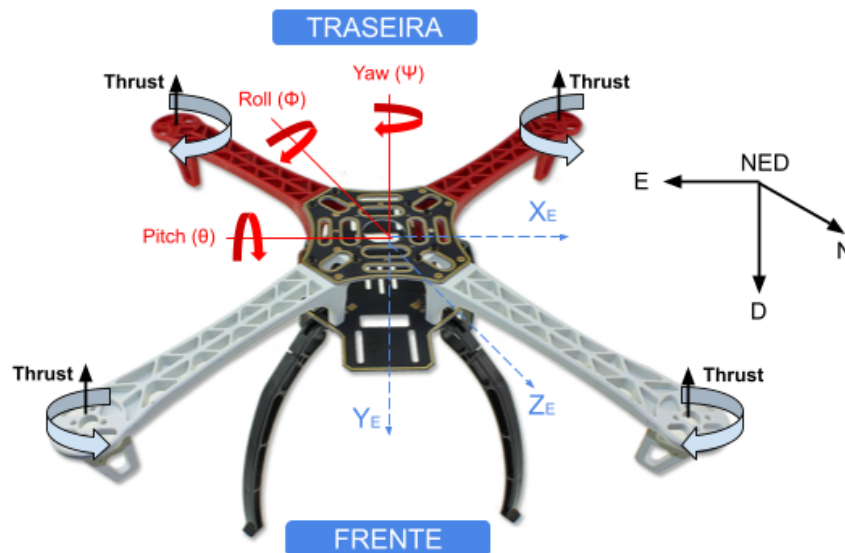


Figura 5 – Orientação de uma Aeronave. Fonte: Próprio Autor

Denominado de enxames, dois ou mais ARPs independentes podem atuar ao mesmo tempo realizando tarefas distintas, com um objetivo global final que dificilmente seria executado individualmente. Enxames são altamente escalonáveis e auto organizáveis, contudo se não forem bem estruturados podem gerar sobrecargas nos indivíduos do conjunto (SHRIT et al., 2017). Neste estudo foi utilizado um exame subdividido em um par de tarefas bem definidas.

Outra questão importante na construção do ARP é a estabilização, por esse mo-

tivo serão avaliados modos de calibração e ajustes por meio de PID (Proporcional Integral Derivativo) <sup>1</sup>. Já para a estabilização da imagem geralmente é utilizado estabilizador gimbal <sup>2</sup> instalado na parte inferior da estrutura do *frame*. Contudo, para desencarecer, é possível instalar câmeras direcionais fixas com amortecedores de impacto.

O sistema principal de operação por controladora de solo pode ser facilmente alterado para um computador de bordo nesse tipo de ARP, por exemplo Raspberry Pi, para processamento do algoritmo. A figura 6 a seguir apresenta a arquitetura do sistema como um todo, onde detalha suas comunicações e integrações entre os componentes principais.

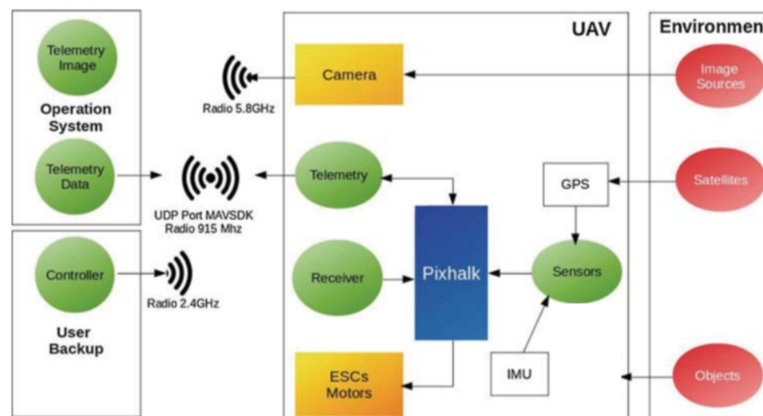


Figura 6 – Arquitetura do hardware e comunicação. Fonte: Próprio Autor

### 2.1.1 Dispositivos de Controle de Voo

Para garantir a estabilidade de um voo, é necessário a utilização de uma gama sensores que detectam o movimento do ARP e o ambiente que ele está inserido, como o IMU, de forma que haja convergência dos mesmos, aplicados a um algoritmo de processamento dos dados.

<sup>1</sup> Técnica de controle de processos para minimizado do sinal de erro pela ação proporcional, zerado pela ação integral e obtido com uma velocidade antecipativa pela ação derivativa.

<sup>2</sup> Suporte articulado que permite a rotação de um objeto em torno de um eixo estabilizado

Essa função é desempenhada pelo controlador de voo, em inglês FC (*Flight Control*) que é placa de circuito, que além garantir a estabilidade do voo também registra parâmetros e comandos do usuário (EBEID; SKRIVER; JIN, 2017).

Os FCs de baixo custo (YANG; LIN; CHEN, 2016) analisados foram determinados pela disponibilidade no laboratório de pesquisa, seus detalhes descritos a seguir, e também, apresentados na figura 7. Os detalhes desses FCs são apresentados nos tópicos a seguir.



Figura 7 – Controladoras de Voo de Baixo Custo. Fonte: Próprio Autor

### 2.1.2 ArduPilot

O ArduPilot é um dos FC pioneiros a permitir o uso de sistemas de veículos autônomos confiáveis. Possui um conjunto básico de ferramentas adequadas para quase qualquer veículo e aplicação. Possui uma comunidade extensa para apoio nas implementações do código aberto, porém foi desconsiderada no estudo por possuir um hardware limitado, desatualizado e com problemas recorrentes de firmware (TEAM, 2019).

### 2.1.3 DJI Naza

O sistema de controle de voo NAZA-M Lite, possui alta confiabilidade e estabilidade, o FC contém amortecimento interno, controladores, giroscópio de 3 eixos, acelerômetro de 3 eixos e barômetro. Também possui um algoritmo de estabilização de atitude avançada, a precisão do pairar é de aproximadamente 2,5 m na horizontal e 0,8 m na vertical (DJI, 2019).

Contudo é pouco adaptável quanto a falhas no GPS, sendo assim, dependente de seus sensores e aumentando significativamente a taxa de erros. Possui uma certa limitação de acesso a parâmetros e envio de comandos a aeronave, sendo necessário a utilização softwares específicos, e dessa forma foi desconsiderado do estudo.

### 2.1.4 Pixhawk

A placa de controle de vôo a PixHawk 4, contém além dos componentes comuns para estabilização de ARP, um processador principal FMU-STM32F765 que também permite a programação de parâmetros e comandos de movimento de forma flexível.

A Pixhawk é uma das controladoras mais estáveis e de baixo custo, possui código, protocolos e padrões abertos, além de interação contínua com a comunidade de profissionais engenheiros de computação e aeronáutica, garantindo qualidade, segurança e eficiência de produtos para sistemas autônomos modernos (MEIER, 2019). Conforme análises realizadas por (YANG; LIN; CHEN, 2016) e dos fatos apresentados anteriormente, foi determinada a Pixhawk 4 como escolha para os estudos desta pesquisa.

## 2.2 Frameworks

Os *frameworks* de controle são ferramentas que auxiliam na programação dos algoritmos e organização dos pacotes, versões, compartilhamento, entre outras facilidades. Por questões de afinidade foram utilizados nesse estudo:

- Linguagem de programação Python para todo o sistema.

- Caderno de Anotações Google Colaboratory (INC, 2019) para testes independentes.
- Gerenciador de pacotes Anaconda (WANG; OLIPHANT, 2019) para os testes em *desktop* com sistema operacional Windows.
- Repositório de códigos GitHub (WERNER; HYETT; CHACON, 2019) para controle de versionamento.
- Gerenciador de *Datasets* de Imagens Roboflow (NELSON; DWYER, 2020) para processo de rotulagem dos dados.

São os chamados kit de desenvolvimento de software, do inglês SDK (*Software Development Kit*), que contribuem para realização da comunicação do sistema computacional e as FCs. Todos os SDKs relacionados as FC escolhidas possuem o mesmo pacote de comunicação de micro veículo aéreo, em inglês MAVLink (*Micro Air Vehicle Link*), dessa forma, foi necessário definir o SDK com o melhor equilíbrio entre robustez, velocidade, simplicidade e popularidade, garantindo assim a confiabilidade e fácil manutenção do sistema, como apresentado abaixo.

## 2.2.1 ROS

O ROS (*Robot Operating System*) é um *framework* amplamente utilizado na robótica, sua estrutura é definida por “nós” que se comunicam entre si utilizando um modelo de troca de mensagens por meio de publicação e assinatura (consumo) em locais denominados como “tópicos”(SMITH, 2020)/. Dessa forma, os nós se comunicam passando mensagens uns para os outros em um sistema de editor-assinante, onde um tópico atua como um armazenamento intermediário para alguns dos nós para publicar seu conteúdo enquanto outros se inscrevem para receber esse conteúdo por meio dos pacotes.

### 2.2.1.1 MAVROS

O pacotes MAVROS possui um drive próprio com implementação em linguagem C para comunicação por meio desses tópicos com um protocolo de comunicação

MAVLink e uma ponte UDP (*User Datagram Protocol*) MAVLink, um nó de comunicação extensível com *proxy* <sup>3</sup>. Com isso ele ganha uma grande flexibilidade de uso, porém não possui integração com o Windows, e esse processo dependente de uma gama de outros pacotes e protocolos de comunicação que acabam por dificultar a utilização de forma prática (ERMAKOV, 2020), e por conta disso não foi mais utilizado neste estudo.

### 2.2.2 MAVSDK

O MAVSDK é um SDK implementado originalmente em C++, mas atualmente possui uma coleção de bibliotecas para integração com outras linguagens, como Python. Possui fácil adaptação a Pixhawk e interação com MAVLink. O fluxo dos dados também são publicados como tópicos de ponto a ponto com retransmissão (OES, 2019).

A forma de gerenciamento dos veículos no MAVSDK é por meio de uma interface de programação simples API (*Application Programming Interface*) , com acesso fácil e intuitivo, de forma que possibilita uma implementação mais rápida. O MAVSDK foi definido para esta pesquisa também por sua pré implementação nos simuladores utilizados.

## 2.3 Interface de usuário para Voo

O software de controle voo para ARPs auxilia na configuração, obtenção de dados, planejamento e execução de um plano de voo, dessa forma otimizando o tempo e reduzindo custos. A estação de controle de solo, em inglês GCS (*Ground Control Station*) é um centro de controle de voo terrestre que fornece essas funções, isso permite um maior monitoramento das ações que estão sendo executadas tanto em tempo real quanto offline (WILLEE, 2019).

Dessa forma uma GCS deve conter: planejamento de missões, exibição dos dados dos equipamentos, cronograma de telemetria, terminal de dados no solo, equipamento de comunicação e dispositivo de proteção do mesmo, possibilidade

---

<sup>3</sup> Aplicativo de servidor de recursos que atua como intermediário entre um cliente (solicitante) e o servidor (fornecedor).

de ajuste de PID, e integração com outros sensores e equipamentos. Tudo isso utilizando uma interface gráfica, do inglês GUI (*Graphical User Interface*) (HAQUE; KORMOKAR; ZAMAN, 2017). Os GCSs analisados são descritos abaixo e apresentados na figura 8.

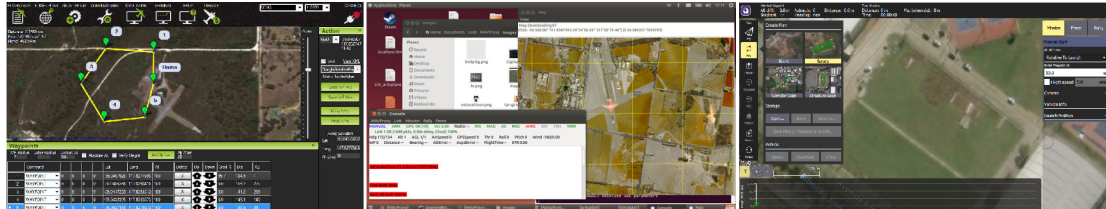


Figura 8 – Estação de controle de solo Mission Planner à esquerda, MAVProxy no centro e QGroundControl à direita. Fonte: (WILLEE, 2019)

### 2.3.1 MAVProxy

MAVProxy é uma GCS, escrita em Python, frequentemente usado por desenvolvedores que usam uma interface de linha de comando com módulos gráficos apenas para edição de mapas e missões. Sua comunicação utiliza MAVROS, um pacote pertence ao *framework* ROS, ele atua como um relé de comunicação capaz de conectar o software GCS (TRIDGELL, 2019).

Dados de voo relevantes, como velocidade do vento, taxa de ascensão ou nível da bateria podem ser monitorados em registros de arquivos, sua GUI simplificada pode ser adaptada a hardwares menos robustos, porém torna o processo difícil para usuários menos experientes, outra questão é a baixa taxa de atualização e suporte, e por conta disso foi desconsiderada do estudo.

### 2.3.2 Mission Planner

O Mission Planner, utiliza mapas do Google e Bing. Possui uma interface com um simulador de voo de computador, para criar um simulador de ARP integrado, em inglês SITL (*Software In The Loop*) e integração com todos os veículos ArduPilot. É um dos GCS mais utilizados e possui interface para Windows, Mac OSX e

Linux. Porém foi desconsiderado do estudo devido a pouca afinidade, versatilidade e tempo de uso, e também, por possuir uma grande lista de erros ainda pendentes de solução (OBORNE, 2019).

### 2.3.3 QGroundControl

O QGroundControl é uma GCS compatível com Ardupilot e PixHawk, é de código aberto e possui integração com vários sistemas como Windows, Mac OSX, Linux, iOS e Android. Permite a completa modificações no controlador PID inclusive com o ARP no ar. Possibilita a integração de múltiplos ARPs para configuração ou utilização em paralelo de modo facilitado.

Esta GCS foi definida para este estudo devido à versatilidade quanto a interface, sendo a mais intuitiva das analisadas. Dessa forma, atende tanto usuários iniciantes oferecendo uma ótima GUI, e também, para os mais experientes permite modificação total dos parâmetros e do software como um todo (EBEID; SKRIVER; JIN, 2017).

## 2.4 Simulação

Prosseguindo com os materiais utilizados, a simulação tem um papel fundamental durante a pesquisa, ela permite executar os testes em diversos modelos de algoritmos que seriam realizados em campo, contudo com a segurança e rapidez. Para realização de testes em ambientes reais, é de grande importância passar previamente por uma simulação, que contenha a maior similaridade possível com elementos da realidade. No caso do uso de um ARP real em testes no ambiente real, caso haja alguma intercorrência, um acidente por exemplo, os custos aumentam consideravelmente.

Os simuladores de ambiente considerados para análise possuem código aberto, compatibilidade com as ferramentas previamente definidas, proximidade com a física do mundo real no que diz respeito a posição do solo e objetos, forças angulares e lineares aplicadas, iluminação, colisões e proximidade visual. Além do mais, é necessário que todas essas características sejam controláveis e auditáveis para coleta e posterior análise dos dados.

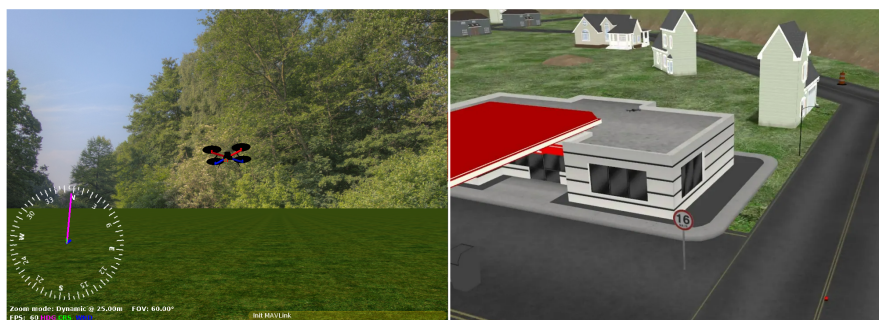


Figura 9 – Simuladores JMAVSim à esquerda e Gazebo à direita. Fonte: (TEAM, 2019)

### 2.4.1 JMAVSim

O jMAVSim é um simulador simples, leve e de fácil configuração desenvolvido pela própria PX4. Possui um mundo simulado pequeno, ideal para testes básicos como decolagem, pouso, comandos, e resposta em condições de falha (TEAM, 2019).

O ambiente gráfico é criado por Java3d. Contudo foi desconsiderado do estudo por conta das operações que necessitam de objetos complexos e câmeras, onde o mesmo não é indicado devido a sua grande limitação de características nas inserções de objetos e sensores no mundo simulado, impossibilitando uma inspeção visual detalhada (HENTATI et al., 2018).

### 2.4.2 Gazebo

Gazebo é um ambiente de simulação desenvolvido inicialmente na University of Southern California, utiliza gráficos Ogre3D, e possui uma variedade de modelos de recursos online, capacidade de importar malhas de outros softwares de modelagem de forma detalhada, permite a implementação de novos sensores para esboçar uma característica de ambiente do mundo real. Apesar disso, não é prático utilizar essas modificações de ambiente, sendo necessário instalar e ajustar um conjunto de arquivos implementados em um formato XML (*eXtensible Markup Language*) personalizado chamado de SDF (*Simulation Description Format*) (HENTATI et al., 2018).

Está disponível apenas no Linux, e permite a implementação de vários robôs

no mesmo ambiente com tarefas variadas, contudo apesar das características que se aproximam do mundo real, não possui uma interface realista, dessa forma os dados de imagem extraídos são muitos divergentes do mundo real. Apesar das questões positivas levantadas e do mapa inicial desenvolvido para a pesquisa, esse simulador foi desconsiderado do estudo já que seus gráficos ainda que mais simplificados, exigem um computador com mais capacidade de processamento e memória (KOENIG, 2019).

### 2.4.3 Airsim

O AirSim foi definido para uso neste estudo, é um simulador foto realístico desenvolvido exclusivamente para desenvolvimento de aprendizado de máquina, está disponível para Windows e Linux, e foi desenvolvido pela equipe pesquisa da Microsoft. O gráfico é baseado na Unreal Engine 4, é previamente interligado com interface do Mavlink, sendo possível executar SITL e HITL (*Hardware in the Loop Simulation*) (SHAH et al., 2018).

Possui APIs próprias com design modular para recuperação dados e para controle de veículos de forma facilitada e independente com diversos sensores como câmeras monoculares e de profundidade. Apesar de limitadas as APIs são suficientes para a maioria das simulações de aprendizado de máquina, e acessíveis através de várias linguagens de programação, incluindo C++, C#, Python e Java (HENTATI et al., 2018).

A interface Unreal Engine oferece simulações física e visualmente realistas de modo acessível, como colisões de objetos, chuva, direção do vento, neblina entre outros, e diversos dados dos sensores obtidos no momento da execução como apresentando na Figura 10. Também possui uma grande comunidade para discussão e loja exclusiva para incorporação de objetos, projetados para ser extensível a novos tipos de estruturas, veículos, plataformas de hardware e protocolos de software (MICROSOFT, 2019).

Apesar de exigir uma GPU (*Graphics Processing Unit*)<sup>4</sup> de alta performance (cerca de 8Gb) em suas documentações, computadores mais recentes com ao me-

---

<sup>4</sup> Microprocessador especializado em processar elementos visuais em computadores com maior eficiência que o CPU (*Central Processing Unit*)



Figura 10 – Mundo 'Vizinhança' do Airsim, com os dados dos sensores (direita) e opções de alteração de clima (esquerda). Fonte: (MICROSOFT, 2019)

nos 4GB de RAM (*Random Access Memory*), processador principal *QuadCore* e 512MB de GPU são suficientes para executar a simulação de maneira satisfatória.

## 2.5 RSL

A metodologia escolhida para esta parte do estudo, onde são abordados trabalhos relacionados, segue o protocolo detalhado originalmente proposto por Kitchenham (KITCHENHAM, 2004) apresentado na Figura 11.

Esse modelo possui um processo de pesquisa metodologicamente bem definido a ser aplicado que identifica, analisa e interpreta as evidências disponíveis relacionadas às questões de pesquisa. O método empregado oferece mais comprometimento e qualidade científica em comparação com uma pesquisa bibliográfica comum (FREITAS, 2019; KITCHENHAM; CHARTERS, 2007).

As evidências são materializadas em artigos primários publicados recuperados de fontes de pesquisa selecionadas. A recuperação dessas obras acontece por meio de um texto de pesquisa, em inglês *SS (Search String)*, que resume as questões a serem pesquisadas. Assim, foram definidos critérios que garantam o equilíbrio, a viabilidade e a acessibilidade das informações para a execução do estudo por meio



Figura 11 – Etapas para o desenvolvimento de uma revisão sistemática da literatura. Fonte: Próprio Autor

desta SS (KITCHENHAM; CHARTERS, 2007), os critérios de condução são:

- Possuindo mecanismo de busca que permite a utilização de expressões lógicas ou mecanismo equivalente, retornando o esperado da SS.
- Incluir em seu acervo com relevância no meio acadêmico nas áreas de computação e engenharia.
- Os motores de busca devem permitir a busca no texto integral das publicações.
- Data de publicação própria entre 1<sup>o</sup> de janeiro de 2014 e 1<sup>o</sup> de junho de 2019 (época da realização).
- Deve ter a característica de filtrar apenas o artigo.
- A publicação deve estar disponível no idioma inglês.

A pesquisa foi realizada a partir de quatro bibliotecas digitais, são elas: ACM (*Association for Computing Machinery*) ([dl.acm.org](http://dl.acm.org)), IEEE (*Institute of Electrical and Electronics Engineers*) ([www.ieee.org](http://www.ieee.org)), Scopus ([www.scopus.com](http://www.scopus.com)) e Compendex ([www.engineeringvillage.com](http://www.engineeringvillage.com)). Essas bases foram escolhidas pelo experimento realizado por (DYBA; DINGSOYR; HANSEN, 2007). Conforme argumentado

acima, algumas bases de dados atualmente conhecidas como Google Scholar, PubMed e bibliotecas de dissertações e teses não foram consideradas, por não atenderem aos critérios de direcionamento citados. No entanto, sabe-se que grande parte dos artigos contidos nessas plataformas também estão disponíveis nas bases selecionadas, por serem da área de estudo e de alta relevância acadêmica.

Para auxiliar no desenvolvimento deste estudo, foi utilizada a ferramenta de Parsifal (FREITAS, 2019). Possui as principais etapas necessárias para o desenvolvimento do RSL, além de possibilitar a colaboração online dos autores e a geração de relatórios analíticos. Assim, é produzido um extenso material, com todas as informações necessárias dos artigos, para sintetizar o resultado neste trabalho.

## 2.6 Legislação

Antes de prosseguir com os métodos computacionais utilizados no trabalho, para realização de um voo seguro é necessário atender as regras em vigor determinadas. A agência brasileira de controle do espaço aéreo responsável pela regulamentação dos ARPs é o DECEA, sob a administração do Ministério da Defesa. No entanto, o registro e o controle das aeronaves são de responsabilidade da ANAC. Embora as inspeções de linhas de transmissão de energia utilizando ARPs sejam permitidas, elas apresentam algumas limitações, como proibição de utilização em áreas residenciais, proximidade abaixo de 5 metros, limite de 2,5 km do local de decolagem e a carga útil máxima é de 2kg para aeronaves do tipo micro em modo VLOS (*Visual Line-Of-Sight*) (AQUINO, 2019).

As aeronaves utilizadas neste estudo, de propriedade da Unifei (Universidade Federal de Itajubá) possuem identificação correspondente no SISANT (Sistema de Aeronaves Não Tripuladas) juntamente a ANAC. Esse é um registro obrigatório que define cada aeronave em funcionamento e o seu propósito (ASCOM, 2017). Além disso, é necessário uma autorização de uso para operar o ARP remotamente por intermédio de aparelhos de radio frequência, essa autorização é concedida pela Anatel. Isso permite o envio e recebimento de dados por meio de controle remoto ou telemetria de frequências específicas, contudo em grande parte os equipamentos usados já possuem uma autorização emitida pelo próprio fabricante.

Cada voo a ser realizado deve ser informado ao sistema SARPAS (Solicitação de Acesso de Aeronaves Remotamente Pilotadas), sob responsabilidade do DECEA, onde são detalhados: a localização geográfica da partida, a área de abrangência, a aeronave utilizada, as pessoas envolvidas e seus contatos (VARGAS, 2019). Existem diferentes modalidades de voo, em geral inspeções em estruturas podem ser realizadas fora do princípio da sombra, que compreende as proximidades da estrutura a ser inspecionada. A altura de voo em relação ao solo é outro critério determinante na modalidade de voo.

O atendimento a todos esses órgãos garante a lisura do procedimento do voo, e também, que os demais operadores da aérea que estejam atuando com aeronaves tripuladas ou não, sejam notificados.

## **2.7 Visão Computacional**

A visão computacional é tecnologia para a construção de sistemas artificiais que obtém informação de imagens de um ambiente e as processa com uma finalidade. A fonte do processo são quaisquer dados multidimensionais, geralmente são representações digitais compostos por pixels com valores finitos e discretos (PETROU; PETROU, 2010). Nas subseções abaixo, serão descritos os métodos que foram empregados no estudo para realizar a solução no rastreamento de linhas de transmissão de energia.

### **2.7.1 Pré-Processamento**

Com a intenção de obter maior desempenho e melhorar a análise computacional, um pré processamento digital da imagem pode ser realizada para evidenciar características visuais de acordo com o objetivo da solução, como por exemplo, conversão de cores; aumento contraste; melhorar foco; reduzir ruído; eliminar distorções; extrair elementos de interesse; transformação da orientação, entre outras.

#### **2.7.1.1 Redimensionamento**

Redimensionar uma imagem proporciona o ajuste de suas dimensões mantendo a proporção da imagem original preservada, pode ser necessário tanto para se

adequar a um modelo proposto, quanto para otimização do desempenho de processamento (LIN et al., 2019). O redimensionamento para menor foi utilizado neste estudo.

### **2.7.1.2 Componentes**

O método de coloração converte um espaço de cor de uma imagem em outro, também pode ser utilizado juntamente com uma conversão para alteração dos componentes (canais) de cores de uma imagem. Em geral é utilizado para otimização dos modelos, já que trabalhando com menos canais há um aumento significativo do desempenho (ENDO et al., 2016). A descoloração com a redução de canais foi utilizada neste estudo.

### **2.7.1.3 Ruído**

A presença de ruído é prejudicial para qualquer solução computacional, contudo sua raridade também pode ser prejudicial para o sistema inteligente. Dessa forma, a redução de ruído visa remover detalhes que não são relevantes proporcionando um aspecto de sARPização da imagem (ENDO et al., 2016). Contudo o ruído também tem a capacidade de generalizar uma solução, viabilizando a redução da raridade de modo que o modelo saiba lidar com essas determinadas situações (HULSE; KHOSHGOFTAAR, 2009). Ambas situações foram utilizadas nesse estudo.

## **2.7.2 Detecção de bordas**

Um processo de segmentação tem como objetivo final a extração de atributos para análise, dessa forma, uma imagem recebida é subdividida em conjunto de pixels que representam regiões constituintes de similaridade ou objetos de interesse, com referência a alguma característica ou propriedade computacional, tais como cor, intensidade, textura ou continuidade (MILANO; HONORATO, 2014).

A detecção de bordas, do inglês ED (*Edge Detector*), em uma imagem define e armazena os pixels que possuem alguma descontinuação ou mudança brusca em relação aos seus vizinhos. John F. Canny em (CANNY, 1986) propôs um algoritmo, que leva seu nome, para detecção de todas as bordas de uma imagem. A segmentação nesse algoritmo facilita a localização de formas retas nas imagens.

Seu funcionamento completo consiste em aplicar um filtro gaussiano para suavizar a imagem e remover o ruído, encontrar os gradientes de intensidade da imagem e determinar bordas potenciais ao aplicar Sobel duplo <sup>5</sup>, verificar se o pixel faz parte de uma borda “forte” suprimindo todas as outras bordas que são fracas e não conectadas à bordas com forte relação de valores entre os pixels vizinhos (BRADSKI; KAEHLER, 2008). A figura 12 apresenta a aplicação do filtro gradiente no algoritmo Canny.

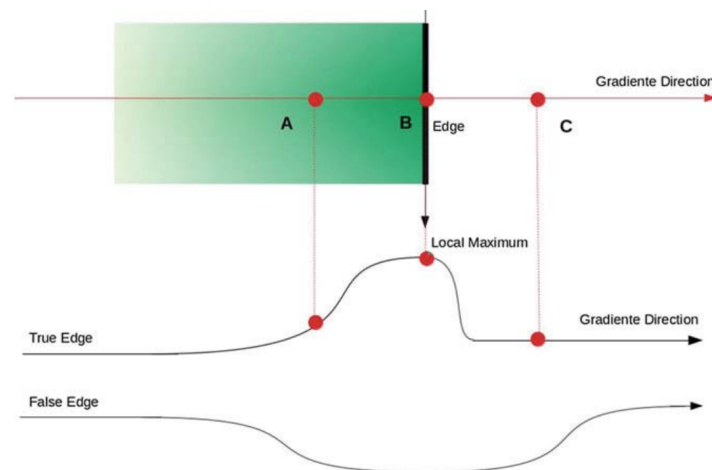


Figura 12 – Filtro gradiente Canny. Fonte: (FILHO et al., 2020)

### 2.7.2.1 Detectores de borda

Neste trabalho foram utilizados três algoritmos no estudo de ED em imagens. Um exemplo de aplicação desses algoritmos são apresentados na figura 13.

A TH (Transformada de Hough) é um algoritmo que recebe uma imagem binária com candidatos que podem pertencer a uma determinada reta, dessa forma é possível encontrar contornos que possam ser parametrizadas por uma equação bem definida como retas (PIVETTA; MANTOVANI; ZOTTIS, 2017). Representando a distância perpendicular da reta até a origem do sistema de coordenadas da imagem e sua inclinação, a partir de uma quantidade pré definida de intersecções de

<sup>5</sup> Operador utilizado para determinar bordas potenciais por meio do gradiente da função de intensidade, obtendo a norma ou vetor de gradiente horizontal e vertical.

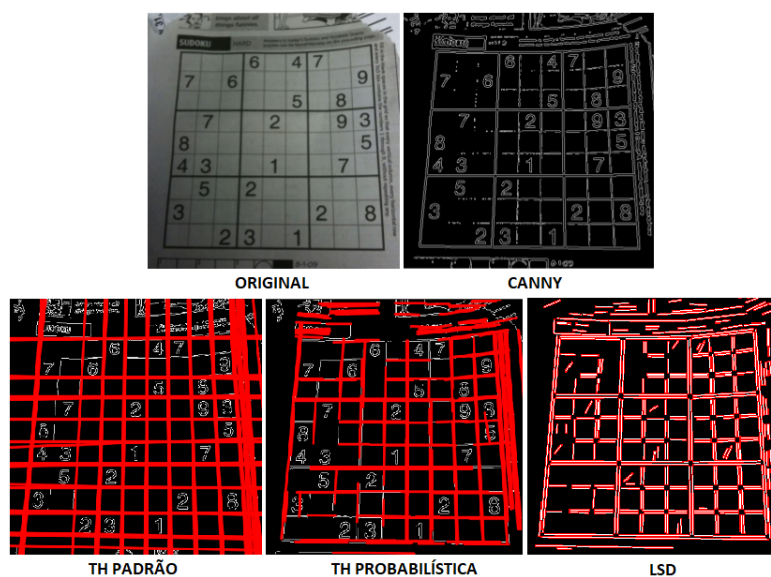


Figura 13 – Exemplificação da Transformada da linha Hough Padrão, Probabilística e *Line Segment Detector*. Fonte: (HUAMÁN, 2019) Adaptada

curvas entre dois pontos diferentes se cruzando em um plano, é possível concluir que ambos os pontos pertencem a uma mesma linha (HUAMÁN, 2019).

A THP (Transformada de Hough Probabilístico) é mais eficiente em soluções que necessitam contemplar a imagem como um todo, isso porque sua saída são os extremos das linhas detectadas em relação as dimensões da imagem (KIRYATI; ELDAR; BRUCKSTEIN, 1991).

O detector de segmento de linha, do inglês LSD (*Line Segment Detector*), foi projetado para trabalhar com poucos parâmetros externos, ajustando automaticamente o número de detecções falsas. Essa técnica usa uma abordagem de validação ao contrário, conforme a teoria de Desolneux, Moisan e Morel <sup>6</sup>. O LSD tem um tempo de execução proporcional ao número de pixels da imagem, ou seja, sua detecção é de tempo linear para retornar resultados precisos ao subpixel (GIOI et al., 2012).

<sup>6</sup> Baseado na teoria do Gestalt e na geometria estocástica, onde um estudo para compreender os padrões espaciais aleatórios é preciso antes de compreender o todo em uma imagem.

### 2.7.3 Aprendizado de máquina

O aprendizado de máquina, ou em inglês ML (*Machine Learning*) é um estudo pertencente ao campo da inteligência computacional, onde são desenvolvidos algoritmos capazes de receber entradas de dados de treinamento a partir de amostras, melhorarem automaticamente aprendendo com seus erros, realizar ajustes estatísticos e fazer previsões sobre novos dados para uma determinada tomada de decisão em uma solução de modo adaptável, sem efetivamente programar cada situação (KRUSE et al., 2011).

O conceito iniciou com Turing em (TURING, 1956), a partir do questionamento da capacidade dos computadores imitarem o comportamento ou as respostas de um ser humano, a ponto de não ser possível fazer distinção. Atualmente esse aprendizado automático é usado em muitas tarefas computacionais específicas, como na visão computacional, onde o desenvolvimento de algoritmos explícitos é impraticável devido a grande quantidade de variáveis que influenciam a solução.

Em geral esse método utiliza RNA (Redes Neurais Artificiais), que são inspiradas pelo sistema nervoso central dos animais, onde grandes tarefas complexas podem ser divididas entre vários micro-módulos de diversas características do mesmo problema, até a obtenção do resultado final desejado. Dessa forma as RNAs simulam o comportamento dos neurônios do cérebro, e resulta em modelos complexos capazes de realizar previsões da solução (RUSSELL; NORVIG, 2002). Esse método foi utilizado também embutido em outros métodos aqui apresentados.

No modelo Perceptron, que é o modo mais simples de RNA, cada um desses neurônios artificiais recebe uma ou mais entradas numéricas que passam por uma função estatística ( $f$ ), denominada de função de ativação, a partir de seus pesos ( $w$ ) e viés ( $b$ ) cada neurônio possui parâmetros que serão tomados durante o treinamento da rede para aproximar um modelo matemático global de um problema. A Figura 14 representa a estrutura de um neurônio em sua ativação.

A composição desses neurônios artificiais em camadas são denominadas MLP (*MultiLayer Perceptron*), onde novas camadas representaram a possibilidade de novas retas classificadoras dos elementos de entrada (KRUSE et al., 2011). Sua solução se baseia no algoritmo de correção por retropropagação do erro, e camada de saída recebe os estímulos da camada intermediária para se obter a resposta.

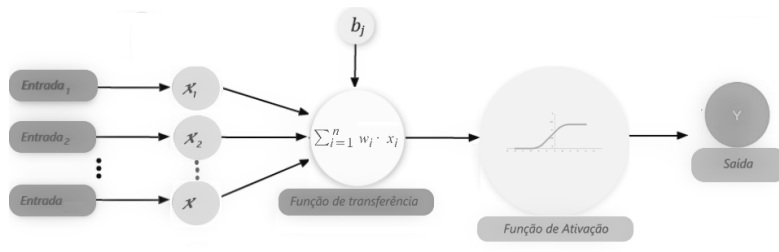


Figura 14 – Estrutura de neurônio artificial com a função de ativação. Fonte: Próprio Autor

Nesse contexto, o objetivo é minimizar a função objetiva, ou função de perda, determinada pela equação 2.1 mensurada com valores entre 0 e 1, onde é calculada a relação do somatório da diferença dos valores ( $y'$ ) e real ( $y$ ) sobre todas as instâncias ( $m$ ), o número de classe é definido por  $c$  (KRUSE et al., 2011).

$$L = \begin{cases} -\frac{1}{m} \sum_{i=1}^m y_i \cdot \log(y'_i), & \text{se } c = 2 \\ -\frac{1}{m} \sum_{i=1}^m y_i \cdot \log(y'_i), & \text{se } c > 2 \end{cases} \quad (2.1)$$

Quando há um grande volume de dados, as atualizações desses valores podem ser realizados após a composição de um lote, que é um acúmulo para treinamento da rede a cada época trabalhada. Os otimizadores são usados para minimizar a perda ou aumentar a precisão, encontrando o mínimo local da função de custo. Nesse estudo são utilizados os otimizadores *Momentum*, RMS (*Root Mean Square*) e Adam (*Adaptive Momentum Estimation*) (KRUSE et al., 2011). Suas características são apresentadas abaixo:

- *Momentum*: Uma média exponencialmente ponderada dos gradientes são calculadas para os pesos  $w$  e  $b$ , para depois usar esse gradiente para atualizar os pesos. Dessa forma, os gradientes passados suavizam a atualização superando as oscilações de gradientes com ruídos.
- RMS: As atualizações dos parâmetros em  $w$  é maior que em  $b$  nos gradientes em relação as raízes de sua respectiva média exponencial. Isso aumenta os passos maiores na direção horizontal, e assim converge mais rápido e diminui o número de iteração para convergir para o valor ideal.

- Adam: A descida do gradiente estocástico utiliza uma taxa de aprendizado para cada peso, ela é adaptada à medida que o aprendizado se desenrola. Esse algoritmo de otimização pode lidar com gradientes esparsos em problemas ruidosos, atuando bem com aprendizado profundo.

## 2.7.4 Aprendizado Profundo

Aprendizado profundo, no inglês DL (*Deep Learning*), é uma classe específica de técnicas de aprendizado de máquina, pode ser utilizada para tarefas de classificação, regressão e extração de recursos, e necessita de múltiplas camadas profundas de representação e abstração das características de modo efetivo. O conjunto de dados onde são obtidas essas características é denominado *Dataset*. Essa técnica permite a extração automática de característica embutido na "caixa preta" da RNA (RUSSELL; NORVIG, 2002).

O ReLU (*Rectified Linear Unit*) é a função de ativação simples, onde é retornado zero se receber qualquer entrada negativa, mas para qualquer valor positivo  $x$  ela retorna esse mesmo valor 2.2, sua derivada monotônicas permite que seu modelo leve em consideração não linearidades e boas interações. A função de ativação mais usada no mundo desde 2017, pois apresenta ótimos resultados em quase todas as redes neurais de aprendizado profundo (BRITO et al., 2019).

$$f(x) = \max(0, x) \quad (2.2)$$

A seguir serão apresentadas variações de algoritmos DL, que permitem um melhor resultado na aplicação a visão computacional.

### 2.7.4.1 Rede Neural Convolutacional

Uma rede neural convolutacional, em inglês CNN (*Convolutional Neural Network*), é uma variação da MLP biologicamente inspirada na tarefa de processamento de imagens, dessa forma, várias classes de camadas diferentes de redes neurais profundas aplicam filtros em vários níveis para gerenciar, extrair e classificar informações visuais de uma fonte digital (BRITO et al., 2019). Sua arquitetura base é apresentada na figura 15.

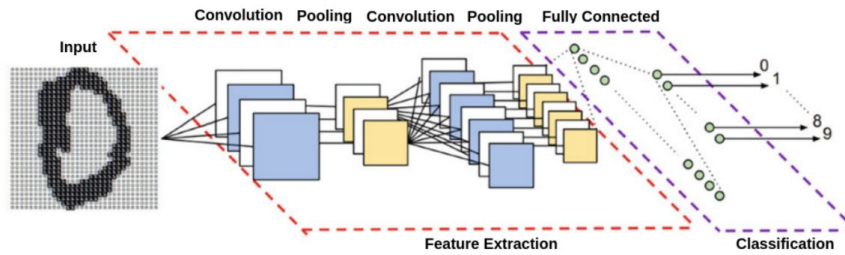


Figura 15 – Exemplo de arquitetura de uma CNN. Fonte: (BRITO et al., 2019)

A arquitetura CNN pode ter diversas combinações de camadas de acordo com proposta de solução. As camadas iniciais e intermediárias geralmente tem uma função de convolução, onde os neurônios são mapeados de acordo com um núcleo subdividido e uma operação em um grupo de *pixels* dentro da imagem. A imagem é então filtrada e classificada de acordo com os recursos importantes e condensado para o processamento do conjunto atual e aumentar a velocidade da rede. Por fim, as camadas totalmente conectadas proporcionam a saída da rede (ZEILER; FERGUS, 2014). Esse método foi utilizado neste estudo por razão da necessidade de detecção de características visuais utilizando uma câmera digital.

### 2.7.5 Detector de múltiplas caixas de passagem única

Uma outra arquitetura utilizada na visão computacional é o detector de múltiplas caixas de passagem única, no inglês SSD (*Single Shot MultiBox Detector*), foi apresentado em 2016 utilizando uma arquitetura CNN baseada na VGG (*Visual Geometry Group*)-16, uma rede profunda de 16 camadas usado pelo Grupo de Geometria Visual da Universidade de Oxford (PINTO et al., 2019).

Contudo a SSD descarta as camadas totalmente conectadas e insere novas camadas intermediárias, permitindo uma diminuição progressiva do tamanho da entrada e extração de recursos em múltiplas escalas, isso resulta em detecção de objetos em tempo real (PINTO et al., 2019). O SSD pertence à categoria de detectores de disparo único, que removem o estágio de geração de proposta presente nos detectores de dois estágios e prevê as pontuações das classes próximas às coordenadas da caixa delimitadora da imagem. Seu núcleo foi construído em torno de

uma técnica chamada *MultiBox*, que é um método para propostas de coordenadas de caixa delimitadora agnósticas de classe (LIU et al., 2016).

Uma rede de pirâmide de características, no inglês FPN (*Feature Pyramid Network*), pode ser agregado a rede SSD, se trata de um extrator de características que tira uma imagem em escala única de um tamanho arbitrário como entrada, e saídas de tamanho proporcional de mapas de recursos em vários níveis, de forma totalmente convolucional independente da CNN, utilizando a saída de ativações de recursos pelo último bloco residual de cada etapa.

A construção de pirâmides de recursos dentro de redes convolucionais profundas para serem usadas em tarefas como detecção de objetos, gera uma semântica mais forte em cada nível. Ele substitui o extrator de recursos de detectores como o RCNN (*Region Based Convolutional Neural Networks*) rápido. A FPN introduz um pequeno custo extra para essas novas camadas, mas tem um peso mais leve na implementação geral (LIN et al., 2017).

## 2.7.6 Aprendizado por Reforço profundo

O aprendizado de reforço, no inglês RL (*Reinforcement Learning*), é uma sub-área do aprendizado de máquina, seu objetivo principal é atuar com um agente em um ambiente da forma que retorne a maior recompensa cumulativa possível. O aprendizado é realizado por meio de exploração e conhecimento adquirido, sem a necessidade de rotulagem de dados (SUTTON; BARTO, 2018).

O ambiente é um espaço  $\mathbf{S}$  limitado, estático ou dinâmico, onde o agente pode atuar e ser avaliado. O agente é definido como o tomador de decisões  $\mathbf{E}$  em um ambiente, a cada ação  $\mathbf{A}$  tomada é recebida uma recompensa  $\mathbf{R}$ . Todas essas informações são armazenadas em memória, para processamento dos lotes pelo agente, que definirá novas políticas de valores  $\mathbf{Q}$ . Esse processo é apresentado na figura 16.

Nesse processo de aprendizagem o ambiente é representado na forma de Processo de Decisão de Markov <sup>7</sup>, em inglês MDP (*Markov Decision Process*). No MDP uma matriz de transição probabilística  $\mathbf{P}$ , também chamada de política  $\boldsymbol{\pi}$ , é

---

<sup>7</sup> Propriedade que presume que os estados futuros não dependem de eventos que ocorreram no passado, dependendo apenas do estado presente

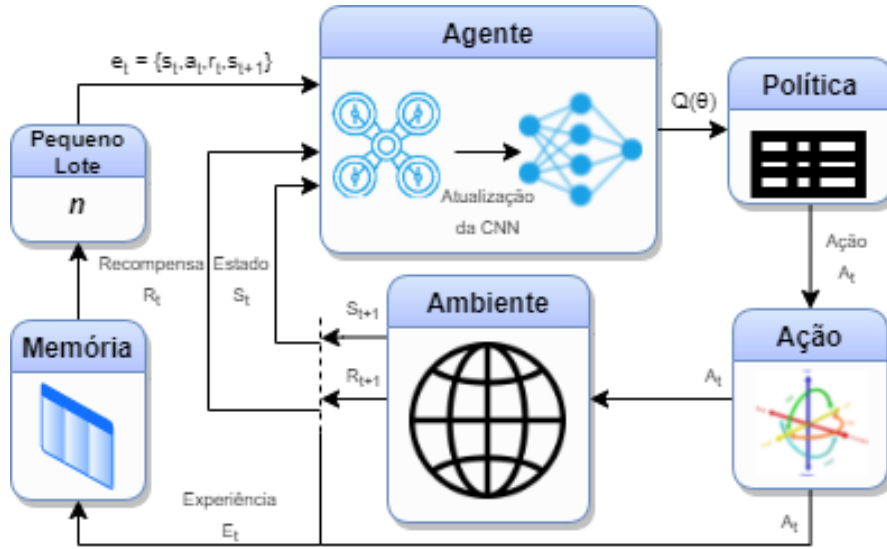


Figura 16 – Exemplo de arquitetura de aprendizado DRL. Fonte: Próprio autor

utilizada para definir o futuro sucessor ( $s'$ ) a partir do atual  $s$ , conforme apresentado na equação 2.3.

$$P[S_t + 1 | S_t] = P[S_t + 1 | S_1, S_2 \dots S_t] \quad (2.3)$$

Essa política  $\pi$  depende da recompensa esperada em cada estado, que é obtida pela soma da recompensa ponderada por  $\gamma$  de todos os estados passados por esse agente 2.4.

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k} + 1 \quad (2.4)$$

Dessa forma, é possível obter o valor da função 2.5 esperada para transição de estados de acordo com a probabilidade de uma recompensa maior futura. E o processo MDP, é definido pela equação do conjunto finito de possíveis ações a se tomar em um estado 2.6.

$$v(s) = E[R_t + 1 + \gamma v(S_t + 1) | S_t = s] \quad (2.5)$$

$$R_s^a = E[R_t + 1 | S_t = s, A_t = a] \quad (2.6)$$

Para utilização em espaços de estados de grandes e complexas dimensões, o RL foi combinado com DL. Denominado DRL (*Deep Reinforcement Learning*), ele possui a capacidade de identificar recursos de alto nível diretamente dos dados obtidos e atuar na política de valores, determinando qual ação tomar em um determinado estado como mostra a equação 2.7.

$$q_\pi(s, a) = E_\pi[G_t | S_t = s, A_t = a] \quad (2.7)$$

O algoritmo determina a qualidade de uma determinada ação em qualquer local do ambiente utilizando uma política ideal, e por conta disso  $\gamma$  é importante, já que ele define o quanto valiosa é a recompensa no futuro em comparação com o próximo estado, esse processo é representado pela equação 2.8 nomeada de idealidade Bellman (SUTTON; BARTO, 2018).

$$q_*(s, a) = R_s^a + \gamma \sum_{s' \in SP_s} \max_{a'} q_*(s', a') \quad (2.8)$$

O processo de aprendizagem se estende por muitos episódios para se obter uma compreensão melhor do ambiente, e ao decorrer do treinamento o agente diminui o ritmo de exploração do ambiente e se baseia mais na política atual, isso é representado por  $\epsilon$ . Esse método foi utilizado no estudo, e possui vantagens em relação aos demais por alcançar bons resultados de longo prazo, auto correção de erros de treinamento, e possibilidade de solução diferenciada dos modelos regulares.

### 2.7.6.1 Algoritmos de Aprendizado por Reforço

São utilizados três algoritmos DRL de alto desempenho neste estudo, desenvolvidos a partir de um conjunto de implementações aprimoradas das estruturas de linha de base da OpenAI (RAFFIN et al., 2020), são eles:

- DQN (*Deep Q Learning*): O objetivo é treinar uma política Q-learning  $Q^*$  maximizando a recompensa acumulada e descontada do retorno  $\gamma$ , além disso uma técnica de *replay* da experiência é utilizada baseada na amostra eficiente

aleatória de ações anteriores em vez da ação mais recente. É o método mais intuitivo, de convergência lenta, mas eficiência é alta.

- A2C (*Advantage Actor Critic*): O Crítico estima a função de valor  $Q$  e  $V$ , o Ator atualiza a distribuição da política na direção sugerida pelo Crítico, como com gradientes de políticas. A2C é um método fora da política, e usa estimativas de vantagem para calcular a proposta de valor para cada par de estado de ação adiantado obtendo uma convergência mais rápida.
- PPO (*Proximal Policy Optimization*): O equilíbrio entre complexidade da amostra e facilidade de ajuste é explorado, por meio do cálculo da atualização minimização da função de custo em cada etapa, garantindo que o desvio da política anterior seja relativamente pequeno. É um algoritmo de política, mais simples de implementar podendo atuar com penalidade na divergência dos valores de  $Q$  ou *clip* (recortes) na função objetiva  $V$ .

Serão apresentados a seguir ferramentas utilizadas em conjunto com os algoritmos na visão computacional.

## 2.7.7 Frameworks para Visão Computacional

Algoritmos de aprendizado de máquina são extensos e exigem diversas estruturas matemáticas complexas para sua execução, por conta disso diversos kits de ferramentas, em inglês *Framework*, que são conjuntos de metas de aprendizagem que fornecem a estrutura para todos os padrões de aprendizagem acadêmica, foram desenvolvidos pela comunidade para auxiliar na programação dos experimentos.

Todos os *frameworks* utilizados possuem algum tipo de facilidade de implementação, possibilidade de uso tanto de CPU quanto GPU no treinamento e execução, documentação e comunidade para ajuda e discussão, código aberto, adesão aos algoritmos abordados nesse estudo.

### 2.7.7.1 OpenCV

OpenCV (*Open Source Computer Vision Library*) é um *framework* desenvolvido principalmente para a visão computacional 2D e 3D em tempo real, ou seja, com

alta taxa de imagens por segundo, do inglês FPS (*Frames Per Second*). Foi desenvolvido pela Intel, possui chamadas de APIs para C++, Java e Python (BRADSKI; KAEHLER, 2008). Possui um uso abrangente, com mais de 80.000 repositórios relacionados no Github (WERNER; HYETT; CHACON, 2019), além de livros, e grandes comunidades dedicadas exclusivamente a esse *framework*.

Também possui algoritmos para segmentação, reconhecimento, rastreamento, realidade aumentada, inteligência computacional, entre outras ferramentas (GARAGE et al., 2009). Neste estudo, seu uso foi restrito ao pré-processamento de imagens, segmentação e detecção de bordas.

### 2.7.7.2 CNTK

CNTK (*Microsoft Cognitive Toolkit*) é um *framework* desenvolvido pela Microsoft, disponível para Windows e Linux com modelos ONNX (*Open Neural Network Exchange*), que descreve suas redes neurais como uma série de passos computacionais através de um gráfico direcionado. Sua estrutura permite combinações facilitadas de diversos modelos populares (TEAM, 2020).

Foi considerado inicialmente nesse trabalho por conta da sua ferramentas em alto nível e integração pré-moldada com o simulador Airsim. Contudo foi observado de forma empírica uma deficiência no desempenho nas máquinas instaladas, o projeto principal está descontinuado, e detém uma comunidade pequena com poucos usuários ativos, e por conta disso não foi usado nos experimentos (TEAM, 2020).

### 2.7.7.3 PyTorch

PyTorch é um *framework* recente baseado em Python e Torch, que são tensores semelhantes ao NumPy Arrays, foi desenvolvida pelo laboratório de pesquisas em inteligência artificial do Facebook. Possui recursos de alto nível para Computação tensor com forte aceleração de GPU Redes neurais profundas construídas em um sistema de auto gradiente baseado em fita (FAIR, 2020).

Esse *framework* foi utilizado na implementação dos algoritmos de DRL devido sua recente rápida ascensão em repositórios no GitHub e nas citações científicas de ML, o que aumenta a discussão e contribuição na comunidade (STEVENS;

ANTIGA; VIEHMANN, 2020). Também possui alta capacidade de depuração, devido ao seu modo *verbose* mais completo (KOVALEV; KALINOVSKY; KOVALEV, 2016).

#### **2.7.7.4 TensorFlow**

TensorFlow é o *framework* mais relevante no ML (STEVENS; ANTIGA; VIEHMANN, 2020) foi desenvolvido pelo Google que recentemente lançou o TensorFlow 2.0. Seus cálculos são expressos como grafos de fluxo de dados mantendo um estado, e suas chamadas podem ser realizadas por meio de APIs Python e C++ (ABADI et al., 2015).

#### **2.7.7.5 Keras**

Keras é uma interface de alto nível para biblioteca TensorFlow 2, foi desenvolvido inicialmente para sistema operacional de robôs inteligentes neuroelétricos. Essa ferramenta fornece abstrações essenciais e blocos de construção para o desenvolvimento e envio de soluções de aprendizado de máquina com alta velocidade de iteração e possibilita execuções de alto desempenho, especialmente para grandes servidores (CHOLLET, 2020).

Foi projetado com uma interface amigável para permitir experimentações rápidas e práticas de aprendizado de máquina de forma modular e extensível, e por essa questão necessita de menos número de linhas para uma implementação de algoritmo. Juntos são os *frameworks* mais populares e com melhor desempenho considerando acurácia, tempo de treinamento e tempo de teste (KOVALEV; KALINOVSKY; KOVALEV, 2016). O Keras foi definido para uso neste estudo no desenvolvimento dos algoritmos de DL.

#### **2.7.7.6 TensorBoard**

Para facilitar o acompanhamento de um treinamento do modelo, foi definido para esse estudo uma plataforma baseada na web, o TensorBoard, indicado para uso juntamente com o TensorFlow (KUDLUR et al., 2020). A interface dessa ferramenta é apresentada na figura 17

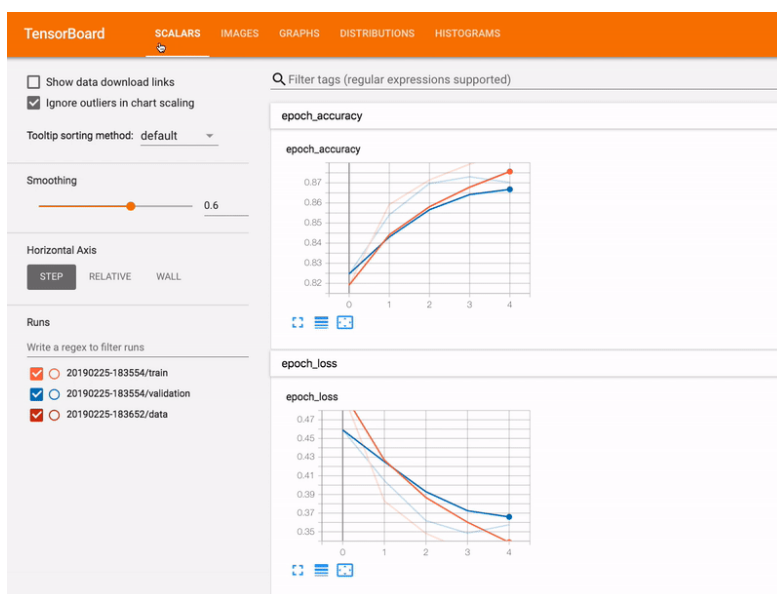


Figura 17 – Interface do TensorBoard. Fonte: (KUDLUR et al., 2020)

Se trata de uma interface com uma API, que possibilita o registro e visualizações em tempo real de gráficos de perda e acurácia de um treinamento, visualizar o gráfico da rede, avaliar o resultado do teste com um conjunto de dados visuais, entre outras funcionalidades.

### 2.7.8 Estágio no Exterior

Com o propósito de investigar soluções no estado da arte de visão computacional, foi firmada uma parceria da Unifei com a MDH (*Mälardalens högskola*), da Suécia, sediado em Västerås. O projeto também é uma importante contribuição para um projeto de pesquisa na MDH financiado pela Agência Sueca de Inovação Vinnova. O tema escolhido foi atividades de busca e salvamento, do inglês SAR (*Search And Rescue*), a busca e identificação de pessoas perdidas em situação de emergência é uma atividade muito importante, é realizada para prestar assistência a vidas humanas em perigo. Os VANTs atuam diretamente nessa atividade para maior capacidade na cobertura da área em um menor tempo de operação (POTEYEVA et al., 2007).

Pesquisadores brasileiros e suecos vêm desenvolvendo métodos, técnicas e ferramentas de interpretação de imagens de solo obtidas por sensores embutidos em VANT, com o objetivo de auxiliar e complementar os esforços de pesquisadores humanos. Motivado principalmente pela redução de custos e eficiência, proporcionado pela versatilidade no processo de fotos capturadas em tempo real em posições estratégicas, em comparação com o alto custo e por vezes de baixa eficiência, quando realizado em áreas com poucos profissionais ou fotos de satélites (FILHO et al., 2020).

Neste trabalho conjunto, alternativas de reconhecimento de imagem de baixo processamento para uso em VANT foram exploradas, a fim de enfrentar desafios como localização em grandes áreas, alvo pequeno e orientação. Os resultados preliminares mostram que o método proposto é viável para ações de busca e resgate de pessoas. Dessa forma, já foram produzidos um conjunto de dados, uma base inicial para treinamento, uma arquitetura para experimentos e, principalmente, um conceito teórico para a realização de busca e resgate de pessoas utilizando imagens de VANT.

A partir dos resultados provisórios, é possível observar a viabilidade do procedimento adotado no estudo. Até agora, realizar pré-processamento, detectar utilizando aprendizado profundo e seguir utilizando algoritmos de rastreamento produziram os melhores resultados para o contexto de pesquisa. Um resumo do estudo realizado com mais informações está disponível no Apêndice C seção 8.

### 3 Desenvolvimento

Neste capítulo serão apresentados todos os detalhes referente a proposta de desenvolvimento do modelo do estudo. Inicialmente serão abordados os aspectos do problema de rastreamento de linhas de transmissão de energia, em seguida será esmiuçada a arquitetura da solução e de sua implementação, e os modelos dos métodos de visão computacional.

#### 3.1 Modelagem

Para uma rastreamento de linhas de transmissão de energia de forma detalhada, é necessário a obtenção de imagens o mais próximo possível da estrutura e do segmento de linha, evitando a colisão, e respeitando a legislação, permitindo assim uma inspeção detalhada dos elementos alvos. Para isso foi determinado o planejamento de trajetórias ideais para o rastreamento contínuo, a figura 18 representa essas trajetórias.

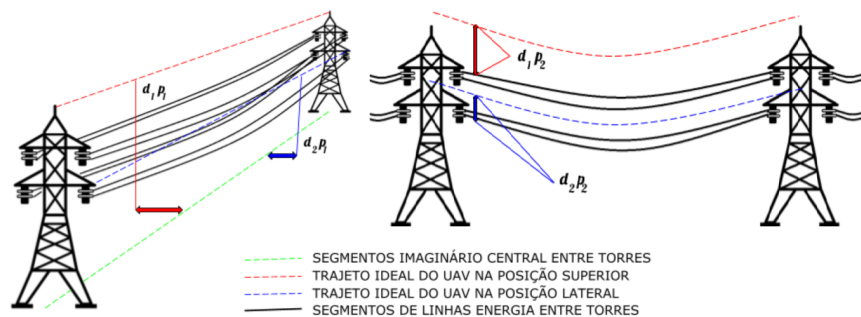


Figura 18 – Solução de rastreamento proposto. Fonte: Próprio Autor

Para obter esse ideário, seja  $\rho$  a posição em relação a visualização de duas ou mais torres por um observador externo,  $d$  a distância em metros de dois pontos entre dois segmentos definidos entre duas torres, e  $S$  o segmento de linha entre duas torres.

Temos que, assumindo uma imagem paralela ao plano, a distância horizontal entre trajetória do UAV (na posição superior) e o centro das duas torres deve ser o menor possível, a equação 3.1 se refere ao rastreamento central das linhas de transmissão de energia entre as torres.

$$d_1 p_1 = s_1 \quad (3.1)$$

Da mesma forma, assumindo uma imagem perpendicular ao plano, a distância entre o UAV na posição lateral e os segmentos de linhas deve ser centralizado em 3.2, que compete a visualização de todos os fios de uma das laterais das torres de transmissão de energia.

$$d_2 p_2 = (s_2 - s_3)/2 \quad (3.2)$$

Já as demais distâncias necessitam de uma constante  $C$  para que a legislação vigente seja atendida e se obtenha uma visualização completa dos fios, em 3.3 e 3.4 são apresentados a distância superior e lateral do UAV respectivamente.

$$d_2 p_1 = c_1 \cdot s_1 \quad (3.3)$$

$$d_1 p_2 = c_2 \cdot s_1 \quad (3.4)$$

Após determinar os valores, caso necessário, inicia-se um processo de correção simples de trajetória do UAV em relação a diferença obtida por essas distâncias. Para garantir essa situação, os UAVs devem estar equipados com um orientador e estabilizador da câmera, já que a movimentação do UAV não é constante, e isso pode influenciar nos resultados.

Para fins de comparação de resultados dos modelos, serão considerados uma orientação de câmera e 3 movimentos para cada posição do UAV:

- Drone Superior : Guinada para Frente (+Y) e Giro em YAW nos dois sentidos, com câmera orientada para baixo (+90° FPV em PITCH)

- Drone Lateral : Guinada para Frente (+Y) e eixo Z nos dois sentidos, com câmara orientada para lateral (+90° à esquerda ou -90° à direita FPV em YAW)

Ambos UAVs atuarão simultaneamente, podendo também ser usado na posição espelhada, totalizando quatro UAVs colaborativos e com atividades distintas. Atuando em conjunto e em comunicação constante, as decisões de movimentos de cada agente inteligente especialista em suas ações pode ser compartilhado entre eles de modo colaborativo. Além disso, é possível fazer uma cobertura completa da estrutura e obter imagens mais detalhadas para inspeção de pontos específicos.

## 3.2 Arquitetura do Sistema

O modelo estrutural da arquitetura geral do sistema, em formato de fluxograma, é apresentada na 19, nele é indicado os processos específico em cada nicho atuante.

No fluxograma apresentado, as setas azuis indicam a transferência de dados ou ações entre uma máquina e outra por meio de um protocolo de comunicação. As setas vermelhas representam a vinculação das execuções internas de uma mesma máquina.

A proposta é um protótipo geral do projeto para atuação contínua da solução, com o objetivo principal de atuar em ambiente real.

O sistema envolve diversos atuantes, inicialmente todas as máquinas necessitam estar em condições de operação e ter sucesso na conexão entre elas. Em seguida o operador posiciona o UAV para início da tarefa. O computador de bordo necessita receber uma sequência de imagens recolhidas da câmara embarcada.

A partir disso é possível realizar as operações necessárias para definição da ação como variável de resultado discreta, que definirá o movimento e velocidades, essa informação é recebida pelo controlador de voo que determina a execução pelo UAV. Essa ação também é definida como classe no *Dataset* a ser construído.

Quando o modo de enxame está ativo, todos os UAVs se comunicam cooperativamente, de forma que seus comandos sejam aguardados e compartilhados entre si. Esse processo se repete até que a tarefa definida seja encerrada, ou conforme é

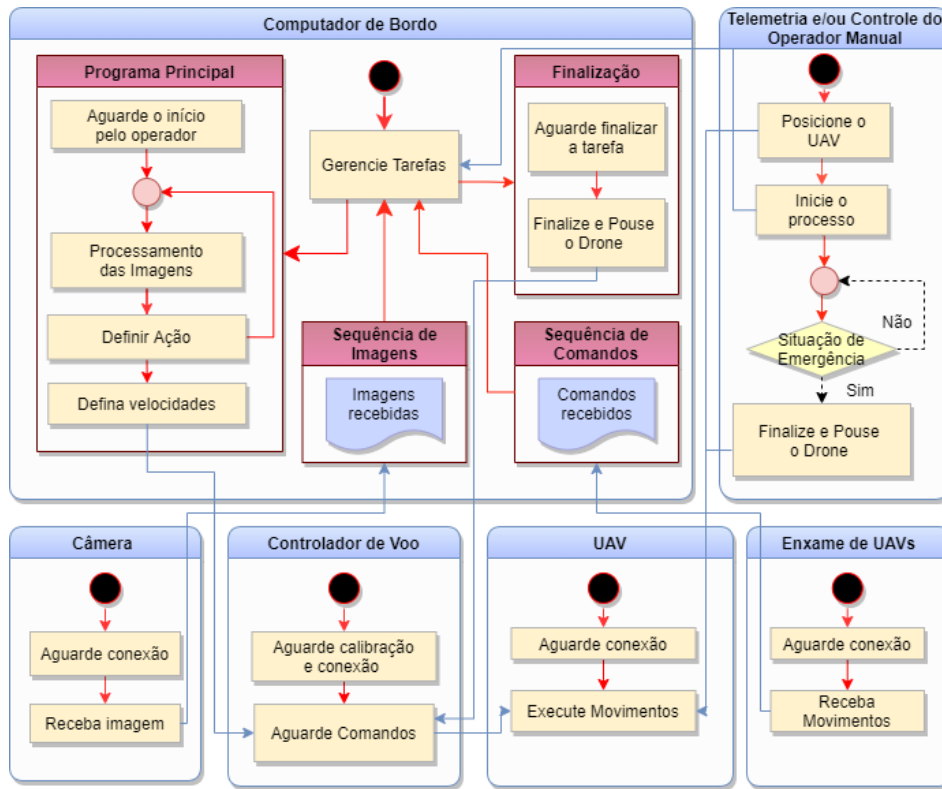


Figura 19 – Fluxograma da Arquitetura do Sistema. Fonte: Próprio Autor

determinado pela legislação, ocorra uma inconsistência no processo que necessite de atuação do operador.

### 3.3 Detecção de Bordas

Nesta seção é detalhado o algoritmo desenvolvido para rastrear as linhas de transmissão utilizando LSD, TH e THP, o passo a passo do processo é apresentado na Figura 20.

Inicialmente, são definidos os parâmetros do sistema, como posição inicial do UAV, disponibilização e frequência da obtenção das imagens, tamanho do trajeto a ser percorrido, e velocidade padrão dos movimentos definidos. Após a obtenção da primeira imagem, é realizado um redimensionamento para e conversão da imagem para escala de cinza, permitindo assim uma otimização dos processos a seguir.

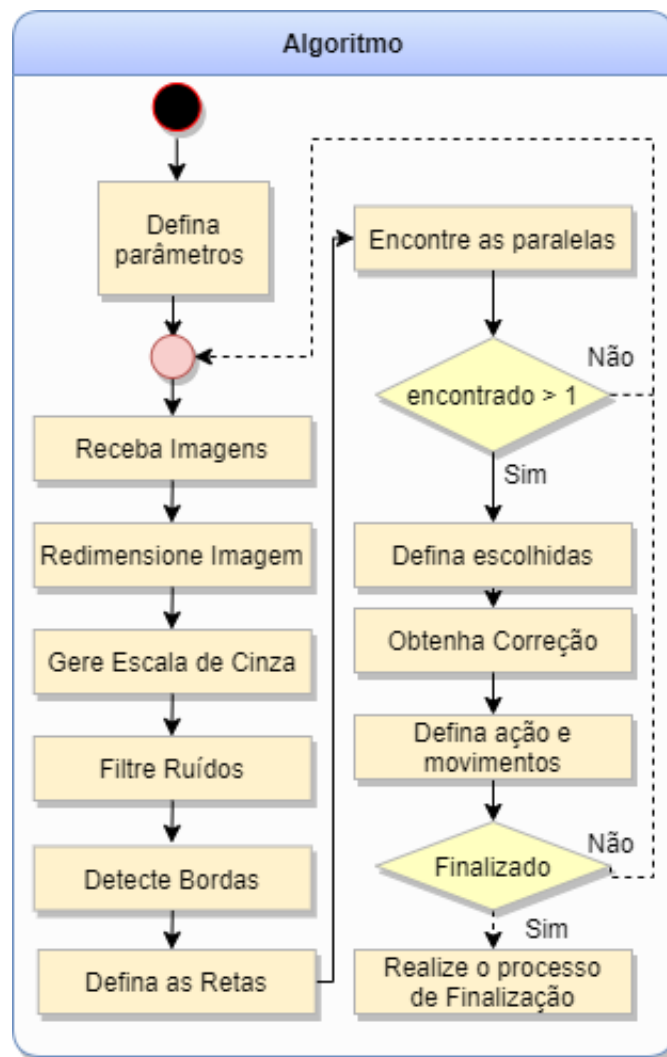


Figura 20 – Fluxograma do funcionamento do algoritmo de ED. Fonte: Próprio Autor

Um processamento na imagem determinado como mais próximo do ideal para a situação geral é realizado para facilitar a identificação dos segmentos de retas de acordo com as imagens obtidas, dessa forma, o filtro de ruído com tamanho de núcleo de 3 ou 5 é aplicado para experimentos.

Já na detecção das bordas, um limiar de [50,300] é utilizado para classificar quaisquer bordas gradientes com esse intervalo como válidas. As bordas são pas-

sadas para um TH, THP e LSD (MARTINS et al., 2020).

Após esses processos, são encontrados todos os conjuntos de retas paralelas aceitas, essa restrição é definida pelo tamanho mínimo definido pela média dos resultados obtidos, para desconsiderar traços pequenos pertencentes as torres ou elementos externos.

Caso não haja nenhum encontrado o processo deve ser reiniciado, o conjunto selecionado é o que possui os maiores tamanhos de segmentos de retas. Esse valor é utilizado para determinar o sentido do giro, ou, Subida e Descida a ser realizado pelo UAV superior ou lateral respectivamente.

O angulo formado entre o eixo central da imagem com espaço central das retas paralelas é obtido por meio da função da tangente hiperbólica, como visto na figura 21. foi escolhido por ter um comportamento estável para obtenção de um ângulo único no cálculo a partir de um ponto, no caso a extremidade da reta.

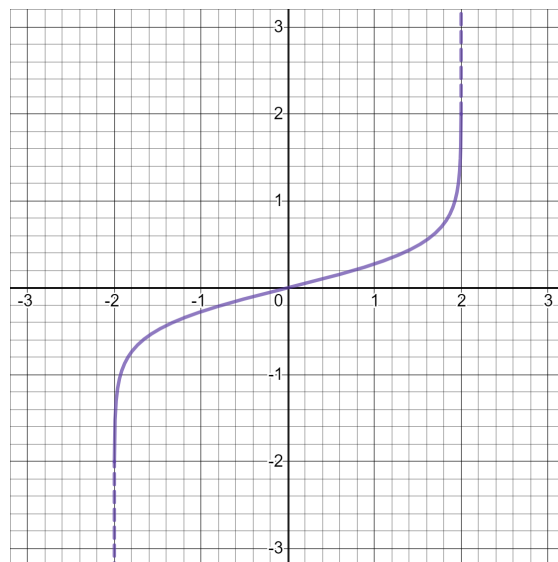


Figura 21 – Função arco tangente hiperbólica. Fonte: (PINTO et al., 2019)

Conforme a equação 3.5, caso exista mais de um angulo  $\alpha$  de ajuste identificado no conjunto  $\mathbf{A}$ , o valor é definido a partir do menor valor da diferença com a anterior  $\theta_{ultimo}$ .

$$Ação(x) = \begin{cases} esquerda, & \text{se } 0 < x < \frac{1}{3} \\ centro, & \text{se } \frac{1}{3} \leq x \leq \frac{2}{3} \\ direita, & \text{se } \frac{2}{3} \leq x \leq 1 \end{cases}, \text{ onde } x = \text{Piso}(\theta_a - \theta_{ultimo}), \forall a \in A \quad (3.5)$$

De modo de resultado com variável discreta, esse valor deve ser definido no intervalo do terço mais significativo do ângulo total de visão, determinando assim a ação que se transformará nos movimentos dos UAVs.

### 3.4 Aprendizagem Profunda

Nesta seção é apresentada a modelagem de um algoritmo de alta capacidade utilizando o SSD, a figura 22 desfia o processo de treinamento, avaliação e utilização do modelo.

No algoritmo de treino, inicialmente os parâmetros de treinamento são definidos e um modelo base SSD é então definido. Esse modelo é pré-processado com camada de extrações de características já definidas, dessa forma o treinamento é mais rápido e eficaz.

Os rótulos são carregados em formato TFRecord juntamente com as imagens, divididas em 79% para treino, 20% para validação e 1% para teste. A divisão foi estabelecida de modo empírico. O treinamento é realizado por uma quantidade pré-definida de passos, que é limitado principalmente por conta do tempo de treinamento. Essa divisão foi definida empiricamente, por meio de observações das soluções em outros assuntos, relacionados a visão computacional.

Os pesos e parâmetro são então ajustados na rede e avaliado, e então o treinamento prossegue. Caso o modelo obtenha um melhor rendimento, o modelo é exportado. Após finalização de todo o processo são realizadas inferências nas imagens de teste, para então apresentar os resultados do treinamento.

No programa principal o modelo de melhor resultado é carregado para realização das inferências a partir das imagens recebidas, até que a ação seja identificada, e o movimento do UAV definido e executado.

Dois formatos de identificação de classes são utilizados, o primeiro, classe única (monoclasse), utiliza apenas os cabos para identificação e definição de caixas deli-

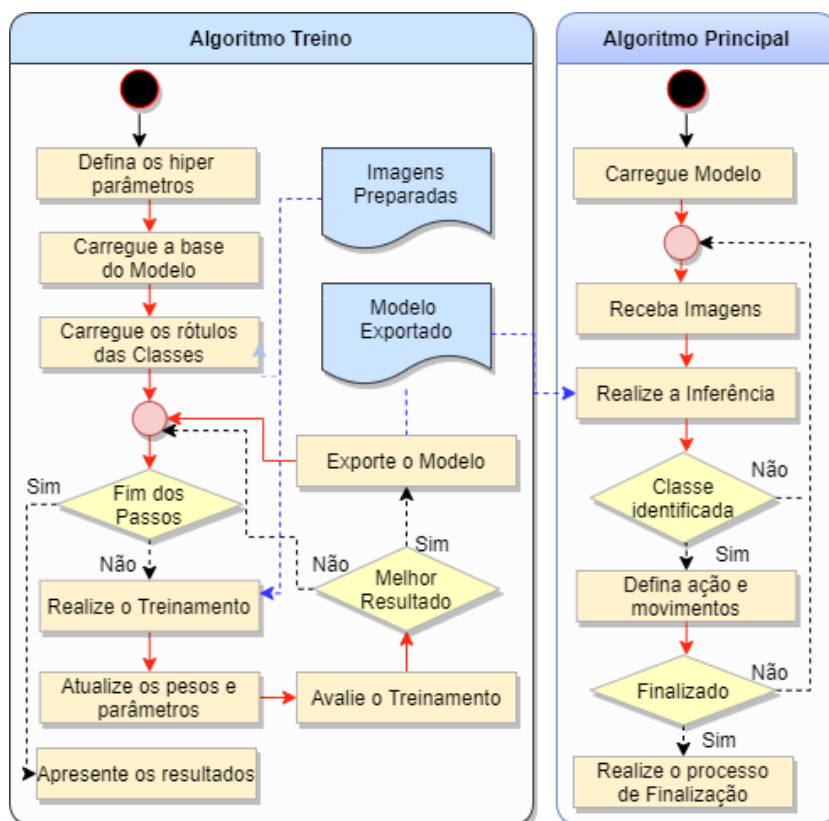


Figura 22 – Fluxograma do funcionamento do algoritmo SSD. Fonte: Próprio Autor

mitadoras, o segundo, múltiplas classes (multiclasse), define separadamente cada ação a ser tomada para cada drone definido.

### 3.5 Aprendizado por reforço

O fluxograma do algoritmo utilizado no treinamento e utilização, são apresentados na figura 23.

Da mesma forma do DL o treinamento inicia definindo o modelo da RNA e os hiper parâmetros, que nesse caso incluem o número de episódios, variável  $\gamma$  definida com um valor alto [0.9,1.0], e variável  $\epsilon$  que inicia com valores altos e decai de acordo com o treinamento [1.0;0.05]. Também é definido um modelo de

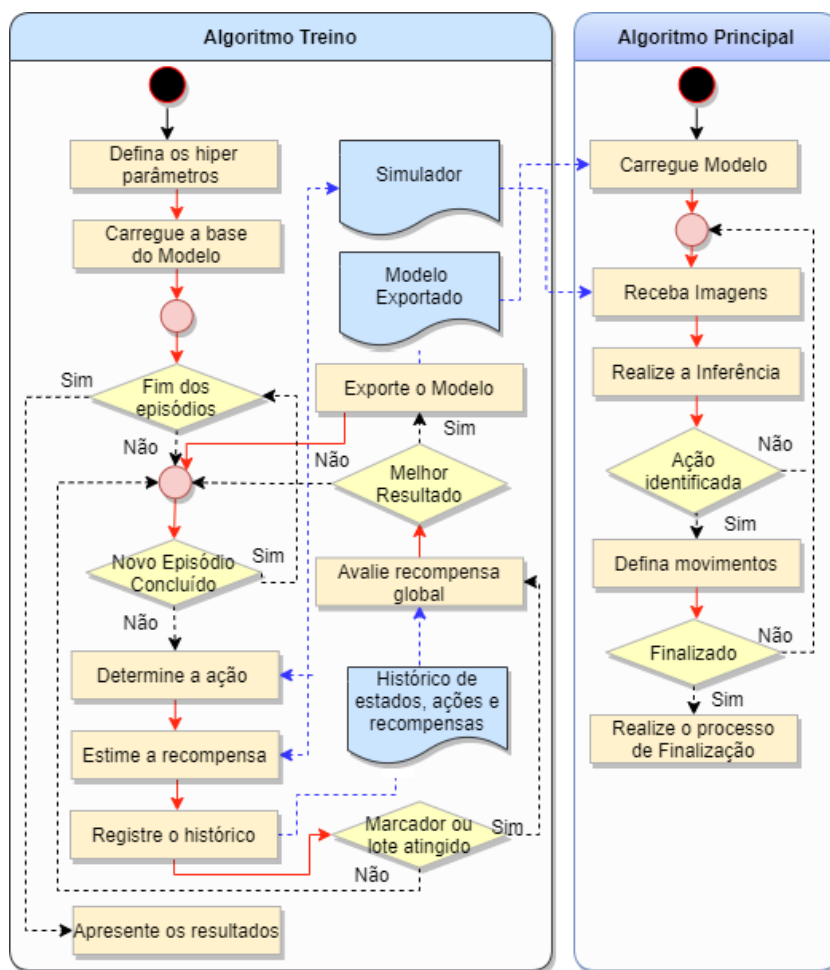


Figura 23 – Fluxograma do funcionamento do algoritmo de Aprendizado por reforço. Fonte: Próprio Autor

rede CNN utilizado pelo agente.

Um laço regula a quantidade de episódios a serem executados, e outro laço controla a continuidade de cada episódio. A cada passo do episódio imagens do simulador são carregadas para a rede que determina a ação que deve ser executada dependendo da aleatoriedade de  $\epsilon$ , essa ação é executada no simulador que imediatamente retorna a recompensa estimada dessa ação.

Caso a recompensa receba um marcador de encerramento, uma punição extra negativa é aplicado. O mesmo ocorre caso o UAV ultrapasse o ponto da pró-

xima torre do percurso, contudo é recebido um marcador de meta intermediária alcançada, e uma recompensa positiva.

Os valores são estimados na recompensa média que o agente pode receber ao longo do episódio. Essa estratégia evita que o agente permaneça preso a um máximo local de recompensa, possibilitando uma melhor exploração e solução.

Em todos os casos é realizado um registro de histórico contando o estado atual (imagem), ação e recompensa obtida. Após atingir um mínimo de lote de memória definido, a recompensa global é avaliada, ou seja, os registros dos últimos passos são processados na CNN e o modelo exportado em caso de melhor resultado comparado ao anterior. Caso o marcador seja de encerramento, o limite de passos ou a última meta foram atingidos, um novo episódio se inicia.

Na execução do algoritmo principal, o melhor modelo é carregado e as imagens são recebidas. Ao realizar a inferência na rede utilizando a imagem coletada, uma probabilidade de ações serão exibidas de acordo com a política, a de melhor resultado é escolhida, e os movimentos do UAV são executados.

Esse processo se repete até o final do percurso desejado, nesse caso não são necessárias estimativas e avaliação de recompensa, contudo esses dados podem ser utilizados para aprimorar o modelo.

## 4 Experimentos e Resultados

Esse capítulo apresenta inicialmente os resultados obtidos da síntese obtida na RSL, em seguida serão apresentados os cenários de testes simulados e reais. Uma análise dos experimentos é realizada com cada solução do estudo, a fim de selecionar o melhor de cada técnica.

Finalmente, com a intenção de validar as soluções obtidas neste estudo, também foram analisados alguns experimentos com a execução do percurso no ambiente sintético do Airsim, e em imagens reais coletadas de torres de transmissão de energia.

### 4.1 RSL

As análises realizadas nos artigos estão sintetizadas na tabela 3 as informações coletadas dos artigos selecionados, a forma mais econômica é utilizar somente uma câmera para processamento do algoritmo de detecção de bordas, que possui pouca demanda computacional, contudo a aplicabilidade em outros ambientes ficam limitadas.

Alguns artigos apresentaram de forma secundária soluções utilizando aprendizado de máquina com visão computacional. Para voo seguro, é possível desenvolver sistemas de redundância para sensores e motores (AFMAN et al., 2018).

Infere-se que um laser ou câmera infra vermelho especial pode auxiliar o algoritmo no rastreamento de linhas de transmissão, nos artigos estudados que aplicaram equipamentos diferenciados houve um grande retorno quanto a precisão e estabilidade da solução. Porém encarece a aplicação, e dependendo da modelagem matemática, pode não ser generalizado para todas as situações e imprevistos do ambiente externo.

Tabela 3 – Características identificadas nos artigos (MARTINS et al., 2020)

Referência	Material	Método	Testes	ARP	País	Acurácia (%)	Custo	Qualidade
(ZHANG et al., 2016)	Laser	DL com visão binocular	Voo real	NI	China	97	Médio	4.0
(HUI et al., 2019)	Câmera	DL com KCF	Voo real	Multirotor	NI	90	Baixo	5.0
(LUQUE-VEGA et al., 2014)	Câmera Infra vermelho	GCS com GPS	Voo real	Multirotor	NI	NI	Médio	4.0
(ZORMPAS et al., 2018)	Câmera	ED	Voo real	Multirotor	Grécia	NI	Muito Baixo	3.0
(DENG et al., 2016)	Laser	Algoritmo adaptativo	Laboratório	Multirotor	NI	NI	Médio	4.0
(HAMELIN et al., 2019)	Câmera e Lidar	Controle de tempo discreto	Voo real	Multirotor híbrido	NI	NI	Médio	4.5
(WU et al., 2019)	Sensores magnéticos	Meta heurística	Laboratório	Multirotor	NI	95	Médio	4.5
(ZHOU et al., 2016)	Câmera	ED	Voo real	Multirotor	Usa	NI	Baixo	4.5
(XIE et al., 2017)	Múltiplos	Multi sensor ajustável	Voo real	Helicóptero	China	98	Muito Alto	4.5
(DENG(B) et al., 2014)	GPS e IMU	GPS ajustável	Voo real	Asa fixa e multirotor	China	NI	Baixo	3.0
(MENENDEZ et al., 2016)	Câmera	ED	Laboratório	NI	NI	97	Médio	4.5
(GERKE; SEIBOLD, 2014)	Câmera	ED	Laboratório	Dirigível	NI	NI	Baixo	3.0
(ARAAR; AOUF, 2014)	Câmera	Servo visual	Laboratório	Multirotor	NI	NI	Baixo	4.5
(GAO et al., 2018)	Sensor Magnético	Algoritmo de modelo inverso	Simulação	NI	NI	NI	Baixo	3.0
(CHANG et al., 2017)	Laser	Algoritmo adaptativo	Laboratório	Multirotor híbrido	NI	NI	Alto	4.5

Para uso generalizado da solução, pode-se destacar outros tipos de aprendizado, como profundo, que permite alta precisão com processamento e treinamento relativamente baixo, ou mesmo aprendizado profundo que é pouco explorado nesse cenário, ambos utilizados recentemente com ótimos resultados em outros cenários com desafios semelhantes.

Essas tecnologias podem fornecer uma generalização para diferentes estruturas de torres em diferentes ambientes. Outros detalhes das observações realizadas nos artigos se encontram no Apêndice A seção 6.

## 4.2 Legislação

Para realização de um voo seguro e em consonância com o arcabouço regulatório em vigor, apresentado na legislação para acesso ao espaço aéreo brasileiro, um resumo do processo de solicitação de voo foi desenvolvido no estudo, conforme apresentado na Figura 24.

Inicialmente é necessário possuir a documentação exclusiva para o ARP denominada de SISANT, no anexo 8.3 é apresentado o documento solicitado pelo autor e emitido para autorizar a aeronave utilizada no estudo. O documento possui informações das características da aeronave, e indica explicitamente aptidão para realização de voos não recreativos, com o ramo de atividade específico para treinamento, pesquisa e desenvolvimento.

Para este estudo, foram utilizados equipamento de rádio frequência que possui amplo mercado no Brasil, e são homologados pela Anatel por padrão. A equipe definida nos voos foi sempre formada de integrantes do grupo Viscap (Visão, Sistemas de Computação e Aplicações), da própria Unifei, e possuem cadastro de piloto no DECEA.

As missões realizadas para este estudo atuaram sempre fora do princípio da sombra, ou seja, fora da delimitação em torno da estrutura e acima de 5 metros de altura e 30 metros do entorno, evitando assim também a atuação do campo eletromagnético produzido pelas linhas de transmissão, conforme representado na Figura 25. Dessa forma, não se faz necessário autorização específica das empresas responsáveis pela estrutura, ou qualquer outra autorização especial.

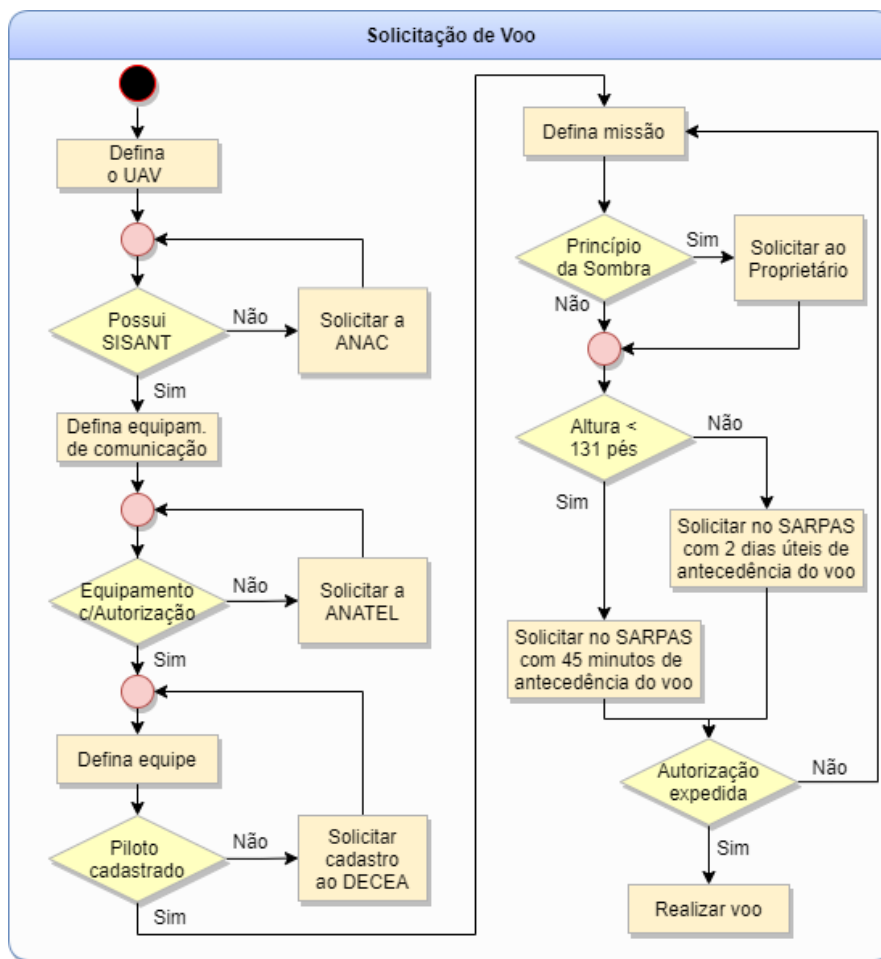


Figura 24 – Fluxograma para solicitação de voo seguindo a legislação atual. Fonte: Próprio Autor

Neste estudo os voos atingiram no máximo 130 pés de altura em relação ao solo, e sempre em condições meteorológicas visuais VLOS. Não são abordadas as questões específicas de proximidade de aeródromos por não existirem na proximidade com a Unifei, contudo essa e outras variáveis podem influenciar diretamente no prazo e decisão da autorização de voo por parte do DECEA. As condições aqui apresentadas para este estudo se aplicam ao perfil 1 do padrão ICA 100-40. No anexo B 8.3 são apresentadas 6 solicitações de voo para testes do equipamento, obtenção das imagens e testes, autorizadas em diferentes locais e datas.

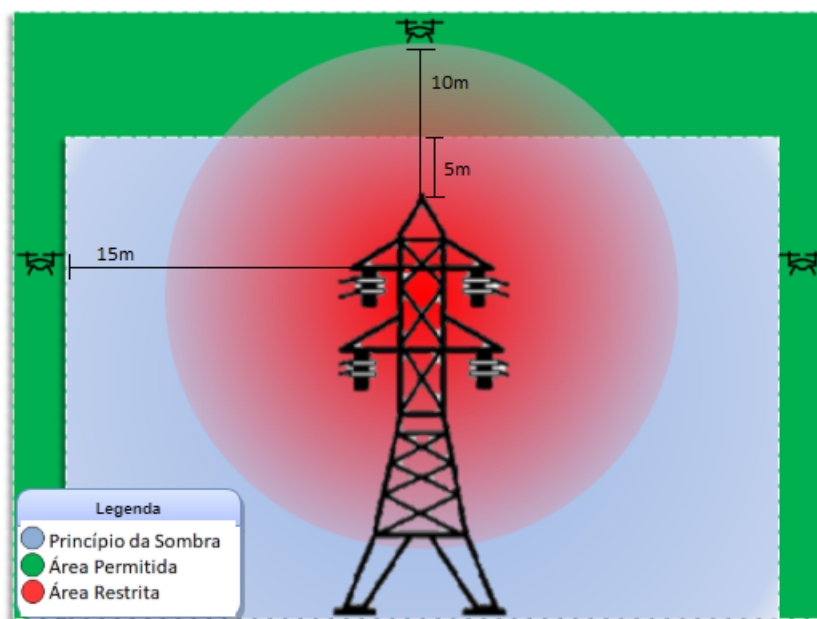


Figura 25 – Área de atuação na torre de alta tensão. Fonte: Próprio Autor

### 4.3 Ambiente de simulação

Um cenário de teste foi criado, contendo torres de suporte, linhas de transmissão, e ambientes diferentes para idealizar um cenário e, por consequência, ter mais elementos para serem processados pelo algoritmo de visão computacional.

#### 4.3.1 Torres

Foram reproduzidas no estudo 3 diferentes tipos de torres de alta tensão, utilizadas na transmissão de energia de longas distâncias, e que são comuns nas MBT (KIR-TLEY, 2020). São elas, uma de 115 kV e duas de 230 kV, conforme apresentado na Figura 26.

#### 4.3.2 Mundo

Dois diferentes ambientes também foram reproduzidos neste estudo para os testes, de acordo com apresentado na Figura 27. O primeiro retrata uma simulação limpa,

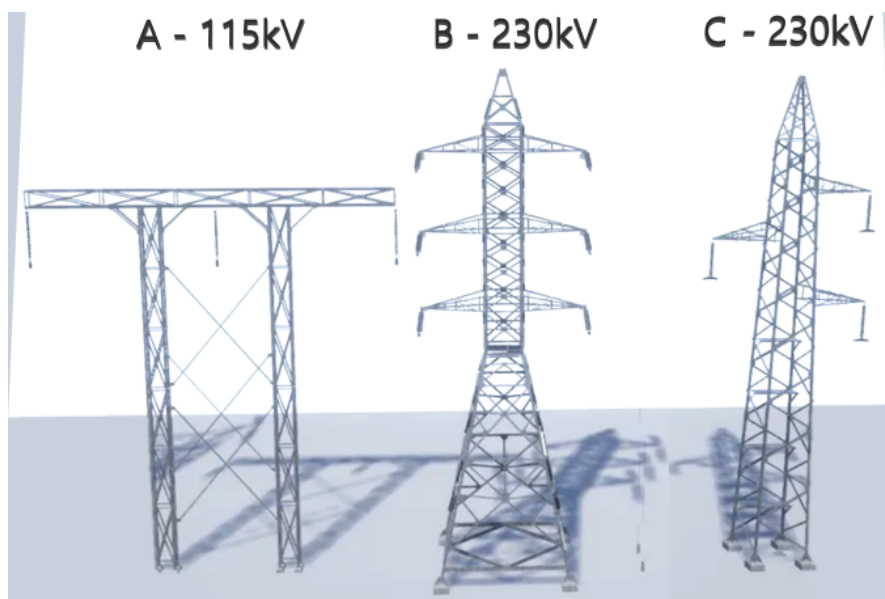


Figura 26 – Torres de transmissão de alta tensão utilizadas no simulador Airsim.  
Fonte: Próprio Autor

somente com as torres e linhas conectadas. Já o segundo possui um solo mais realista, com diferenciação de relevo e objetos em solo não pertencentes a estrutura de linha de transmissão de energia.

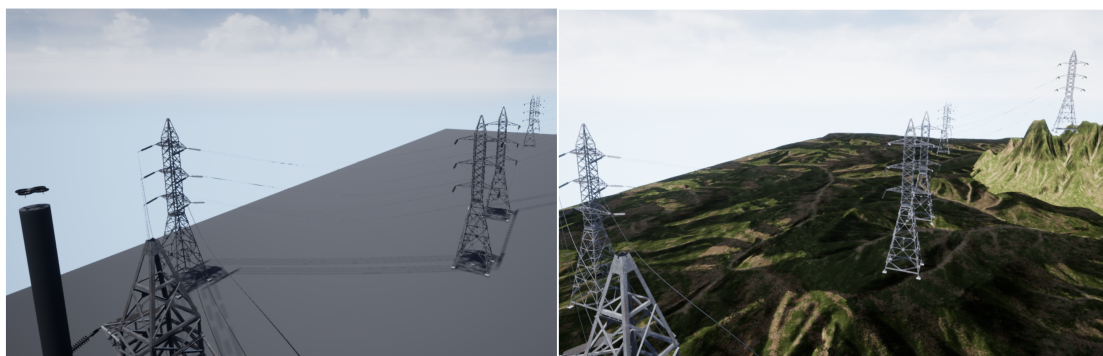


Figura 27 – Ambientes de testes utilizadas no simulador Airsim. Fonte: Próprio Autor

## 4.4 Preparação do Dataset

O *Dataset* produzido com imagens reais e simuladas levaram em consideração percursos e posições nas inspeções em linhas e torres de transmissão de energia, com os 3 movimentos possíveis no trajeto, de fácil e difícil detecção. No total foram colhidas e rotuladas 1005 imagens de 3 tipos de torres diferentes, em 3 ambientes reais e 3 ambientes simulados. Essas imagens foram rotuladas de acordo com a posição da linha representada na imagem, para um dos *Dataset*, e em outro, de acordo com a ação discreta a ser executada pelo ARP. Uma amostra dessas imagens rotuladas é apresentada na Figura 28.



Figura 28 – Imagens do *Dataset* rotuladas, rótulos em linhas à esquerda, visão de cima no centro, e visão lateral à direita. Fonte: Próprio Autor

Na preparação do *Dataset* foram as imagens divididas em 905 imagens para

treinamento (90%), 80 imagens para validação (8%) e 20 imagens para teste (2%). Com a intenção de melhorar o aprendizado do modelo, foi gerado um reforço de diferenciação de contraste [-10%,+10%] e brilho [-10%,+10%] totalizando 1300 imagens para o treinamento, 116 para validação e 32 para testes, mantendo uma distribuição regular entre as classes.

Não foram consideradas outras formas de acréscimo de *Dataset* como rotação, cortes, oclusão, mosaico e alteração de angulação pois alteram diretamente o resultado da inferência, e pouco contribuem. Contudo testes de treinamento foram efetuados utilizando incremento de ruído e borrão tendo uma pequena melhora em relação ao uso somente do contraste e brilho em escala de cinza.

## 4.5 Análise das Soluções

Para realizar avaliações no sistema, é interessante conhecer as medidas importantes, esta medição é feita por porcentagem de resultados positivos ou negativos obtidos, com estes valores pode-se obter uma melhor detecção a medida que o treinamento avança.

Esses conjuntos de imagens reais passarão por processamento de manipulação, e pelas técnicas de reconhecimento apresentadas, considerando  $n$  o número de instâncias, VP (Verdadeiros Positivos), FP (Falsos Positivos) e FN (Falsos Negativos), sua eficácia será avaliada com Precisão, Revocação, *F1 Score* e/ou recompensa (GOUTTE; GAUSSIER, 2005).

A equação da precisão 4.1 mede a porcentagem quão correto são as previsões de um modelo. Essa métrica permite uma melhor avaliação quando FPs são considerados mais prejudiciais que os FNs.

$$P = \frac{VP}{VP + FP} \quad (4.1)$$

A equação da revocação 4.2 mede o quão corretos são os resultados dados como positivos. Nesse caso, os FNs são considerados mais prejudiciais que os FPs.

$$R = \frac{VP}{VP + FN} \quad (4.2)$$

A equação do F1 Score 4.3 é uma métrica para o valor harmônico entre a precisão e a revocação. Essa métrica permite uma rápida avaliação do modelo, indicando quando a precisão ou a revocação está baixa.

$$F1 = 2. \frac{P * R}{P + R} \quad (4.3)$$

O critério usado no RL será da recompensa média máxima atingida e tamanho médio de episódio do treinamento total, em relação ao total de passos executados.

Para o treinamento e avaliação da solução em simulador, foi utilizado um computador com processador Intel(R) Core(TM) i7-4770 de 3.40GHz, com 16GB DDR3 SDRAM (*Synchronous Dynamic Random-Access Memory*) instalada, sistema operacional Windows 10 de 64 bits e placa de vídeo NVIDIA GeForce GT710 de 2GB GPU G-DDR3.

## 4.6 Detecção de bordas

Conforme apresentado no modelo, experimentos com ED foram planejados, e os resultados são apresentados na tabela 4.

Tabela 4 – Experimento utilizando ED em imagens testes com diferentes parâmetros

<i>Grid</i>	LI	LS	Superior			Lateral		
			TH	THP	LSD	TH	THP	LSD
3 x 3	0	100	13.0	10.75	0.0	21.0	21.75	0.0
3 x 3	100	100	3.0	5.75	2.0	16.0	16.0	19.0
5 x 5	50	100	29.5	<b>25.5</b>	<b>9.25</b>	61.0	19.25	<b>29.75</b>
5 x 5	50	200	28.25	26.5	<b>9.25</b>	53.75	18.5	21.75
5 x 5	50	300	27.25	23.0	<b>9.25</b>	50.5	<b>23.0</b>	21.25
5 x 5	100	100	29.25	24.5	<b>9.25</b>	60.0	18.5	26.25
5 x 5	100	200	27.75	23.75	<b>9.25</b>	53.25	18.25	21.5
5 x 5	100	300	26.5	23.0	<b>9.25</b>	50.0	20.5	21.25
5 x 5	150	100	28.25	24.5	9.0	56.0	24.0	24.75
5 x 5	150	200	26.5	24.25	9.05	53.75	22.25	21.25
5 x 5	150	300	26.5	22.25	9.05	49.5	19.5	21.25

Nela são apresentados o grid, LI e LS utilizado no algoritmo Canny, e então calculadas a média dos resultados referentes a quantidade de linhas potenciais utilizando os algoritmos TH, THP e LSD em imagens testes com imagens aleatórias contendo linhas. Os resultados inferiores a 5 linhas válidas para todos algoritmos foram descartados, e os resultados com mais linhas válidas são destacados na tabela.

Após definidos os melhores parâmetros para cada algoritmo de acordo com a tabela anterior, foram realizados experimentos com imagens de linhas de transmissão de energia reais e do simulador, comparando com as ações preditivas de acordo com modelo estabelecido na seção anterior. Os valores obtidos são apresentados na tabela 5.

Tabela 5 – Experimento utilizando os algoritmos e determinando as ações

	Real						Simulador					
	Cima			Lateral			Cima			Lateral		
	<i>P</i>	<i>R</i>	<i>F1</i>	<i>P</i>	<i>R</i>	<i>F1</i>	<i>P</i>	<i>R</i>	<i>F1</i>	<i>P</i>	<i>R</i>	<i>F1</i>
TH	0,29	<b>0,40</b>	0,33	0,17	<b>0,14</b>	0,15	0,17	<b>0,20</b>	0,18	0,29	<b>0,40</b>	0,33
THP	<b>0,31</b>	<b>0,40</b>	<b>0,35</b>	<b>0,18</b>	<b>0,14</b>	<b>0,16</b>	<b>0,22</b>	<b>0,20</b>	<b>0,21</b>	<b>0,36</b>	<b>0,40</b>	<b>0,38</b>
LSD	0,23	0,30	0,26	0,17	<b>0,14</b>	0,15	0,17	<b>0,20</b>	0,18	0,29	<b>0,40</b>	0,33

Foi definido para comparação com outros métodos o THP. Os parâmetros selecionados para esse algoritmo possui grid 5x5 e limiar [0,100] para o EDs Canny, na identificação dos segmentos de linha.

## 4.7 Aprendizado Profundo

Nesta seção são avaliados treinamentos de 6 modelos populares SSD de alta performance utilizados no estudo, para treinamento do *Dataset* desenvolvido, algumas configurações e características dos modelos são apresentadas na tabela 6. Definição para NA (Não se Aplica).

Esses modelos passaram por um treinamento de 10.000 passos, com tamanho de lote 12, e 50 imagens aleatórias para cada etapa de avaliação. Com o treinamento concluído, o gráfico de inferência recém-treinado é extraído, para uso posterior na realização da detecção dos objetos. Além disso, a exportação dos pesos treinados

Tabela 6 – Comparação de alguns modelos de SSD populares

Modelo	Ativação	FPN	Lote	Otimizador	Arquiteturas	Nº Parâmetros
Inception v2	ReLU	NA	24	RMS	Resnet	10M
Mobilenet v1	ReLU	NA	24	RMS	Resnet	3M
Mobilenet FPN	ReLU	[3,7]	64	<i>Momentum</i>	Resnet	4M
Mobilenet v2	ReLU	NA	24	RMS	Inverse	3M
Resnet 101 FPN	ReLU	[2,7]	64	<i>Momentum</i>	Resnet	42M
Resnet 50 FPN	ReLU	[3,7]	64	<i>Momentum</i>	Resnet	25M

é usado para uma simples inferência com as mesmas imagens brutas do conjunto de teste, para uma rápida verificação empírica da qualidade do modelo.

### 4.7.1 Monoclasse

O primeiro treinamento foi realizado utilizando somente uma classe, com os dados balanceados, determinada pela inclusão de cada cabo de energia elétrica e sua extensão na imagem em uma caixa delimitadora, e nomeada de linha. Após a identificação do segmento de linha é possível identificar com detectores o angulo formado por essa linha e determinar a ação a ser tomada. Os resultados em gráficos dos treinamentos são apresentados abaixo.

Na Figura 29 que representa a função de perda em relação a validação, os melhores resultados são obtidos pelos modelos Resnet e MobileNet v1 FPN. Especialmente no MobileNet, é apresentado valores próximos ao ideal para uma RNA, e com tendência de queda.

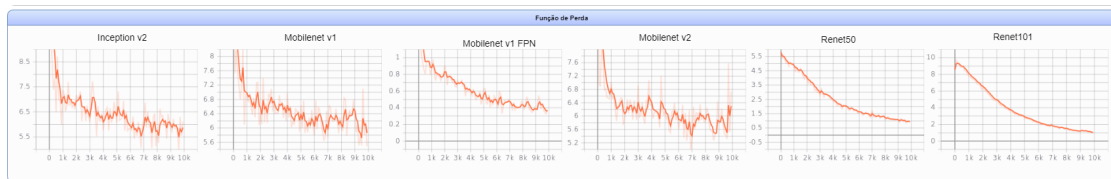


Figura 29 – Função de Perda dos Modelos SSDs Monoclasse. Fonte: Próprio Autor

Conforme é apresentado na Figura 30, a precisão média nos treinamentos dos modelos, obteve um bom resultado somente no Resnet101 e Mobilenet v1 FPN.

Ambos ultrapassam 25% de precisão, indicando uma taxa de acerto ainda baixo em relação as caixas delimitadoras esperadas.

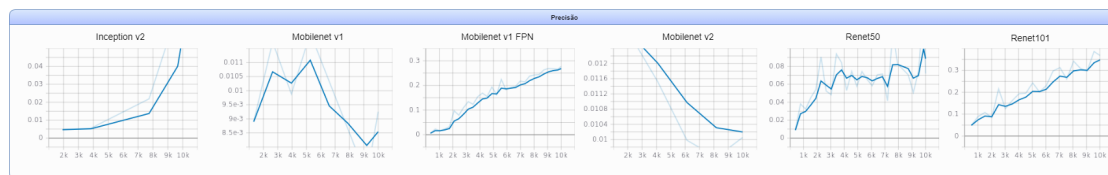


Figura 30 – Média da Precisão dos Modelos SSDs Monoclasse. Fonte: Próprio Autor

Nos gráficos de revocação, apresentados na figura 31, não indicam uma boa atuação para evitar os falsos negativos, ou seja, as caixas delimitadoras e seus rótulos não representam o esperado nas imagens. Apesar disso o MobileNet v1 FPN chegou a quase 10%, e pode tentar ser melhorado com mais treinamento ou imagens rotuladas.

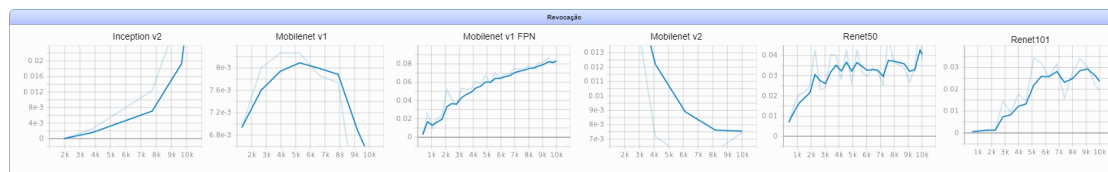


Figura 31 – Média da Revocação dos Modelos SSDs Monoclasse. Fonte: Próprio Autor

Após obtenção da precisão e revocação, calculou-se o  $F1 = 14,29$  obtidos pelos valores dos gráficos, indicando um desempenho muito menor que a técnica anterior, modelada a detecção de bordas. Apesar dos baixos resultados obtidos nos modelos de única classe, a inferência pode reduzir o espaço de atuação do detector de retas na imagem capturada pelo ARP no rastreamento.

## 4.7.2 Multiclasse

No modo multiclasse todos os modelos atingiram o objetivo de reduzir o valor da perda, além disso, os modelos MobileNet v1 FPN, Resnet 50 e 101 apresentaram

melhores resultados, indicando que possuem um melhor mapeamento dos parâmetros da rede. Contudo o modelo Resnet 101 possui uma atualização exponencial de valores muito alta no início do treinamento, indicando que a função de perda possui uma atualização muito alta, e sempre fica presa em um mínimo local para esse *Dataset*. Os gráficos da função de perda dos treinamentos são apresentados na figura 32.

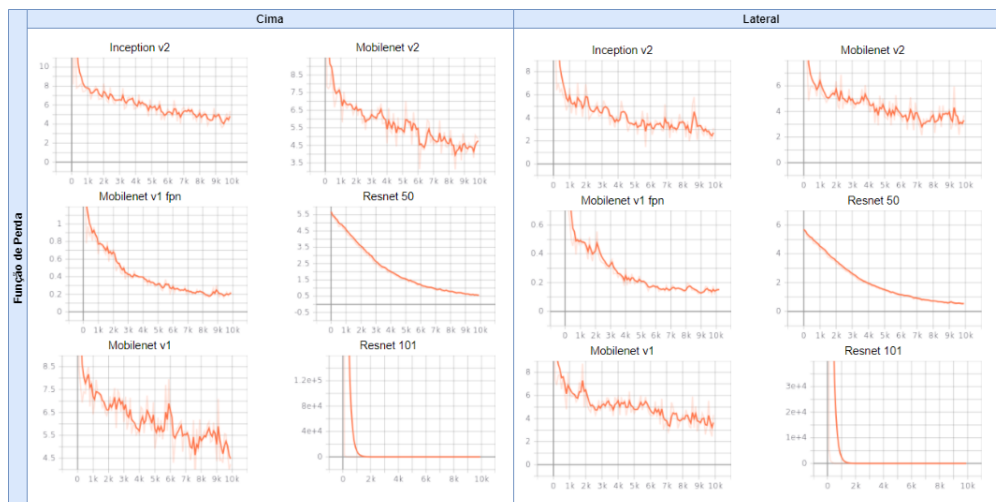


Figura 32 – Função de Perda dos Modelos SSDs Multiclasse. Fonte: Próprio Autor

A média ponderada da precisão, outro indicativo de sucesso no treinamento, apresentada na figura 33. Os modelos Resnet50 e MobileNet v1 FPN obtiveram os melhores indicadores, com tendência a avanço em um treinamento mais longo, chegando a 70% e 65% de precisão respectivamente. Apontando que o treinamento utilizando por mais de uma classe pode ajudar o algoritmo a uma melhor inferência da ação e caixa delimitadora a serem obtida.

A revocação média apresentada na Figura 34, segue a tendência dos demais gráficos e modelos, com os resultados em torno de 70%, indicando que a RNA conseguiu extrair os detalhes para identificação das ações de movimentos.

A revocação nesse cenário de identificação de ações de rastreamento em linhas de transmissão se torna importante na atuação em ambientes poucos conhecidos, já que reflete em maior parte, a determinação das caixas delimitadoras com falsos positivos.

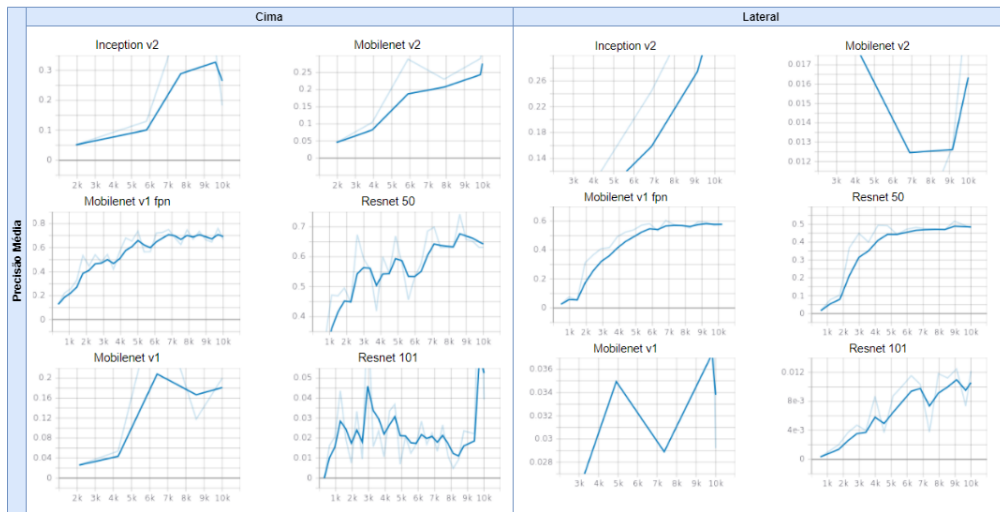


Figura 33 – Média ponderada da Precisão dos Modelos SSDs Multiclasse. Fonte: Próprio Autor

O modelo de melhor atuação no treinamento realizado foi o MobileNet v1 FPN, os resultados oferece uma maior segurança para um treinamento com maior duração no *Dataset* proposto e aplicação no comparativo com os demais métodos.

## 4.8 Aprendizado por reforço

Nesta seção serão avaliados os 3 algoritmos de aprendizado por reforço e o modo de recompensa para estudo na solução proposta, utilizando o ambiente de simulação do mundo limpo e diferentes torres.

### 4.8.1 Recompensa

Para identificação da melhor forma de recompensa testes matemáticos foram realizados de forma empírica, levando em consideração o objetivo principal, ajustar a visão das linhas de transmissão e arredores próximo na busca de identificação de problemas. Sendo a reta  $AB$  o segmento entre duas torres sendo uma torre o ponto  $A$  e a próxima torre o ponto  $B$ , e a posição do ARP o ponto  $P$

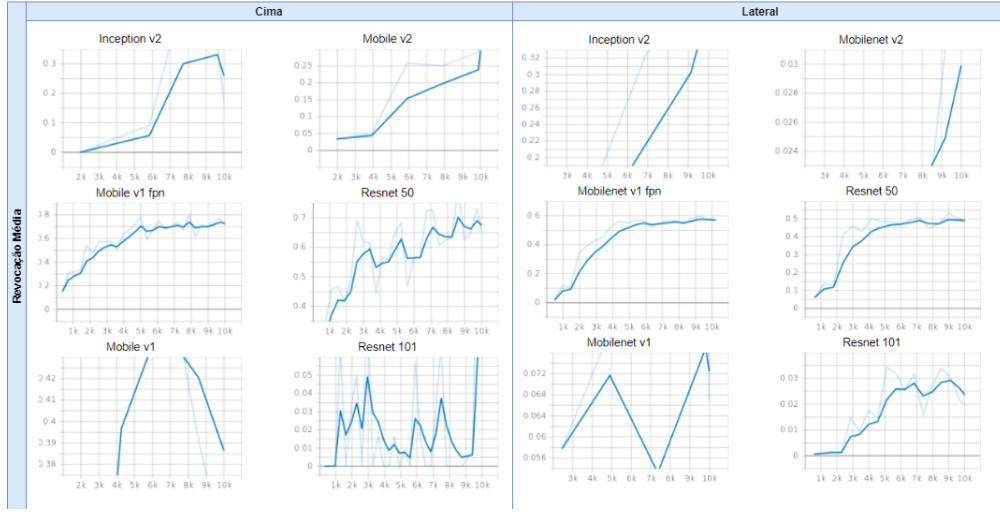


Figura 34 – Média da Revocação dos Modelos SSDs Multiclasse. Fonte: Próprio Autor

Inicialmente foi determinado a recompensa de distância simples entre dois pontos  $P$  e  $B$ , obtida pela equação 4.4 . Apesar da execução correta do trajeto ao longo do treinamento, o campo de visão das linhas ficou comprometido e a recompensa prosseguia crescente, já que bastava o ARP encontrar apenas um caminho até a próxima torre.

$$distancia(P, B) = \sqrt{(P_x - B_x)^2 + (P_y - B_y)^2 + (P_z - B_z)^2} \quad (4.4)$$

Uma recompensa por meio da distância mínima de um ponto para o centro dos segmentos de linhas foi testado. utilizando vetores em módulo, a distância medida de  $P$  até  $AB$  dada pela equação 4.5 negativada é dada como recompensa, ou seja, quanto mais próximo do zero maior a recompensa.

$$distancia(P, AB) = \left\{ \begin{array}{l} 0, \text{ se } A = P \text{ ou } B = P \\ |P - A|, \text{ se } \theta(AB, AP) > 90 \\ |P - B|, \text{ se } \theta(AB, BP) > 90 \\ \frac{|(B-A) \times (A-P)|}{|B-A|}, \text{ caso contrário} \end{array} \right\} \quad (4.5)$$

A recompensa por episódio no decorrer do treinamento mostrou que a recompensa atendia a solução proposta, já que incentiva o agente a centralizar o campo de visão nas linhas. Contudo, esse modo de recompensa incentiva a execução de movimentos desnecessários que distancia o agente da próxima torre.

Dessa forma, uma composição das recompensas é proposta para acelerar o aprendizado do agente, a equação 4.6 define o peso de relevância para cada distância obtida negativada.

$$\text{recompensa} = (\text{distancia}(P, B) * 0.1 + \text{distancia}(P, AB) * 0.9) * -1 \quad (4.6)$$

Esse formato recompensa permite ao agente identificar a melhor ação a ser tomada, para atingir a próxima torre se mantendo centralizado quanto ao campo de visão das linhas de transmissão de energia. Foi então definido os valores para os marcadores, -50 para o punitivo e +20 para o de meta, definidos na seção modelo 46, e sem limite tanto de sucesso, quanto de falha, para o acúmulo de recompensa.

## 4.8.2 Algoritmos

Os algoritmos de RL propostos na seção de materiais e métodos foram treinados em um ambiente de espaço contínuo, com ações discretas e agente individual atuante. Como hiperparâmetros, foi utilizado como taxa de aprendizado de 0,002, tamanho de lote 32, por conta da quantidade de passo, episódios e baixo desconto, que auxiliam no treinamento a longo prazo de acordo com observações empíricas em outras soluções. O tempo de treinamento foi de até 15.000 passos, os gráficos de tamanho do episódio e recompensa média são apresentados na Figura 35.

De acordo com os resultados apresentados, o aprendizado por reforço profundo teve melhor atuação no ARP posicionado na parte superior da estrutura, sendo

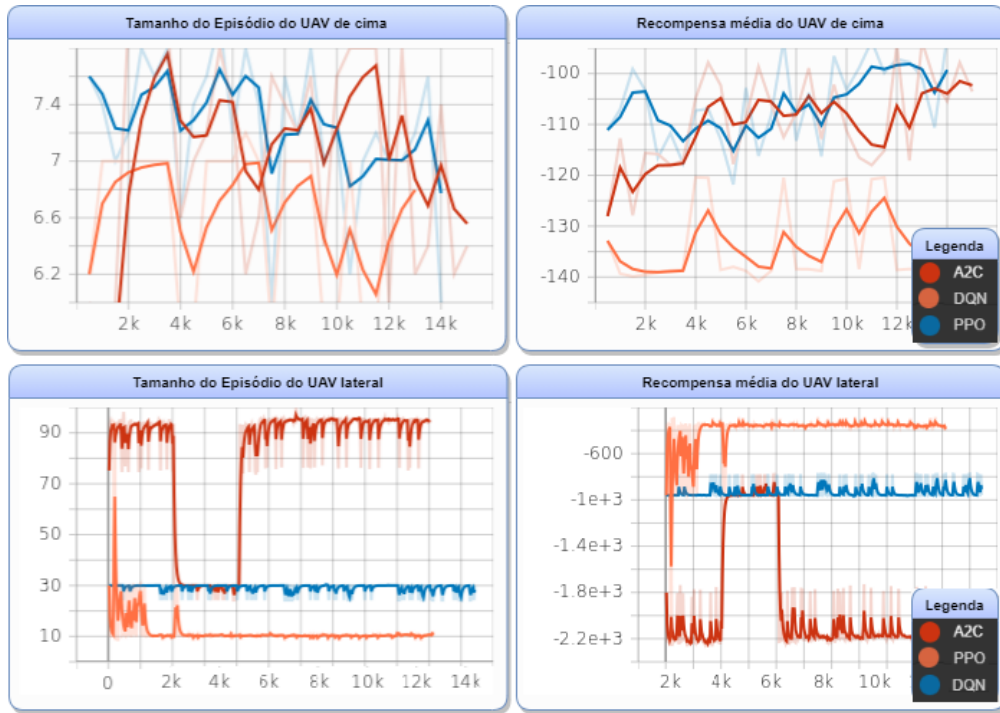


Figura 35 – Resultados do Treinamento de Aprendizado por Reforço Profundo. Fonte: Próprio Autor

que o ARP lateral, mesmo com baixa taxa de atualização, não conseguiu progredir com o aumento da recompensa.

O algoritmo A2C obteve os melhores resultados no geral, obtendo uma tendência de aumento de recompensa e permanência maior em um mesmo episódio ao decorrer do treinamento. Os algoritmos PPO e DQN também obtiveram bons resultados, contudo não indicam uma tendência a melhor convergência em um treinamento de maior prazo.

## 4.9 Experimentos no Percurso

Conforme é apresentado na Figura 36, o percurso desenvolvido no simulador foi baseado exclusivamente no conjunto de torres próximo a subestação, onde foram realizadas capturas de imagem para o *Dataset*.

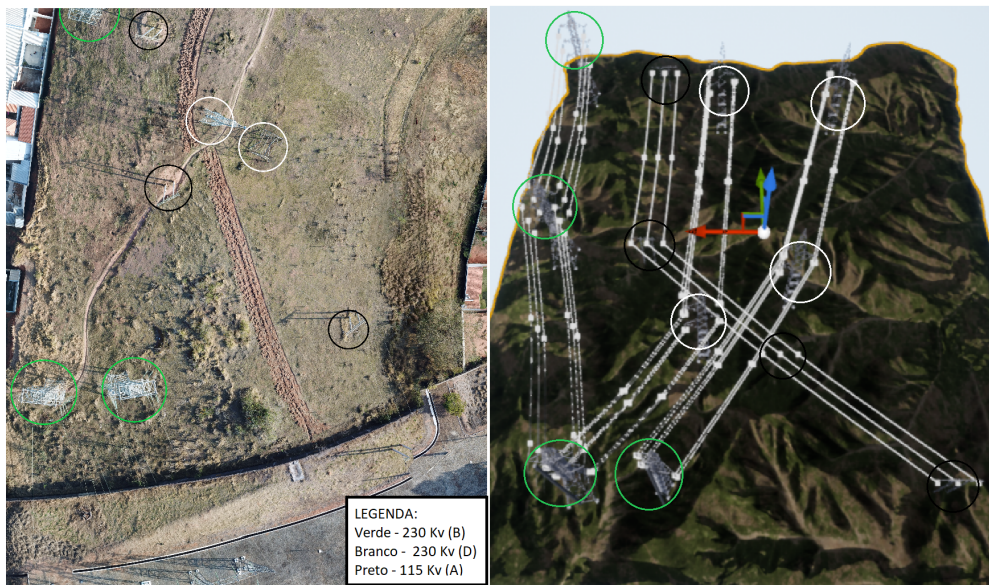


Figura 36 – Percurso Real à esquerda e no Simulador Airsim à direita. Fonte: Próprio Autor

Foi levado em consideração a utilização de todos movimentos do modelo definido, além de utilizar as torres e ambientes apresentados na sub seção 5.3 Ambiente de simulação.

#### 4.9.1 Variáveis de Ambiente

Com a intenção de analisar amplamente os modelos treinados, foram definidos experimentos de teste para examinar o quanto certos fatores afetam as variáveis a eles relacionadas. Para inspecionar o efeito exato de cada fator e suas combinações, o planejamento fatorial foi o modelo escolhido para os experimentos. A ideia é selecionar níveis para cada fator escolhido e ver como cada nível impacta no experimento. Uma vez que o projeto fatorial trabalha com fatores e variáveis, 3 fatores com 2 níveis e 3 variáveis cada foram definidos, e são apresentados na Tabela 7.

As variáveis de medição são executadas estritamente separadas para cada técnica e situação, em  $[m]$  procura-se observar se o ARP permanece estável o mais centralizado possível em relação ao conjunto de linhas de transmissão, a variável

Tabela 7 – Variáveis de Ambiente para teste dos modelos

Fator	Nível	Variável
Brilho (B)	Claro (C) $5 \frac{lumens}{m^2}$	Distância Correta Média [m]
	Escuro (E) $1 \frac{lumens}{m^2}$	
Distância (D)	Perto (P) $5m$	Distância Total Atingida [a]
	Longe (L) $10m$	
Atmosfera (A)	Normal (N) <i>Limpo</i>	Tempo médio de execução individual [t]
	Temporal (T) <i>Chuva e Vento</i>	

[**a**] o quanto o ARP consegue percorrer o corredor antes de perder a visão completa das linhas de transmissão, e [**t**] o tempo médio que uma única inferência leva do recebimento da imagem a definição da ação.

Os níveis procuram identificar em quais circunstâncias determinada técnica possui melhor ou pior resultado. O Brilho<sup>1</sup> (B) influencia diretamente ao horário que determinada inspeção pode ser realizada, já a Distância (D) implica diretamente no campo de visão necessário na obtenção das imagens do corredor, e por fim, a Atmosfera (A) intervem na possibilidade de operacionalização em casos de alteração climática.

#### 4.9.2 Algoritmo de Controle em Pares de buffer

Foi desenvolvido um algoritmo para controle dos ARPs de forma colaborativa discreta, seu pseudocódigo é apresentado abaixo. Assim é garantido a execução em sistema de enxame, onde cada ARPs tem sua função e atuam em conjunto.

Esse algoritmo principal em Algoritmo 1 é executada para cada solução proposta separadamente, inicialmente os pacotes necessários são carregados e inicializados, um laço de controle é executado até que a situação final seja atingida, podendo ser uma quantidade de torres extensão em metros, posição de GPS entre outras, os resultados do percurso são apresentados no fim do programa.

O algoritmo de controle de buffer em Algoritmo 2, é a função chamada por cada ARP no programa principal. Seu objetivo consiste em garantir que o ARP identifique e realize uma ação, para que o outro ARP possa também realizar.

<sup>1</sup> Unidade de medida óptica de fluxo luminoso emitido por um ponto luminoso em um ambiente

---

**Algorithm 1** Controle Principal do Sistema

---

```
1: ... ▶ Pacotes
2: buffer_cima ← []
3: buffer_lateral ← []
4: ARP_cima ← NovoDrone()
5: ARP_lateral ← NovoDrone()
6: ... ▶ Inicializações
7: while controle = True do ▶ Percurso
8:   buffer(ARP_cima,ARP_lateral,buffer_cima,buffer_lateral)
9:   buffer(ARP_lateral,ARP_cima, buffer_lateral, buffer_cima)
10: ... ▶ Resultados
```

---

**Algorithm 2** Controle de *Buffer*

---

```
1: function BUFFER(ARP1, ARP2, buffer1, buffer2) ▶ Cooperação
2:   realizado ← Falso
3:   while realizado = False do
4:     inferencia(ARP1, buffer1)
5:     if tamanho(buffer1) > 0 and tamanho(buffer1) ≥
       tamanho(buffer2) then
6:       acaoReal(ARP, buffer1[tamanho(buffer1) – 1])
7:       realizado ← True
```

---

O algoritmo simplificado de inferência é apresentado em Algoritmo 3, a função é responsável por determinar a ação que o ARP realizará a partir da última imagem obtida por ele, e o tipo de solução que deseja ser utilizada, caso o haja sucesso na identificação, a ação é executada e armazenada em memória temporária (*buffer*).

---

**Algorithm 3** Identificação da Ação e Armazenamento

---

```
1: function INFERENCIA(ARP, buffer) ▶ Ação
2:   movimento ← identificar_movimento(ARP, 'Tipo de Solucao')
3:   if movimento not NULL then
4:     acaoReal(ARP, movimento)
5:     buffer.push(movimento)
```

---

### 4.9.3 Comparação dos Métodos

Nesta subseção são apresentados os resultados comparativos das técnicas selecionadas, tanto para o ARP superior quanto para o lateral, nas técnicas de detecção

de bordas, aprendizado profundo e por reforço. Cada um dos dados de experimento são obtidos a partir de 10 execuções de determinado método e variável de observação.

Na tabela 8 são relacionados o tempo médio de treinamento e execução para cada método desenvolvido para solução. O tempo de treinamento é compreendido do início até o encerramento do treinamento ou definição de parâmetro (no caso do EDs) em cada modelo sugerido, ou seja, o tempo para que o modelo definido esteja pronto para o uso no algoritmo principal. O tempo de execução é o intervalo entre o recebimento da imagem até a definição da ação para utilização do ARP no simulador.

Tabela 8 – Comparação do Tempo médio de treinamento e execução

Método	Treinamento (horas)	Execução (segundos)
ED	2	0,3
DL Mono classe	20	2,72
DL Multi classe	22	3,10
RL	90	0,87

É possível observar que ED mantém a maior velocidade de execução, devido principalmente a não utilização de uma rede neural. O RL também possui uma execução rápida apesar de possuir uma rede neural, isso se deve principalmente ao fato da estrutura de aprendizado por reforço que permite uma rede simplificada. Já o modelo DL possui uma tempo de execução consideravelmente maior, isso decorre principalmente do tamanho da rede neural, e estrutura de identificação e retorno por caixas delimitadoras.

Já na comparação das variáveis nos experimentos é possível observar que o erro médio no ED e DL é menor quando comparado com o RL, principalmente quando há geração de ruído, como no temporal e escuro, e isso mostra uma baixa taxa de detecção correta. O melhor resultado foi do DL, já que o ED não se adaptou bem na variação de altitude. Esse resultado mostra também que o RL pode ser treinado mais intensamente em situação de instabilidade da imagem recebida, já que geralmente a taxa de aprendizagem usada na rede neural profunda é baixa, enfatizando o treinamento a longo prazo. Presume-se que os resultados obtidos no DL pode refletir elementos externos que interferem na identificação da ação como:

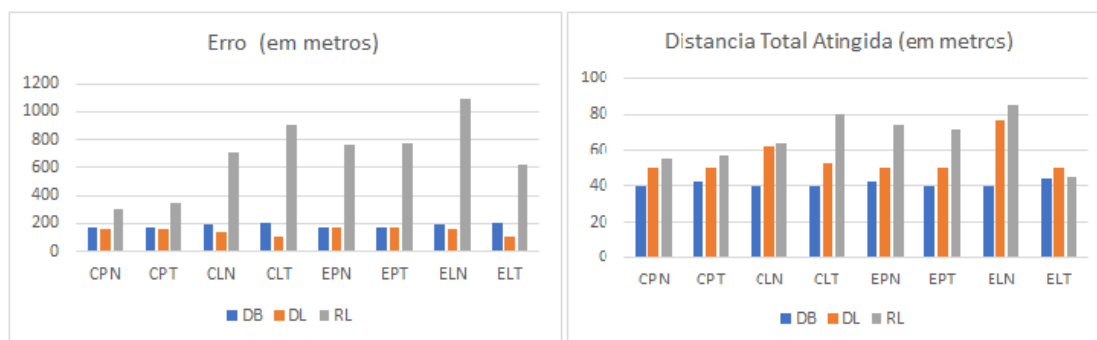


Figura 37 – Gráficos das variáveis analisadas. Fonte: Próprio Autor

trilhas, meio fio, estradas, objetos, entre outros.

Quanto a distância total atingida o DL e ED obtém resultados semelhantes, com uma pequena vantagem do DL, nos mesmos níveis citados anteriormente, isso se dá pelo fato da capacidade de generalização da técnica empregada. O melhor resultado é obtido pelo RL na maior parte dos níveis, entende-se que isso ocorre por conta do treinamento por reforço, que evita as piores recompensas, obtidas principalmente ao sair do campo de visão das linhas. Esse fato mostra que são necessárias diferentes restrições de limites, e um treinamento com imagens obtidas em campo para ser possível observar uma real generalização do rastreamento de longo percurso do corredor de linhas de transmissão.

#### 4.9.4 Experimento Externo

Para o experimento externo a pretensão foi verificar se o comportamento dos ARPs se manteriam iguais aos da simulação. O equipamento para execução em campo é mais limitado, sendo um computador com processador Intel(R) Core(TM) i2-4770 de 2.40GHz, com 4GB DDR2 SDRAM instalada, sistema operacional Linux Ubuntu 16 de 64 *bits* e sem placa de vídeo dedicada.

##### 4.9.4.1 Drone Real

Primeiramente, foi feita uma análise da hélice entre as opções compatíveis com o motor, para a seleção de qual era a mais adequada para a aplicação da solução.

Todas as hélices foram avaliadas comparando-se a relação PD (*Pitch Diameter*) e os parâmetros reais de desempenho de eficiência da hélice e eficiência geral do motor estimados, a figura 38 apresenta esses resultados.

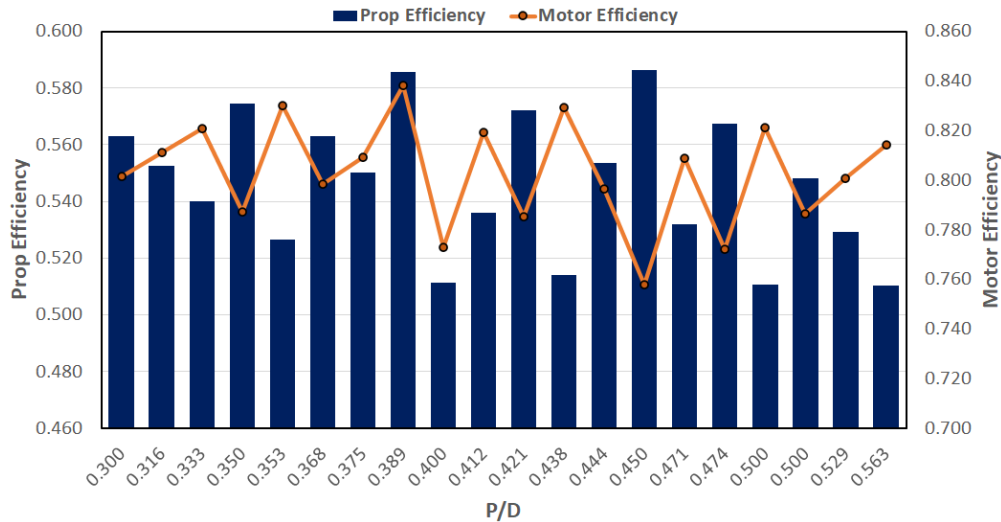


Figura 38 – Eficiência mecânica do motor e da hélice baseado no diâmetro da hélice. Fonte: (DANTAS et al., 2021)

Com os resultados apresentados acima, a hélice escolhida tem 3,5 polegadas de *Pitch*. Esta hélice garantiu uma relação ideal entre a energia mecânica e elétrica mantendo a eficiência. Como afirmado antes, o cálculo de carregamento do disco foi executado e  $95,6N/m^2$ . Para a avaliação do empuxo estático, os parâmetros apresentados foram avaliados para toda a gama de operações, incluindo pairar e condições máximas (DANTAS et al., 2021). Os itens definidos são detalhados no Apêndice B seção 7.

A solução necessita enviar comandos repetidamente para mudança de posição constante do ARP, para reduzir riscos de colisões e garantir boa estabilidade durante o voo, é necessário um processo de calibração, essencial principalmente em situações de rajadas de vento fortes.

A cada novo voo apresentado no anexo B 8.3 os parâmetros de bússola, giroscópio, acelerômetro e horizonte calibrados são calibrados utilizando o software de controle de solo de forma visual e intuitiva. A cada nova calibração o PID

é revisado com incremento/decremento de 5% conforme mostra a Figura 39, os valores finais definidos manualmente e adaptados de acordo com o comportamento observado do ARP é apresentado na tabela 9.

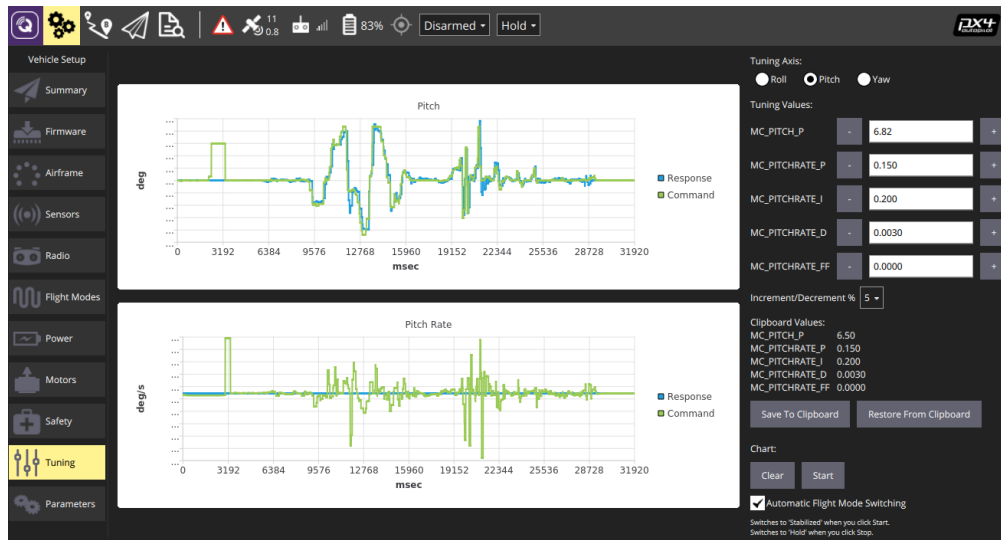


Figura 39 – Tela do QGroundstation para ajuste de PID

Tabela 9 – Parâmetros finais obtidos para ajuste de PID

Eixo	P	P Avaliado	I Avaliado	D Avaliado	FF Avaliado
<i>Roll</i>	6.82	0.150	0.010	0.001	0.000
<i>Pitch</i>	6.82	0.150	0.010	0.001	0.000
<i>Yaw</i>	2.8	0.200	0.100	0.000	0.000

#### 4.9.4.2 Execução do Algoritmo

O algoritmo foi executado em ambiente real no trajeto que pertencente a torre A no percurso apresentado na Figura 36, por uma distância de 22 metros e 19 segundos, utilizando o modelo SSD Mobilenet v1 FPN Multiclasse, em um trecho não pertencente ao *Dataset* de treinamento. A figura 40 mostra esta concretização, onde a estação de solo aparece no canto superior esquerdo, uma imagem obtido do ARP durante execução no canto inferior esquerdo, além de, uma das identificações superior e lateral à direita da imagem.

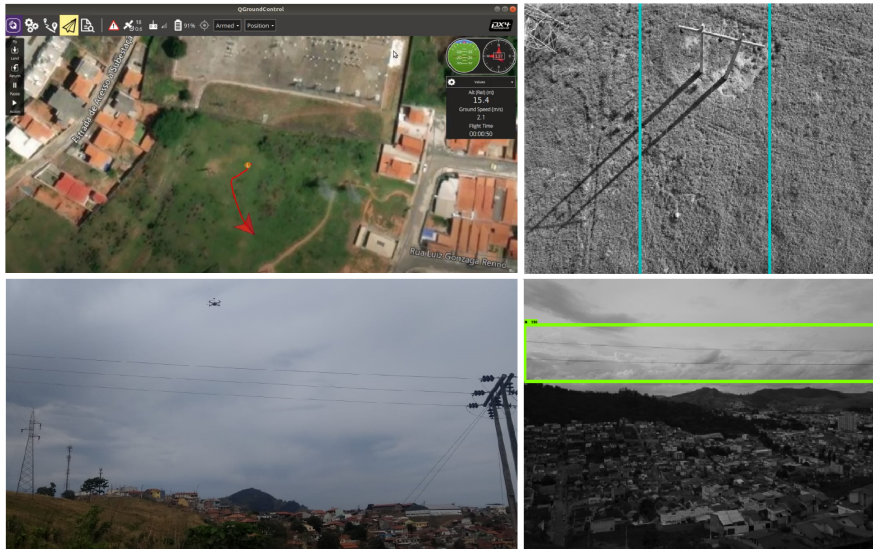


Figura 40 – Imagens da execução em campo. baixo Fonte: Próprio Autor

O teste também foi comprometido pelo clima, devido ao tempo fechado e com vento forte, contudo apesar de curto espaço e tempo foi o suficiente para mostrar uma pequena amostra do método de detecção para uso em ambientes reais. Ao final do período de pesquisa, os resultados obtidos foram mais exitosos no ambiente simulado, mas apresentam grandes possibilidades futuras no ambiente real com a aplicação das sugestões que seguirão na próxima seção.

O sistema final foi implementado em Python, utilizando o *framework* MAVSDK, e as inferências são realizadas com modelo de multi classe SSD Mobilenet v1 FPN. Esse resultado também deve ser capaz de atuar em ambientes e torres semelhantes das apresentadas no estudo, de acordo com os princípios definidos no modelo de DL. Além disso, há uma expectativa de atuação em tempo real, quando aplicado a um computador de bordo embarcado no próprio ARP para o processamento das imagens. Os artefatos desenvolvidos foram disponibilizados publicamente no github (DANTAS, 2021).

# 5 Conclusão

As soluções apresentadas neste estudo apuram algoritmos de visão computacional, pouco abordados na forma aplicada, para a função de rastreamento autônomo de linhas de transmissão de energia em torres de alta tensão.

Os artefatos produzidos, como os *Datasets* e Simuladores, refletem no ambiente computacional, de forma realística, o desafio imposto nas inspeções de grandes extensões de corredores de transmissão de energia por todo o Brasil.

Todas soluções obtiveram resultados satisfatórios a um custo baixo, empregando apenas dois ARPs equipados com câmera frontal, e um computador para processamento das imagens. Destacando-se o aprendizado profundo que apresenta os melhores resultados, e permite um uso generalizado em outros ambientes e estruturas.

## 5.0.1 Limitações do estudo

O ARP é capaz de realizar o rastreamento na posição superior das linhas com a câmera apontada para baixo, ou nas laterais com a câmera apontada para a lateral oposta a sua posição relativa na torre, conforme é apresentado na seção Modelagem 39. Outras posições como diagonal, em giro na parte superior, ou na parte inferior da torre, não foram abordadas neste estudo.

A distância do ARP em relação às linhas tem um alto impacto na detecção do modelo, não sendo recomendada uma distância acima de 10m na parte superior da estrutura alvo até o ARP.

As estimativas do controlador PID estão relacionadas restritamente ao ARP construído, incluindo suas peças auxiliares acopladas à estrutura como trem de pouso e suporte para a câmera, conforme apresentado no Apêndice B seção 7.

Embora exista possibilidade de uso do computador para processamento, existe um *delay* e limitação de distância do ARP à estação de trabalho, por conseguinte, recomenda-se o uso a bordo ou utilizando equipamentos de maior processamento e alcance.

## 5.0.2 Principais Dificuldades

Para treinamento do aprendizado profundo, encontrar um conjunto de dados que contivesse as imagens necessárias para o trabalho mostrou-se uma dificuldade, à vista disso, foi desenvolvido um *dataset* próprio.

O treinamento nos modelos de que possuíam DL são muito dispendiosos, sobretudo o DRL, pois cada modelo necessitava entre 24 e 90 horas para ser treinado.

A implementação de um sistema autônomo de processamento de imagem e tomada de decisão depende muito da atuação de periféricos e *frameworks*, como captura dos dados de imagem, controle de hardware, algoritmos de processamento, entre outras características externas que influenciam diretamente na implementação final. O desafio de construir e desenvolver equipamentos com baixo custo, tem impacto direto nessas questões.

Isto posto, existe ainda uma latência entre a detecção do modelo, atualização dos valores, e execução das ações, pois todos os quadros de imagem foram enviados via transmissor de vídeo para uma estação terrestre, onde o algoritmo foi armazenado, e, após o processamento das imagens, os valores calculados foram enviados de volta para o ARP por telemetria, o que retarda o movimento a cada detecção.

## 5.0.3 Principais Contribuições

Em síntese, este trabalho demonstra a viabilidade de aplicar algoritmos de enxame cooperativos, com inferências em imagens utilizando aprendizado profundo para rastreamento autônomo de linha de transmissão de energia. Essa implementação, porém ainda é inicial e muitas melhorias ainda podem ser feitas no futuro.

A revisão bibliográfica deste trabalho fornece uma vasta modelagem e demonstração das possibilidades de um sistema semelhante ao aqui implementado por meio de uma RSL, com critérios metodológicos bem definidos, resultando em uma análise com materiais relevantes de diferentes estudos sobre rastreamento autônomo de linhas de transmissão de energia.

O estágio de curta duração no exterior na Suécia, promovido pela Unifei e MDH foi de grande valia, pois o aprendizado permitiu a produção de um *dataset* próprio, um algoritmo para processamento de imagem, e a integração desse algoritmo com o ARP para identificação e rastreamento de anomalias em solo.

Com um modelo bem definido a ser implementado, o mérito desse trabalho consistiu em avaliar métodos de detecção para identificar ações discretas de rastreamento de linhas de transmissão de energia. São os algoritmos de ED; DL mono classe; DL multi classe e DRL, que produziram um resultado com futuro promissor.

Além dos algoritmos estudados, foram criados ambientes complexos de diferentes torres e ecossistemas para simulação utilizando uma plataforma de criação 3D avançada com visuais foto realistas. Esses ambientes possibilitaram a validação dos melhores modelos propostos para cada algoritmo estudado.

Ademais, foram desenvolvidos dois *Datasets* robustos com uma coleção de imagens reais e de simulação, que contemplam diferentes ambientes, torres de transmissão e linhas de energia. O primeiro é composto somente por identificação de linhas do tipo *monoclasse*, e o segundo para ações a serem tomadas do tipo *multiclasses*. As imagens originais foram rotuladas manualmente, e disponibilizadas com anotação de nível de pixel no formato TFRecords.

A atuação em modo de enxame permitiu uma inspeção mais precisa e detalhada, mantendo o baixo custo, evidenciando a relevância do trabalho, principalmente em um país com grandes extensões e recursos financeiros limitados no que tange os investimentos em infraestrutura.

## 5.1 Trabalhos Futuros

Como melhorias e trabalhos futuros, sugere-se:

- Incorporação de técnicas ROI (Regiões de Interesse) sistemáticas nos algoritmos de ED, para definir de forma mais precisa os segmentos de linhas;
- Incorporação do *dataset* com novas imagens de outros ambientes complexos ou outros *Datasets*, para aprimoramento dos modelos de DL;
- Exploração de algoritmos de DRL ações contínuas e novas restrições de recompensa/finalização de episódio;
- Implementação do computador de bordo dedicado no experimento em ambiente externo;

## 5.2 Considerações Finais

Todo o estudo desenvolvido neste trabalho mostrou-se relevante e desafiador, realçando o futuro promissor dos ARPs. A experiência foi única, principalmente devido ao aprendizado e as dificuldades enfrentadas no decorrer do estudo. Cabe ainda ressaltar, a importância da realização do estágio de curta duração no exterior proporcionada pela parceria entre a MDH e a Unifei. Por fim, utilização dos ARPs nesses segmentos permitem o dispêndio dos humanos nas tarefas massivas e repetitivas, para um foco em processos que dependem de cognição superior.

# 6 Apêndice A - Revisão Sistemática da Literatura

## 6.1 Definição da pesquisa

A elaboração da questão é de suma importância para o desfecho do RSL. As respostas a essas questões possibilitam a análise específica para determinar o foco inicial da pesquisa e possibilitar a contribuição para o meio acadêmico. Assim, fica definida a seguinte QP (Questão Principal):

*Quais as melhores técnicas utilizadas para rastrear linhas de transmissão de forma automatizada, segura e de baixo custo? O QP busca obter a melhor forma de realizar o rastreamento da linha de transmissão e, para isso, leva em consideração aspectos importantes como eficiência, segurança e custo.*

A partir da QP, as QE (Questões Específicas) que seguem são definidas de forma secundária:

- QE1: Qual o custo de implantação dos materiais e técnicas utilizadas para fiscalizar as linhas de transmissão?
- QE2: Qual a eficiência computacional e energética das técnicas utilizadas para guiar o ARP?
- QE3: Qual o nível de segurança e as localizações das soluções implementadas para rastreamento de linhas de transmissão?

## 6.2 Termos de pesquisa

No contexto do objetivo da pesquisa, este estudo definiu uma estratégia para construir o SS que é representada geralmente pelas palavras-chave relacionadas às questões de pesquisa e que podem ser utilizadas na pesquisa. A estratégia incluiu:

- Identificação de palavras-chave considerando a questão principal de pesquisa;

- Identificação de palavras-chave considerando o artigo controle originado do grupo de pesquisa (RIBEIRO et al., 2018);
- Sinônimos identificados no contexto;

A expressão de pesquisa principal precisa de ajustes específicos na SS para atender aos critérios de condução. Após a reaplicação da estratégia de construção e refinamento do SS a expressão final utilizada foi:

**(“Autonomous” OR “Self-Navigation” OR “Self Navigation” OR Automatic OR AutoPilot) AND (“Remotely Piloted Aircraft” OR RPAS OR ARP OR “Unmanned Aerial Vehicle” OR ARP OR UAS OR DRONE OR “unmanned aerial system” OR Airplane) AND (Imaging OR Inspection OR Maintenance) AND (“Transmission Power” OR “Transmission Line” OR “Power line transmission” OR “Power Network” OR “Power line inspection” OR “Power line”)**

### 6.3 Critérios de seleção e qualidade

A fim de subsidiar a seleção de estudos relevantes que se alinham no esclarecimento das questões propostas de pesquisa, foram definidos CI (Critérios de Inclusão) e CE (Critérios de Exclusão). A inclusão do estudo foi determinada pela relevância, análise do título, resumo, introdução, finalização e quando necessário pela leitura completa do estudo. Os critérios adotados para exclusão e inclusão dos artigos são apresentados nas Tabelas 10 e 11.

Após os critérios de inclusão e exclusão estabelecidos, a próxima etapa foi definir os critérios de qualidade. Para analisar a qualidade dos estudos primários, uma lista de cinco CQ (Critérios de Qualidade) para abordar o artigo geral foi definida e é apresentada na Tabela 12 com base nos estabelecidos por Kitchenham em (KITCHENHAM, 2004).

No critério de nota, para avaliação dos artigos, utilizou-se a escala de pontuação de 0 a 5, sendo que para cada critério atendido integralmente foi adicionado 1 ponto, se parcialmente 0,5 pontos e se não atendeu ao critério nenhuma pontuação foi adicionada. Para os critérios de custo, foi estabelecida uma avaliação qualitativa dos materiais utilizados, uma vez que nem todas as informações numéricas estão acessíveis nas publicações. A escala vai de muito baixa a muito alta. Os ARPs

Tabela 10 – Critérios de Exclusão

CE1	Publicações que não contenham as palavras-chave exatas ou variações delas serão descartadas.
CE2	Publicações que não usam ferramentas de aeronaves não tripuladas serão descartadas.
CE3	Publicações que não tenham o objetivo de monitorar linhas de transmissão serão descartadas.
CE4	Publicações que não usam o modo autônomo serão descartadas.
CE5	Publicações que não possuam todos os processos da obra ou que façam parte de uma obra serão descartadas.
CE6	Publicações que estiverem apenas em planejamento ou que não tenham sido aplicadas ao menos no ambiente de simulação serão descartadas.

Tabela 11 – Critérios de Inclusão

CI1	Publicações que usam aeronaves não tripuladas para inspeção em linhas de transmissão podem ser selecionadas de forma autônoma.
CI2	Podem ser selecionadas publicações que possuam diferentes técnicas de rastreamento autônomo de linhas de transmissão.

Tabela 12 – Critérios de Qualidade

CQ1	O estudo foi descrito de forma clara e adequada?
CQ2	Os métodos ou técnicas usados nos estudos primários foram relatados com clareza?
CQ3	O ambiente virtual usado nos estudos primários foi relatado claramente?
CQ4	A proposta do estudo foi avaliada e/ou validada?
CQ5	Os resultados foram claramente declarados?

fabricados do zero e / ou usando apenas um equipamento serão rebaixados de acordo com o custo. Os ARPs de plateleiras e/ou complementos de equipamento serão aprimorados na classificação de custo. Também é avaliado neste critério o custo de manutenção e resistência do equipamento a partir de informações dos próprios autores. Nos demais critérios, é realizada uma análise geral, mensurada a partir das informações disponíveis na própria publicação.

## 6.4 Condução

A aplicação da metodologia apresentada mostra que a pesquisa retornou um total de 223 artigos (destes, 89 artigos do IEEE, 65 artigos da Scopus, 5 da ACM e 64 artigos do Compendex), visto que a união desses artigos teve duplicatas (85) e artigos que não estavam alinhados com as questões de pesquisa (110), foi necessário aplicar um processo de triagem para obter apenas artigos relevantes para a pesquisa.

O passo seguinte foi aplicar os critérios de inclusão e exclusão resultando em 30 artigos no total e por fim uma seleção mais criteriosa foi feita lendo os artigos na íntegra, retirando 4 artigos com conteúdo duplicado, 5 artigos falsos positivos e 6 artigos cujo texto completo era não disponível. Foram obtidos 15 artigos, após a seleção final dos artigos, o procedimento de extração de dados foi realizado a partir da leitura completa de cada artigo. Os dados extraídos seguiram as especificações descritas a seguir:

- Dados da publicação: título, autor(es), data de publicação, veículo de publicação e resumo da publicação;
- Informações gerais: descrição do objetivo do artigo e dos materiais utilizados (para estipular o custo);
- Técnicas: modalidade utilizada para o reconhecimento das linhas de transmissão para a condução do guia;
- Resultados obtidos: para obter as taxas de erro (%), os locais onde foram realizados os testes e o problema de segurança;
- Critérios de qualidade: nota atribuída do III e comentários adicionais de cada pesquisador

Cada processo é realizado por dois ou três dos autores deste artigo, o que permite a revisão das decisões e elimina qualquer viés que possa ter ocorrido. Os resultados deste RSL foram apresentados e analisados a fim de discutir as evidências dos 15 estudos primários que podem responder às questões de pesquisa levantadas. Conforme processo mostrado na figura 41, o banco de dados IEEE e

Scopus teve uma alta representação com nove e oito estudos incluídos, respectivamente, pois para Compendex foram cinco estudos primários e o ACM teve menos representação sem nenhum estudo selecionado.

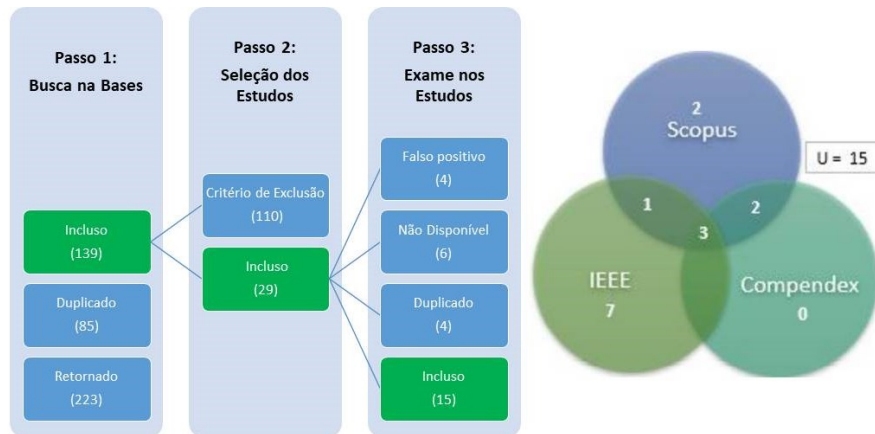


Figura 41 – Processo de seleção de estudos primários (esquerda) e Filtros em cada mecanismo de busca (direita). Fonte Próprio Autor

## 6.5 Análises

A maioria dos artigos estudados propôs o uso do ARP em conjunto com a câmera para inspeção, e são fabricados pela própria equipe, o que por si só proporciona um custo reduzido e melhor se adapta às necessidades do pesquisador. Em (HUI et al., 2019), (WU et al., 2019) e (ZHOU et al., 2016), alternativamente ARPs pré-fabricados foram adaptados e podem gerar um custo maior, principalmente em manutenção. Já em (ZHANG et al., 2016), (DENG et al., 2016), (HAMELIN et al., 2019) e (XIE et al., 2017) foi utilizado, além do material básico, o LiDAR (*Light Detection and Ranging*), uma tecnologia óptica de laser pulsado que mede a reflexão da luz para obter distância dos objetos, e isso aumenta consideravelmente o custo da solução, mas também, aumenta a precisão do rastreamento da linha de transmissão de energia.

Em (DENG(B) et al., 2014) o rastreamento foi realizado usando apenas GPS e IMU, no entanto, o artigo não garante rastreamento preciso, pois há influência

no efeito da exposição de campos próximos a torres de transmissão de energia.

Já em (ZORMPAS et al., 2018), (ZHOU et al., 2016), (MENENDEZ et al., 2016), (GERKE; SEIBOLD, 2014) e (ARAAR; AOUF, 2014) o método usado para identificação das linhas é o filtro Sobel, um detector de gradiente de borda, que filtra uma imagem sARPizada com o núcleo pré definido na horizontal e vertical para obter a primeira derivada, e então atravessa os *pixels* com um vetor gradiente que calcula a cor e intensidade na identificação da borda. Em (ZORMPAS et al., 2018) e (MENENDEZ et al., 2016) o método usado foi a TH com redução o ruído obtendo melhores resultados que o Sobel. Em (ZORMPAS et al., 2018) e (ZHOU et al., 2016) também é utilizado ROI para reduzir a borda da imagem em um determinado objeto, eliminando partes que não deveriam ser processadas.

Alguns artigos usam métodos adicionais para melhorar a precisão ou o desempenho dos algoritmos. Em (HUI et al., 2019), o algoritmo KCF foi usado para rastreamento entre torres após a localização das linhas, este método usa propriedades de matriz rolante para aumentar a velocidade de processamento. Esta técnica é conhecida por ser eficiente, mas dispendiosa principalmente para rastreamento linhas. Em (WU et al., 2019) e (GAO et al., 2018) foi utilizado sensores magnéticos, onde os parâmetros de posição e corrente das linhas são reconstruídos a partir dos dados do campo magnético, e um algoritmo com modelo de aproximação realiza as detecções, o desenvolvimento e uso de laser tem um alto impacto no custo final.

Apenas em três artigos a eficiência energética é abordada, (DENG(B) et al., 2014) integra diferentes tipos de ARPs que auxiliam no rastreamento de forma indireta em locais pré definidos, já os autores em (HAMELIN et al., 2019) e (CHANG et al., 2017) propõem o uso de fio terra suspenso, OGW (*Over Ground Wire*), aumentando consideravelmente o alcance por não exigir muito esforço para manter a altitude. O problema nesses casos é que muitas torres não possuem este tipo de instalação acessível para uso.

## 6.6 Discussão

Abaixo serão discutidos os resultados e as três principais questões de pesquisa:

QE1: Embora os artigos não apresentem valores atuais utilizados nos projetos, é possível ter noção do custo de implementação para a solução realizada. O custo de

manutenção diminui ao usar materiais feitos pelo próprio. Artigos que usam poucos equipamentos, principalmente uma câmera e um sensor, são baratos, enquanto aqueles que usam uma variedade de equipamentos de última geração têm um custo mais alto e os resultados, eles obtêm, são semelhantes a esses.

QE2: Todos os artigos a técnica é considerada viável e possui métodos, locais e equipamentos diferentes não sendo possível comparação direta. Para estipular a eficiência da técnica utilizada foram consideradas as informações do próprio autor. De forma geral, a taxa de erro, quando apresentada, fica em torno de 10% demonstrando assim, considerando as técnicas empregadas como eficientes. Porém, a questão energética foi pouco discutida, em geral, o VANT possuía características padrão de autonomia.

QE3: Devido à baixa taxa de erro e às simulações ou experimentos realizados, presume-se que os ARPs são seguros para aplicação, mas uma abordagem específica para localização e segurança não foi apresentada nos artigos. Em geral, os ambientes de uso ficam em locais remotos e de pouca circulação de pessoas, mas na prática existem linhas de transmissão que passam no meio de grandes centros. Além disso, nenhum artigo teve relação direta com a segurança de suas implementações e equipamentos.

## 7 Apêndice B - Especificações do ARP utilizado no estudo

A tabela 13 apresenta a relação de itens do ARP, e a montagem é apresentada na Figura 42 relacionando pela numeração correspondente a cada item.



Figura 42 – Imagem real do ARP utilizado no estudo. Fonte: Próprio Autor

Além dos itens informados, no interior da Pixhawk possui um dispositivo IMU que agrega vários sensores, como Barômetro/Altímetro, ferramenta que permite ler a pressão do ar e estimar a altitude atual; Acelerômetro um dispositivo com sensores que, quando deslocados de cima para baixo ou para os lados, no espaço de eixo 3D, percebe a direção e velocidade do deslocamento; Giroscópio que mede a velocidade angular e a direção em que o aparelho está girando; e a Bússola que lê as variações no campo magnético da terra, permitindo corrigir as inclinações.

Tabela 13 – Componente para operação do ARP

	Item	Descrição Resumida
a	Estrutura F450 Genérica em X de 450mm no material Polipropileno	Base física (chassi) para fixação de todos os componentes necessários para operação do ARP.
b	Bateria LiPo 14.8v 30c 4S 5200mAh	Fornece a energia para funcionamento dos componentes eletro eletrônicos embarcados.
c	Módulo de Energia XT60 6 a 12 células	Fornece energia a todos os componentes a partir bateria, e possui controlador para medições de consumo, corrente e tensão da bateria.
d	Motores A2212 920kv ReadyTosky sem escova	Combina passos em alta velocidade para girar as hélices com alta eficiência.
e	Pás Hélice 3.5 polegas no material Polipropileno	A hélice de são asas rotativas fixadas por pás nos motores, que produzem um efeito aerodinâmico de deslocamento e sustentação por propulsão ao “cortarem” o ar, com baixa pressão no dorso da pá e uma alta pressão na face.
f	ESC Simonk 30A	Mecanismos que transformam os pulsos PWM recebidos da controladora de voo em sinais e energia controlada para rotação dos motores.
g1	Rádio Controle Flysky/FS-i6	Permitem controlar o voo manualmente por um operador no rádio controle TX a partir do solo ligado a um canal da frequência de 2,4Ghz.
g2	Receptor Controle Flysky/FS-i6	Receptor RX embarcado no ARP que recebe informação de comandos na frequência de 2,4Ghz.
g3	Decodificador PPM Atmega328p	Decodifica os comandos do receptor e os codifica em um pulso PPM apropriado para as controladoras de voo.
h1	Telemetria Readytosky 433 ou 915 MHz	Sistema de comunicação em duas vias por radio frequência para troca de dados a partir do ARP.
h2	Telemetria Readytosky USB 433 ou 915 MHz	Sistema de comunicação em duas vias por radio frequência para troca de dados a partir da estação de solo.
i	Módulo GPS Externo NEO-M8N	Obtém e transmite a controladora de voo dados de GPS e bússola própria.
j1	Transmissor de Vídeo TS832	Transmite o vídeo obtido por uma câmera analógica por meio de frequência 5,8 GHz.
j2	Receptor de Vídeo RC832	Recebe o vídeo transmitido por meio de frequência 5,8 GHz.
k	Câmera Analógica Lente 25mm Resolução 1000TV	Obtém 30 imagens capturadas do ambiente com taxa de 30 quadros por segundo.
l	Controladora de Voo Pixhawk PX4 2.4.8	É um piloto automático que permite a estabilização da aeronave e controles de voo, a partir de sensores embutidos.

# 8 Apêndice C - Resumo do relatório de estágio no exterior

## 8.1 Introduction

Capturing images of the earth's surface is increasingly gaining importance, particularly for the analysis of anomalies in the soil for various purposes, such as geospatial studies, fires, natural disasters, agriculture, etc. One of the most relevant scenarios is the search and identification of people who are lost or at risk, they are called SAR activities, and today they are now commonly assisted by information technology (POTEYEVA et al., 2007).

These systems of support for search and rescue activities, known as SARUAV (*Search And Rescue Unmanned Aerial Vehicles*), have different obstacles to be faced in the current scenario. Adverse conditions stand out in certain regions, the area covered by the UAV camera and the change in lighting, are imposed for high-performance human detection.

## 8.2 Methodology

To develop a decision support system in the search and rescue of people, it is necessary to approach the reality of the professionals involved. Thus, it was proposed to create a dataset with the most different situations. The images were obtained from a DJI Mavic drone with a simple 1/2.3 inches camera, up to 12.35M pixels, and 78.8° 26 mm field of view lenses, Distortion <1.5% and Focus from 0.5 m. Which are considerably less expensive than UAVs with multispectral cameras.

The first set featured images of a volunteer performing positions in different ways: arms open as if he were asking for help; sitting as if he were at rest; and performing ok gestures as if he were dispensing help. The second in the set of images, on the other hand, used clothes of different colors scattered in the UAV's area of operation. So that it could simulate people lying on open fields, shadows

or corners. Finally, the identification points were monitored using the UAV, so that this third set of images could determine the path taken by an individual in this environment.

These image sets will undergo manipulation processing, and by the recognition techniques presented, their effectiveness will be evaluated by precision 4.1 and recall 4.2 considering true positives, false positives and false negatives. The processing speed will be evaluated using the runtime and image flow indicators.

This process will allow you to receive the images for an ROI analysis, where a color segmentation or color filtering will be carried out, which helps to identify specific objects or regions with a different color. From the result of the image, it is possible to cut out anomalies in the image, with a certain safety margin, so that the detection algorithm of the specific object is applied.

With classification and location of the object it will be possible to establish the direction and speed of the identified person, from two frames in a row with a controlled interval, a certain time  $t$  will be determined, the distance will be obtained by the distance between the initial pixels of the object.

Thus, considering the height as distance from the drone to the person  $h$  in meters, the focal length  $f$  in meters, and the width of the person in Image  $p$  in pixels, it is then possible to obtain the approximate size of individual  $wh * pf$ , and also estimate the distance between points euclidean distance defined in pixels for meters  $d_m$ . The average speed  $v_m$  is then obtained by this distance by the time difference  $t$  of obtaining the two frames  $vm = \frac{d_m}{t_1 - t_2}$ .

The direction will also use the initial pixels of the detected object to obtain the tangent of the angle  $\theta = atan((y_1 - y_0), (x_1 - x_0))$  and then relate to the cardinal and collateral points. In addition to the image information, data from the GPS can also be used to determine the individual's real position.

### 8.3 Provisional Results

With the development of the study, it is expected to develop a specialist system for decision making in the identification, search and rescue of people at risk, based on images from cameras embedded in UAVs. This system will be based on the processed information of the person identified in the images, including speed,

direction and position of the people. This information will be obtained when anomalies are identified in the images and classify them using machine learning as people in those images, as shown in Figure 43.

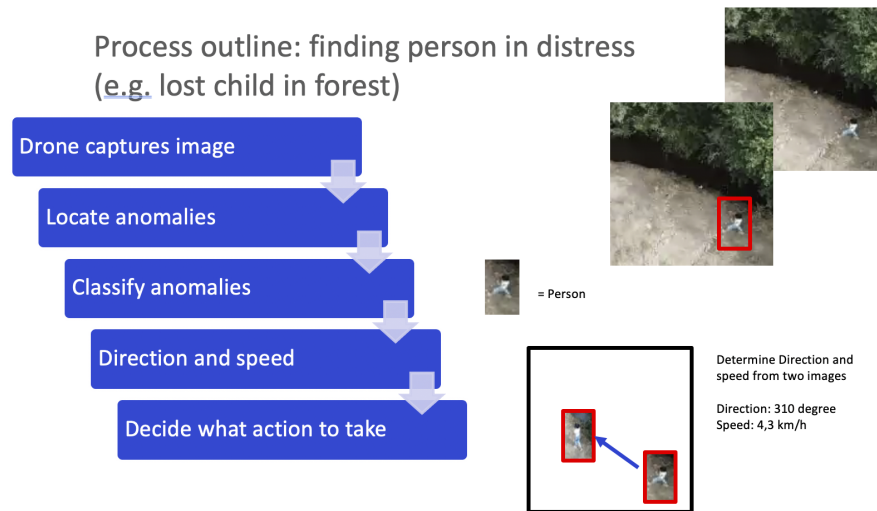


Figura 43 – Process outline of project (adapted from presentation by Peter Funk)

It is also expected to contribute with its own dataset, so an image survey was carried out using UAV for composing the dataset. Through the open source system (NELSON; DWYER, 2020) each person in the image received a mark with the points regarding their location in the image, as well as position information, altitude of the UAV, situation in which they are and camera position at the time of capture, so far there are 231 images and 459 tags.

It is expected to apply segmentation and machine learning to obtain bounding boxes represented by the sub-regions of the original image where the object detection algorithm was applied, and thus obtain the highest probabilities associated with them. A simple conventional neural network, such as MLP, cannot learn complex characteristics, since CNN is a deep learning architecture that can obtain local resources from high-layer inputs and transfer to lower layers to obtain more complex resources (BRITO et al., 2019).

The ResNet (*Residual Neural Network*) is a residual neural network architecture, that is, it has jump connections over the layers, to facilitate the training

of very deep networks. The ResNet-50 architecture has 50 layers, which are less complex and as deep as other current networks like VGG. The ResNet-50 has 850 thousand parameters, and by default this model trains with a decay of 0.0001, and a momentum of 0.9.

The initial indoor design for experimental control and implementation in this project is based on the use of the Crazyflie 2.0 quad UAV, produced by Bitcraze AB was founded in 2011 by quadcopter designers in Sweden (BROWN et al., 2021).

This nano UAV is equipped with four 14000 rpm/V coreless DC motors. Measures only 92 mm between opposing engine axles diagonally and is 29 mm high, the total weight is 27 g, its battery has a 1 cell (3.7 V) LiPo type 240 mAh that provides power for up to 7 minutes of continuous flight. It has communication through telemetry, and allows programming of the solution in Python. It also has a Kalman filter (extended), figure 7, to estimate the current state of the UAV based on received measurements (BROWN et al., 2021).

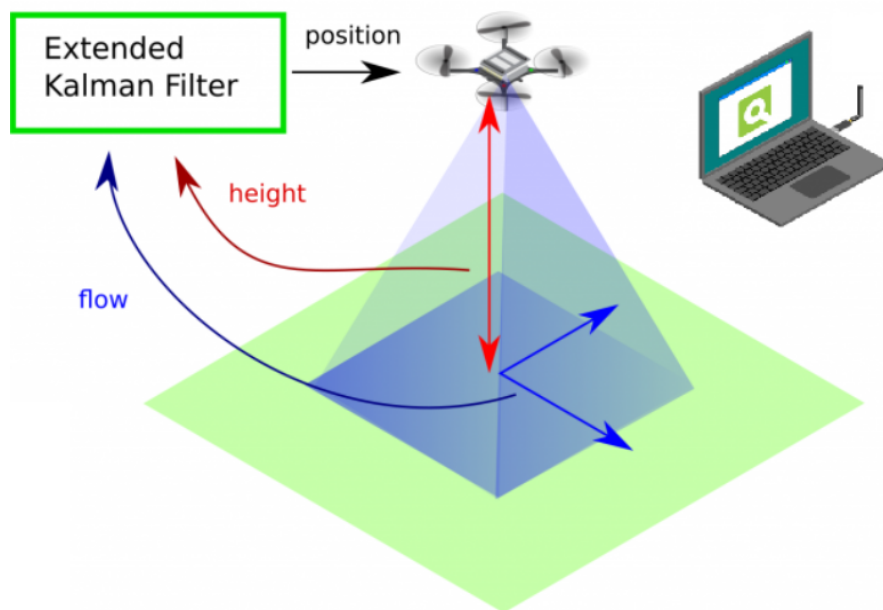


Figura 44 – Communication and relative positioning and internal pose estimation (BROWN et al., 2021)

This report presented a work in progress, which already has a small dataset, an initial basis for training, an architecture for experiments, and mainly a theoretical concept for conducting search and rescue of people using UAV images. It is expected to increase the current dataset with new public images and considerably increase the training time for long- range identification of people and objects. It is also a future objective to improve the tracking algorithm used and obtain more variables that help in the search and rescue of people in emergency situations.

# Anexo A - SISANT da aeronave utilizada no estudo




	<p>REPÚBLICA FEDERATIVA DO BRASIL FEDERATIVE REPUBLIC OF BRAZIL</p> <p>AGENCIA NACIONAL DE AVIAÇÃO CIVIL NATIONAL CIVIL AVIATION AGENCY</p>	
<p><b>CERTIDÃO DE CADASTRO DE AERONAVE NÃO TRIPULADA – USO NÃO RECREATIVO</b> <b>UNMANNED AIRCRAFT INSCRIPTION CERTIFICATE – NON-RECREATIONAL</b></p>		
<p>Esta certidão de cadastro, emitida de acordo com o RBAC-E nº 94, é válida até <b>31/07/2021</b>, salvo em caso de cancelamento, suspensão ou revogação pela Autoridade de Aviação Civil Brasileira.</p> <p><i>This Inscription Certificate, issued in accordance with RBAC-E nr. 94, shall remain valid <b>07/31/2021</b>, unless it is cancelled, suspended or revoked by the Brazilian Civil Aviation Authority.</i></p> <p>Operador (Operator) <b>Antonio Josivaldo Dantas Filho</b></p> <p>CPF (document): 364.710.428-09</p>	<p><b>Nº do cadastro (Inscription Number):</b> <b>PP-201999002</b></p> <p>Uso (Purpose): não recreativo (non-recreational) Ramo de atividade (Business): Treinamento, pesquisa e desenvolvimento Fabricante (Maker): MONTAGEM PRÓPRIA Modelo (Model): F450 Modificado Nº de série (Serial Number): 2019002 Peso máximo de decolagem (MTOW): 3,00 kg Foto (Picture):</p> 	
<p><b>O descumprimento da regulamentação aplicável pode ensejar consequências administrativas, civis e/ou criminais para o infrator.</b></p>	<p>Informações adicionais (additional information): Aeronave utilizada pelos pesquisadores da UNIFEI - IMC no desenvolvimento de algoritmos de guiagem, processamento de imagens e treinamento de pilotos</p>	
<p>O detentor desta certidão de cadastro (o operador), ou aquele com quem for compartilhada sua aeronave, é considerado apto pela ANAC a realizar voos <b>recreativos e não recreativos</b> no Brasil, com a aeronave não tripulada acima identificada, em conformidade com os regulamentos aplicáveis da ANAC. É responsabilidade do operador tomar as providências necessárias para a operação segura da aeronave, assim como conhecer e cumprir os regulamentos do DECEA, da Anatel, e de outras autoridades competentes.</p> <p><i>The holder of this inscription certificate (the operator), or the person with whom this aircraft is shared, is considered apt by Brazilian Civil Aviation Authority to perform <b>recreational and non-recreational</b> flights in Brazil, using the above identified unmanned aircraft, in conformity with the applicable regulations of Brazilian Civil Aviation Authority. It's the operator's responsibility to take the necessary actions to ensure a safe operation, as well as know and comply with the regulations of air traffic control (ATC), telecommunications, and other competent authorities</i></p>		
<p>A validade desta certidão pode ser verificada pelo link <a href="https://sistemas.anac.gov.br/SISANT/Aeronave/ConsultarAeronave">https://sistemas.anac.gov.br/SISANT/Aeronave/ConsultarAeronave</a></p>		
<p>Local e data da emissão (Place and date of issue) <b>Brasília, 18 de julho de 2021</b> <b>Brasilia, July 18th, 2021</b></p>		
<p>Esta certidão de cadastro não é válida para aeronaves não tripuladas acima de 25 kg de peso máximo de decolagem, ou em voos além da linha de visada visual (BVLOS) ou acima de 400 pés ou 120 metros acima do nível do solo (AGL). <i>This inscription certificate is not valid for unmanned aircraft of more than 25 kg maximum takeoff weight, or flying beyond visual line of sight (BVLOS) or over 400 feet or 120 meters above ground level (AGL).</i></p>		

Figura 45 – SISANT da aeronave utilizada no estudo. Fonte: (ANAC, 2021)

# Anexo B - Solicitações de voo usadas no estudo

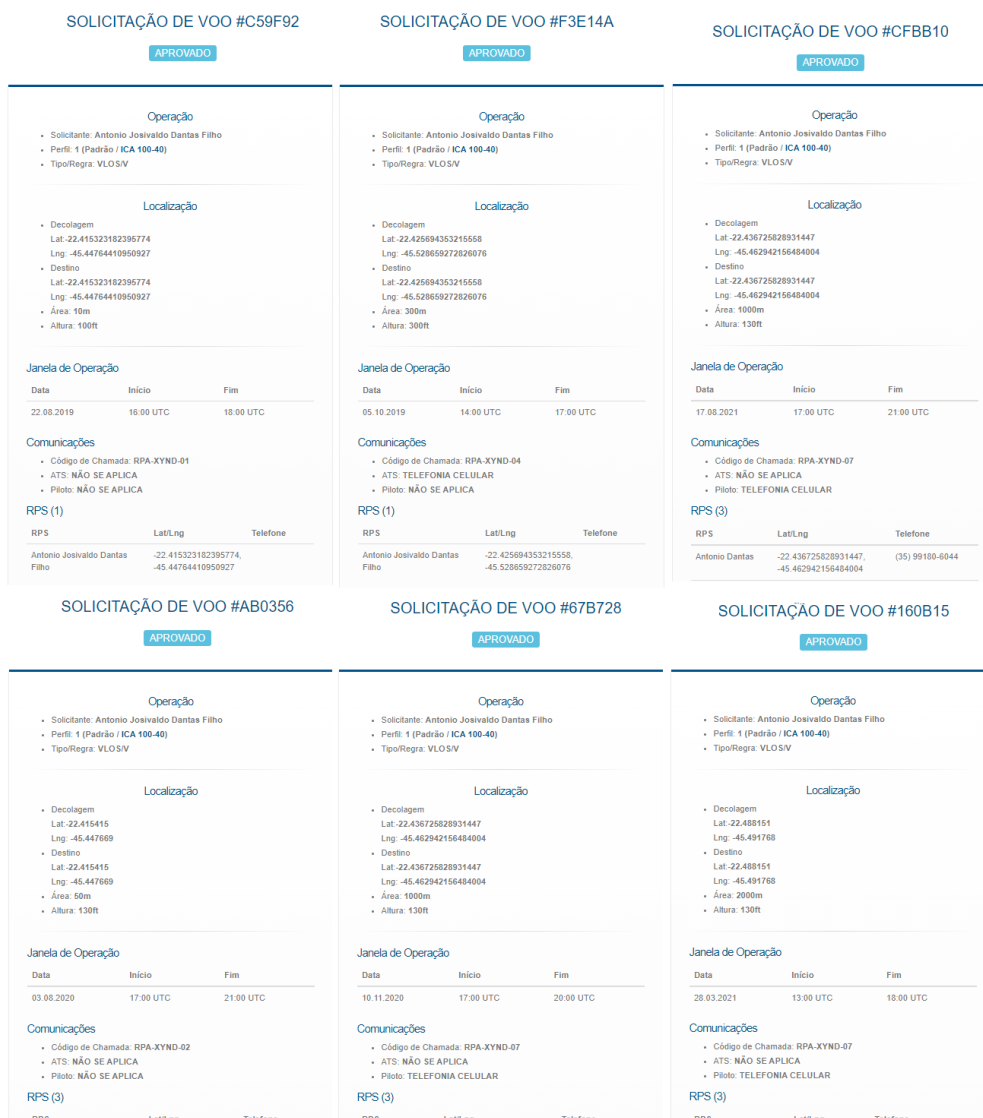


Figura 46 – Solicitações de voo usadas no estudo. Fonte: (VARGAS, 2019)

# Referências

ABADI, M. et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. Software available from tensorflow.org. Available at: <<https://www.tensorflow.org/>>.

AFMAN, J.-P. et al. Towards a new paradigm of uav safety. *arXiv preprint arXiv:1803.09026*, 2018. Available at: <<https://arxiv.org/abs/1803.09026>>.

ANAC. *ANAC emite autorização para drone de inspeção de linhas de transmissão de energia*. 2021. Available at: <<https://www.gov.br/anac/pt-br/noticias/2021/nesta-quinta-feira-10-06-2021-representantes-da-anac-e-da-energias-2013-assessoria-em-sistemas-de-energia-e-comercio-ltda-empresa-detentora-da-autorizacao-participaram-da-solenidade-de-certificacao-da-aeronave-durante-o-evento-o-superintendente-de>>.

AQUINO, T. B. A. C. V. D. *ICA 100-40*. 2019. Available at: <<https://www.decea.gov.br/static/uploads/2015/12/Instrucao-do-Comando-da-Aeronautica-ICA-100-40.pdf>>.

ARAAR, O.; AOUF, N. Visual servoing of a quadrotor uav for autonomous power lines inspection. In: *IEEE. 22nd Mediterranean Conference on Control and Automation*. 2014. p. 1418–1424. Available at: <<https://ieeexplore.ieee.org/abstract/document/6961575>>.

ASCOM, A. de C. S. *Orientações para usuários*. 2017. Available at: <[https://www.anac.gov.br/assuntos/paginas-tematicas/drones/orientacoes\\_para\\_usuarios.pdf](https://www.anac.gov.br/assuntos/paginas-tematicas/drones/orientacoes_para_usuarios.pdf)>.

BIANCHETTI, M. *Cemig instalará 200 subestações com investimentos de R\$5 bi*. 2021. Available at: <<https://diariodocomercio.com.br/economia/cemig-fara-investimento-de-r-5-bi-em-subestacoes/>>.

BICZYSKI, M. et al. Multicopter sizing methodology with flight time estimation. *Journal of Advanced Transportation*, Hindawi, v. 2020, 2020. Available at: <<https://www.hindawi.com/journals/jat/2020/9689604/>>.

BRADSKI, G.; KAEHLER, A. *Learning OpenCV: Computer vision with the OpenCV library*. "O'Reilly Media, Inc.", 2008. Available at: <[https://www.google.com.br/books/edition/Learning\\_OpenCV\\_3/SKy3DQAAQBAJ?hl=pt-BR&gbpv=1](https://www.google.com.br/books/edition/Learning_OpenCV_3/SKy3DQAAQBAJ?hl=pt-BR&gbpv=1)>.

- BRITO, P. L. de et al. A technique about neural network for passageway detection. In: SPRINGER. *16th International Conference on Information Technology-New Generations (ITNG 2019)*. 2019. p. 465–470. Available at: <[https://link.springer.com/chapter/10.1007/978-3-030-14070-0\\_64](https://link.springer.com/chapter/10.1007/978-3-030-14070-0_64)>.
- BROWN, L. et al. Bitcraze crazyflie 2.0, group 15. 2021. Available at: <<http://adamlilja.com/images/quadcontrol.pdf>>.
- CANNY, J. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, Ieee, n. 6, p. 679–698, 1986. Available at: <<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4767851>>.
- CHANG, W. et al. Development of a power line inspection robot with hybrid operation modes. In: IEEE. *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2017. p. 973–978. Available at: <<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8202263>>.
- CHOLLET, F. *Sobre Keras*. 2020. Available at: <<https://keras.io/about/>>.
- DANTAS, A. *Artefatos produzidos na dissertação*. 2021. Available at: <[https://github.com/dantasunifei/artefatos\\_dissertacao](https://github.com/dantasunifei/artefatos_dissertacao)>.
- DANTAS, A. et al. Low-cost uav for medical delivery. *International Journal of Development Research*, v. 11, Mar 2021. Available at: <<https://www.journalijdr.com/low-cost-uav-medical-delivery>>.
- DENG, C. et al. Real time autonomous transmission line following system for quadrotor helicopters. In: IEEE. *2016 International Conference on Smart Grid and Clean Energy Technologies (ICSGCE)*. 2016. p. 61–64. Available at: <<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7876026>>.
- DENG(B), C. et al. Unmanned aerial vehicles for power line inspection a cooperative way in platforms and communications. *J. Commun.*, v. 9, n. 9, p. 687–692, 2014. Available at: <<http://www.jocm.us/uploadfile/2014/0918/20140918110959273.pdf>>.
- DJI. *Naza-M Lite Features*. 2019. Available at: <<https://www.dji.com/naza-m-lite/feature>>.
- DYBA, T.; DINGSOYR, T.; HANSSSEN, G. K. Applying systematic reviews to diverse study types: An experience report. In: IEEE. *First international symposium on empirical software engineering and measurement (ESEM 2007)*. 2007. p. 225–234. Available at: <<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4343750>>.

- EBEID, E.; SKRIVER, M.; JIN, J. A survey on open-source flight control platforms of unmanned aerial vehicle. In: IEEE. *2017 Euromicro Conference on Digital System Design (DSD)*. 2017. p. 396–402. Available at: <<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8049816>>.
- ELIAS, K. M. Torre de transmissão de energia elétrica: novo design e os desafios da inserção no contexto urbano. Universidade Federal de Minas Gerais, 2015. Available at: <<https://repositorio.ufmg.br/handle/1843/BUBD-A5BNFT>>.
- ENDO, Y. et al. Deepprop: Extracting deep features from a single image for edit propagation. In: WILEY MISC LIBRARY. *Computer Graphics Forum*. 2016. v. 35, n. 2, p. 189–201. Available at: <<https://onlinelibrary.wiley.com/doi/full/10.1111/cgf.12822>>.
- ERMAKOV, V. *MAVLink extendable communication node for ROS with proxy for Ground Control Station*. 2020. Available at: <<http://wiki.ros.org/mavros>>.
- FAIR, L. de Pesquisas em IA do F. *Pytorch documentation*. 2020. Available at: <<https://pytorch.org/>>.
- FILHO, A. J. D. et al. A general low cost uav solution for power line tracking. In: SPRINGER NATURE. *17th International Conference on Information Technology–New Generations (ITNG 2020)*. 2020. v. 1134, p. 525. Available at: <[https://link.springer.com/chapter/10.1007/978-3-030-43020-7\\_69](https://link.springer.com/chapter/10.1007/978-3-030-43020-7_69)>.
- FREITAS, V. *Perform Systematic Literature Reviews*. 2019. Available at: <<https://parsif.al/>>.
- GAO, F. et al. A novel inverse method for automatic uav line patrolling with magnetic sensors. In: IEEE. *2018 IEEE International Symposium on Electromagnetic Compatibility and 2018 IEEE Asia-Pacific Symposium on Electromagnetic Compatibility (EMC/APEMC)*. 2018. p. 481–485. Available at: <<https://ieeexplore.ieee.org/document/8393825>>.
- GARAGE, W. et al. *OpenCV*. 2009. Available at: <<https://ci.nii.ac.jp/naid/10030062649/>>.
- GERKE, M.; SEIBOLD, P. Visual inspection of power lines by uas. In: IEEE. *2014 International Conference and Exposition on Electrical and Power Engineering (EPE)*. 2014. p. 1077–1082. Available at: <<https://ieeexplore.ieee.org/document/6970074>>.
- GIOI, R. Grompone von et al. LSD: a Line Segment Detector. *Image Processing On Line*, v. 2, p. 35–55, 2012. <<https://doi.org/10.5201/ipol.2012.gjmr-lsd>>.

- GLOBO, R. T. *G1 Notícia*. 2019. Available at: <<http://g1.globo.com/>>.
- GOUTTE, C.; GAUSSIÉ, E. A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In: SPRINGER. *European conference on information retrieval*. 2005. p. 345–359. Available at: <[https://link.springer.com/chapter/10.1007/978-3-540-31865-1\\_25](https://link.springer.com/chapter/10.1007/978-3-540-31865-1_25)>.
- GUPTA, S. G. et al. Review of unmanned aircraft system (uas). *International journal of advanced research in computer engineering & technology (IJARCET)*, v. 2, n. 4, p. 1646–1658, 2013. Available at: <<https://ieeexplore.ieee.org/document/8798137>>.
- HAMELIN, P. et al. Discrete-time control of linedrone: An assisted tracking and landing uav for live power line inspection and maintenance. In: IEEE. *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. 2019. p. 292–298. Available at: <<https://ieeexplore.ieee.org/document/8798137>>.
- HAQUE, S. R.; KORMOKAR, R.; ZAMAN, A. U. Drone ground control station with enhanced safety features. In: IEEE. *2017 2nd International Conference for Convergence in Technology (I2CT)*. 2017. p. 1207–1210. Available at: <<https://ieeexplore.ieee.org/abstract/document/8226318>>.
- HENTATI, A. I. et al. Simulation tools, environments and frameworks for uav systems performance analysis. In: IEEE. *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*. 2018. p. 1495–1500. Available at: <<https://ieeexplore.ieee.org/document/8450505>>.
- HUAMÁN, A. *Hough Line Transform*. 2019. Available at: <[https://docs.opencv.org/3.4/d9/db0/tutorial\\_hough\\_lines.html](https://docs.opencv.org/3.4/d9/db0/tutorial_hough_lines.html)>.
- HUI, X. et al. A monocular-based navigation approach for unmanned aerial vehicle safe and autonomous transmission-line inspection. *International Journal of Advanced Robotic Systems*, SAGE Publications Sage UK: London, England, v. 16, n. 1, p. 1729881419829941, 2019. Available at: <<https://journals.sagepub.com/doi/full/10.1177/1729881419829941>>.
- HULSE, J. V.; KHOSHGOFTAAR, T. Knowledge discovery from imbalanced and noisy data. *Data & Knowledge Engineering*, Elsevier, v. 68, n. 12, p. 1513–1542, 2009. Available at: <<https://www.sciencedirect.com/science/article/abs/pii/S0169023X09001141?via%3Dihub>>.
- INC, G. *Olá, este é o Colaboratory*. 2019. Available at: <<https://research.google.com/colaboratory/>>.

KIRTLEY, J. L. *Electric power principles: sources, conversion, distribution and use*. John Wiley & Sons, 2020. Available at: <[https://books.google.com.br/books/about/Electric\\_Power\\_Principles.html](https://books.google.com.br/books/about/Electric_Power_Principles.html)>.

KIRYATI, N.; ELDAR, Y.; BRUCKSTEIN, A. M. A probabilistic hough transform. *Pattern recognition*, Elsevier, v. 24, n. 4, p. 303–316, 1991. Available at: <<https://www.sciencedirect.com/science/article/abs/pii/003132039190073E>>.

KITCHENHAM, B. Procedures for performing systematic reviews. *Keele, UK, Keele University*, v. 33, n. 2004, p. 1–26, 2004. Available at: <<https://www.inf.ufsc.br/~aldo.vw/kitchenham.pdf>>.

KITCHENHAM, B.; CHARTERS, S. Guidelines for performing systematic literature reviews in software engineering. Citeseer, 2007. Available at: <[https://www.elsevier.com/\\_data/promis\\_misc/525444systematicreviewsguide.pdf](https://www.elsevier.com/_data/promis_misc/525444systematicreviewsguide.pdf)>.

KOENIG, N. *Gazebo Robot simulation made easy*. 2019. Available at: <<http://gazebosim.org/>>.

KOVALEV, V.; KALINOVSKY, A.; KOVALEV, S. Deep learning with theano, torch, caffe, tensorflow, and deeplearning4j: Which one is the best in speed and accuracy? Minsk: Publishing Center of BSU, 2016. Available at: <[https://elib.bsu.by/bitstream/123456789/158561/1/Kovalev\\_Kalinovsky\\_Kovalev.pdf](https://elib.bsu.by/bitstream/123456789/158561/1/Kovalev_Kalinovsky_Kovalev.pdf)>.

KRUSE, R. et al. *Computational intelligence*. Springer, 2011. Available at: <<https://www.springer.com/gp/book/9781447158493>>.

KUDLUR, M. et al. *TensorBoard: kit de ferramentas de visualização do TensorFlow*. 2020. Available at: <<https://www.tensorflow.org/tensorboard>>.

LIN, C.-T. et al. Real-time single-stage vehicle detector optimized by multi-stage image-based misc hard example mining. *IEEE Transactions on Vehicular Technology*, IEEE, v. 69, n. 2, p. 1505–1518, 2019. Available at: <<https://en.x-mol.com/paper/article/1368356500126982144>>.

LIN, T.-Y. et al. Feature pyramid networks for object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [n.p.], 2017. p. 2117–2125. Available at: <<https://arxiv.org/abs/1612.03144>>.

LIU, W. et al. Ssd: Single shot multibox detector. In: SPRINGER. *European conference on computer vision*. 2016. p. 21–37. Available at: <<https://arxiv.org/abs/1512.02325>>.

LUQUE-VEGA, L. F. et al. Power line inspection via an unmanned aerial system based on the quadrotor helicopter. In: IEEE. *MELECON 2014-2014 17th IEEE Mediterranean electrotechnical conference*. 2014. p. 393–397. Available at: <<https://ieeexplore.ieee.org/document/6820566>>.

MAANYU, K. N.; RAJ, D. G.; CHOUBEY, S. B. A study on drone autonomy. 2020. Available at: <[http://www.ijater.com/Files/5fd7d25b-b3a9-49b8-89fc-5b00f81b082e\\_IJATER\\_51\\_04.pdf](http://www.ijater.com/Files/5fd7d25b-b3a9-49b8-89fc-5b00f81b082e_IJATER_51_04.pdf)>.

MACHADO, G. V. *Resenha Janeiro 21 - Claro Final*. 2021. Available at: <<https://www.epe.gov.br/sites-pt/publicacoes-dados-abertos/publicacoes/PublicacoesArquivos/publicacao-153/topico-574/Resenha%20Janeiro%202021%20-%20Claro%20Final.pdf>>.

MARTINS, W. M. et al. Tracking for inspection in energy transmission power lines using unmanned aerial vehicles: a systematic review of current and specific literature. *IAES International Journal of Robotics and Automation*, IAES Institute of Advanced Engineering and Science, v. 9, n. 4, p. 233, 2020. Available at: <<https://ijra.iaescore.com/index.php/IJRA/article/view/20291>>.

MEIER, L. *Pixhawk Series*. 2019. Available at: <<http://pixhawk.org/>>.

MENENDEZ, O. A. et al. Vision based inspection of transmission lines using unmanned aerial vehicles. In: IEEE. *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. 2016. p. 412–417. Available at: <<https://ieeexplore.ieee.org/document/7849523>>.

MICROSOFT. *Welcome to AirSim*. 2019. Available at: <<https://microsoft.github.io/AirSim/>>.

MILANO, D. de; HONORATO, L. B. *Visao computacional*. 2014. Available at: <[https://d1wqtxts1xzle7.cloudfront.net/35825905/2010\\_IA\\_FT\\_UNICAMP\\_visaoComputacional-with-cover-page-v2.pdf](https://d1wqtxts1xzle7.cloudfront.net/35825905/2010_IA_FT_UNICAMP_visaoComputacional-with-cover-page-v2.pdf)>.

NELSON, J.; DWYER, B. *Roboflow*. 2020. Available at: <<https://roboflow.com/>>.

OBORNE, M. *MissionPlanner*. 2019. Available at: <<https://github.com/ArduPilot/MissionPlanner>>.

OES, J. *Introdução Guia MAVSDK*. 2019. Available at: <<https://mavsdk.mavlink.io/main/en/index.html>>.

ONS, O. N. do S. E. *O sistema em números*. 2021. Available at: <<http://ons.org.br/paginas/sobre-o-sin/o-sistema-em-numeros>>.

PAUL, C. R. *Introduction to electromagnetic compatibility*. John Wiley & Sons, 2006. v. 184. Available at: <[https://books.google.com.br/books/about/Introduction\\_to\\_Electromagnetic\\_Compatib.html](https://books.google.com.br/books/about/Introduction_to_Electromagnetic_Compatib.html)>.

PETROU, M. M.; PETROU, C. *Image processing: the fundamentals*. John Wiley & Sons, 2010. Available at: <[https://books.google.com.br/books/about/Image\\_Processing.html](https://books.google.com.br/books/about/Image_Processing.html)>.

PINTO, L. G. M. et al. A ssd-ocr approach for real-time active car tracking on quadrotors. In: SPRINGER. *16th International Conference on Information Technology-New Generations (ITNG 2019)*. [N.a.], 2019. p. 471–476.

PIVETTA, C.; MANTOVANI, G.; ZOTTIS, F. Transformada de hough. *Universidade Estadual do Oeste do Paraná*, p. 102–103, 2017. Available at: <<https://www.inf.unioeste.br/~adair/PID/Notas/%20Aula/Transformada/%20de/%20Hough.pdf>>.

POTEYEVA, M. et al. Search and rescue activities in disasters. In: *Handbook of disaster research*. Springer, 2007. p. 200–216. Available at: <[https://link.springer.com/chapter/10.1007/978-0-387-32353-4\\_12](https://link.springer.com/chapter/10.1007/978-0-387-32353-4_12)>.

RAFFIN, A. et al. Stable baselines3. *GitHub repository*, GitHub, 2020. Available at: <<https://github.com/DLR-RM/rl-baselines3-zoo>>.

RANGEL, R. K.; KIENITZ, K. H.; BRANDÃO, M. P. Sistema de inspecao de linhas de transmissao de energia electrica utilizando veiculos aereos nao-tripulados. Sep, 2009. Available at: <<http://www.cta-dlr2009.ita.br/Proceedings/PDF/59018.pdf>>.

RESENDE, G. V. de. Segurança na manutenção de linhas de alta tensão. In: APTEL / PETROBRAS. *XVI Seminário Nacional de Telecomunicações – Rio de Janeiro -RJ*. 2017. Painel: Segurança, Meio Ambiente e Saúde. Available at: <<https://www.aptel.com.br/event/92/home-93>>.

RIBEIRO, L. et al. Visual servo control in quadrotor for power line tracking. 2018. Available at: <<https://www.semanticscholar.org/paper/Visual-Servo-Control-in-Quadrotor-for-Power-Line-Ribeiro-Duarte/7510efe77c76fbdf80a5c6752e4bbc77a19802fd>>.

RIEDMAIER, S. et al. Survey on scenario-based safety assessment of automated vehicles. *IEEE access*, IEEE, v. 8, p. 87456–87477, 2020. Available at: <<https://ieeexplore.ieee.org/document/9090897>>.

RUSSELL, S.; NORVIG, P. Artificial intelligence: a modern approach. 2002. Available at: <[https://books.google.com.br/books/about/Artificial\\_Intelligence.html](https://books.google.com.br/books/about/Artificial_Intelligence.html)>.

SHAH, S. et al. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In: SPRINGER. *Field and service robotics*. 2018. p. 621–635. Available at: <<https://arxiv.org/abs/1705.05065>>.

SHRIT, O. et al. A new approach to realize drone swarm using ad-hoc network. In: IEEE. *2017 16th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*. 2017. p. 1–5. Available at: <<https://hal.archives-ouvertes.fr/hal-01696735>>.

SILVA, G. C. da; MUNARO, M. Specifications of an adequate system for climbing poles. In: *Seminário Nacional de Distribuição de Energia Elétrica (SENDI) in Olinda-Pernambuco, Brazil*. [n.p.], 2008. Available at: <<https://www.cpfll.com.br/sites/xxisendi/sendisobre-o-sendi/Paginas/default.aspx>>.

SILVA, R. L. da. *Defeitos e falhas nas linhas de transmissão*. 2011. Available at: <<https://irp-cdn.multiscreensite.com/3e619e2f/pdf/MT-001.012.004-DEFEITOSEFALHASNASLINHASDETRNSM%E2%80%A6.pdf>>.

SMITH, T. *About ROS*. 2020. Available at: <<https://www.ros.org/about-ros/>>.

SRINIVASU, B.; RAO, P. S. Infrastructure development and economic growth: Prospects and perspective. *Journal of business management and Social sciences research*, v. 2, n. 1, p. 81–91, 2013. Available at: <<https://www.semanticscholar.org/paper/Infrastructure-Development-and-Economic-growth/%3A-and-Rao-Srinivasu/13528881527ae784b6e7c3c29489bfbba995d3b6>>.

STEVENS, E.; ANTIGA, L.; VIEHMANN, T. *Deep learning with PyTorch*. Manning Publications Company, 2020. Available at: <[https://books.google.com.br/books/about/Deep\\_Learning\\_with\\_PyTorch.html](https://books.google.com.br/books/about/Deep_Learning_with_PyTorch.html)>.

SUTTON, R. S.; BARTO, A. G. *Reinforcement learning: An introduction*. MIT press, 2018. Available at: <[https://books.google.com.br/books/about/Reinforcement\\_Learning.html](https://books.google.com.br/books/about/Reinforcement_Learning.html)>.

TEAM, A. D. *Ardupilot*. 2019. Available at: <<https://ardupilot.org/plane/index.html>>.

TEAM, M. *The Microsoft Cognitive Toolkit*. 2020. Available at: <<https://docs.microsoft.com/pt-br/cognitive-toolkit/>>.

- TEAM, P. *jMAVSim com SITL*. 2019. Available at: <<https://docs.px4.io/master/en/simulation/jmavsim.html>>.
- TRIDGELL, A. *MAVProxy*. 2019. Available at: <<https://github.com/ArduPilot/MAVProxy>>.
- TRIVEDI, N. B. et al. Geomagnetically induced currents–gic in electric power system at low latitudes in brazil: A case study. In: EUROPEAN ASSOCIATION OF GEOSCIENTISTS & ENGINEERS. *9th International Congress of the Brazilian Geophysical Society*. 2005. p. cp–160. Available at: <<https://www.earthdoc.org/content/papers/10.3997/2214-4609-pdb.160.SBGF402>>.
- TURING, A. M. Can a machine think. 1956. Available at: <[https://link.springer.com/chapter/10.1007/978-3-662-05642-4\\_12](https://link.springer.com/chapter/10.1007/978-3-662-05642-4_12)>.
- VARGAS, T. C. A. J. H. *SARPAS - Drone*. 2019. Available at: <<https://servicos.decea.mil.br/sarpas/>>.
- WANG, P.; OLIPHANT, T. *Your data science toolkit*. 2019. Available at: <<https://www.anaconda.com/products/individual>>.
- WERNER, T. P.; HYETT, P. J.; CHACON, C. W. S. *GitHub: Where the world builds software · GitHub*. 2019. Available at: <<https://github.com/>>.
- WILLEE, H. *Choosing a Ground Station*. 2019. Available at: <<https://ardupilot.org/copter/docs/common-choosing-a-ground-station.html>>.
- WU, Y. et al. Overhead transmission line parameter reconstruction for uav inspection based on tunneling magnetoresistive sensors and inverse models. *IEEE Transactions on Power Delivery*, IEEE, v. 34, n. 3, p. 819–827, 2019. Available at: <<https://ieeexplore.ieee.org/document/8603761>>.
- XIE, X. et al. A multiple sensors platform method for power line inspection based on a large unmanned helicopter. *Sensors, Multidisciplinary Digital Publishing Institute*, v. 17, n. 6, p. 1222, 2017. Available at: <<https://www.mdpi.com/1424-8220/17/6/1222>>.
- YANG, Z.; LIN, F.; CHEN, B. M. Survey of autopilot for multi-rotor unmanned aerial vehicles. In: IEEE. *IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society*. 2016. p. 6122–6127. Available at: <<https://ieeexplore.ieee.org/document/7793820>>.
- ZEILER, M. D.; FERGUS, R. Visualizing and understanding convolutional networks. In: SPRINGER. *European conference on computer vision*. 2014. p. 818–833. Available at: <<https://arxiv.org/abs/1311.2901>>.

ZHANG, X. et al. A binocular vision based auxiliary navigation system of the unmanned aerial vehicle for power line inspection. In: IEEE. *2016 3rd International Conference on Systems and Informatics (ICSAI)*. 2016. p. 291–296. Available at: <<https://ieeexplore.ieee.org/abstract/document/7810970?>>

ZHOU, G. et al. Robust real-time uav based power line detection and tracking. In: IEEE. *2016 IEEE International Conference on Image Processing (ICIP)*. 2016. p. 744–748. Available at: <<https://ieeexplore.ieee.org/document/7532456>>.

ZORMPAS, A. et al. Power transmission lines inspection using properly equipped unmanned aerial vehicle (uav). In: IEEE. *2018 IEEE International Conference on Imaging Systems and Techniques (IST)*. 2018. p. 1–5. Available at: <<https://ieeexplore.ieee.org/document/8577142>>.