



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ
CAMPUS TERESINA CENTRAL
BACHARELADO EM ENGENHARIA MECÂNICA

CAIO DE SOUSA FERREIRA VILANOVA

**APLICAÇÃO DE UMA AERONAVE REMOTAMENTE PILOTADA COMO
PLATAFORMA ROBÓTICA PARA PRÁTICAS EM LABORATÓRIO DA
GRADUAÇÃO EM ENGENHARIA MECÂNICA DO INSTITUTO FEDERAL DO PIAUÍ**

TERESINA

2022

CAIO DE SOUSA FERREIRA VILANOVA

APLICAÇÃO DE UMA AERONAVE REMOTAMENTE PILOTADA COMO PLATAFORMA
ROBÓTICA PARA PRÁTICAS EM LABORATÓRIO DA GRADUAÇÃO EM ENGENHARIA
MECÂNICA DO INSTITUTO FEDERAL DO PIAUÍ

Trabalho de Conclusão de Curso (monografia)
apresentado como exigência parcial para
obtenção do diploma do curso de Bacharelado
em Engenharia Mecânica do Instituto Federal
de Educação, Ciência e Tecnologia do Piauí,
Campus Teresina Central.

Orientador: Prof. Me. Francisco Marcelino
Almeida de Araújo.

TERESINA

2022

Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD

Vilanova, Caio de Sousa Ferreira

V696a Aplicação de uma aeronave remotamente pilotada como plataforma robótica para práticas em laboratório da graduação em Engenharia Mecânica do Instituto Federal do Piauí o / Caio de Sousa Ferreira Vilanova. - 2022.
42 f.: il. color.

Trabalho de Conclusão de Curso (Bacharelado em Engenharia Mecânica)
- Instituto Federal de Educação, Ciência e Tecnologia do Piauí, Campus Teresina Central, 2022.

Orientador : Prof Me. Francisco Marcelino Almeida de Araújo .

1. Drones. 2. Robótica. 3. ROS. 4. Laboratório. I.Título.

CDD - 620

Elaborado por Maria Rosismar Farias CRB 3/631

CAIO DE SOUSA FERREIRA VILANOVA

APLICAÇÃO DE UMA AERONAVE REMOTAMENTE PILOTADA COMO PLATAFORMA
ROBÓTICA PARA PRÁTICAS EM LABORATÓRIO DA GRADUAÇÃO EM ENGENHARIA
MECÂNICA DO INSTITUTO FEDERAL DO PIAUÍ

Trabalho de Conclusão de Curso (monografia)
apresentado como exigência parcial para
obtenção do diploma do curso de Bacharelado
em Engenharia Mecânica do Instituto Federal
de Educação, Ciência e Tecnologia do Piauí,
Campus Teresina Central.

Aprovado (a) em: ____/____/____.

BANCA EXAMINADORA

Prof. Me. Francisco Marcelino Almeida de Araújo (Orientador)
Instituto Federal de Educação, Ciência e Tecnologia do Piauí (IFPI)

Prof. Me. Janiel Fontineles Silva
Instituto Federal de Educação, Ciência e Tecnologia do Piauí (IFPI)

Prof. Ph.D. Nuno Miguel Fonseca Ferreira
Instituto Superior de Engenharia de Coimbra (ISEC)

Prof. Ph.D. Ranulfo Plutarco Bezerra Neto
Tohoku University

Eng. Me. Carlos Erlan Olival Lima
Toyohashi University of Technology (TUT)

AGRADECIMENTOS

A realização deste trabalho se deve, principalmente, pela orientação, dedicação e compreensão de várias pessoas. Agradeço a todos que, em algum momento, contribuíram para a concretização dessa etapa da minha vida. Assim, de uma maneira especial, agradeço:

- aos meus pais Claécio e Gilmara por sempre estarem ao meu lado, apoiando, aconselhando e incentivando os meus estudos;
- a minha irmã Karen, pelo companheirismo e amizade ao longo de todo o processo;
- aos meus amigos de infância Ezer, Mateus, Paulo, Rafaela, Vicente e Vitória, pelos momentos de apoio e descontração;
- a todos os amigos que tive ao longo da trajetória de estudos na graduação;
- aos companheiros do laboratório LABIRAS, Filipe, Igor, Jefferson, Kauê, Matheus e Paulo, que me acompanharam durante a pesquisa deste trabalho;
- ao meu orientador Francisco Marcelino, pela amizade e disponibilidade em compartilhar o conhecimento para que este trabalho fosse realizado, além de todo o suporte proporcionado;
- a professora Lívia Macedo, pela instrução e acompanhamento dessa dissertação;
- ao Instituto Federal de Educação, Ciência e Tecnologia do Piauí, por me permitir um espaço e recursos para desenvolvimento da pesquisa.

“Qualquer tecnologia suficientemente avançada
é indistinguível da magia.”

(Arthur C. Clarke, 1998, p. 36)

RESUMO

Embora originalmente construídos para fins militares, os drones avançaram rapidamente e se tornaram populares como eletrônicos de consumo, especialmente na forma de quadricópteros. Assim, o mercado demanda cada vez mais profissionais que desenvolvam projetos com base em tecnologias de robótica aérea. Diante desse cenário, o presente trabalho apresenta o desenvolvimento de um método de utilização de uma aeronave controlada pelo *Robot Operating System* (ROS) como plataforma para práticas educacionais de robótica móvel em laboratório do curso de Engenharia Mecânica, conforme previsão do Projeto Pedagógico do curso ofertado pelo Instituto Federal de Educação, Ciência e Tecnologia do Piauí (IFPI). Para isso, foi necessário descrever a cinemática envolvendo a utilização de um quadrotor e estruturar o ROS como ferramenta de desenvolvimento de algoritmos para a comunicação com o hardware do dispositivo. Com isso, apesar das restrições de projeto, foi possível concluir a viabilidade da utilização dessa plataforma como ferramenta para uma metodologia ativa para possibilitar a prática dos conhecimentos.

Palavras-chave: Drones. Robótica. ROS. Laboratório. Engenharia.

ABSTRACT

Although originally built for military purposes, drones quickly advanced and became popular as consumer electronics, especially in the form of quadcopters. Thus, the market increasingly demands professionals who develop projects based on aerial robotics technologies. Given this scenario, the present presented the development of a method of using an aircraft controlled by the *Robot Operating System* (ROS) as a platform for educational practices of mobile robotics in the laboratory of the Mechanical Engineering course, as predicted by the Pedagogical Project of the course offered by the Federal Institute of Education, Science and Technology of Piauí (IFPI). For this, it was necessary to describe the kinematics involving the use of a quadrotor and to structure ROS as a tool for developing algorithms for communication with the device's hardware. Thus, despite the design restrictions, it was possible to conclude the feasibility of using this platform as a tool for an active methodology to enable the practice of knowledge.

Keywords: Drones. Robotics. ROS. Laboratory. Engineering.

LISTA DE FIGURAS

Figura 1 – Aula quinzenal do Robótica para Engenharias	12
Figura 2 – Eixos de movimentação da aeronave	16
Figura 3 – Variações de modelos de quadrotores	17
Figura 4 – Diagrama de componentes da aeronave	18
Figura 5 – Sensor de posicionamento da aeronave	18
Figura 6 – Funcionamento dos nós na estrutura do ROS 2	20
Figura 7 – Funcionamento dos tópicos na estrutura do ROS 2	20
Figura 8 – NASA’s VIPER	21
Figura 9 – Quadrotor Vesper	21
Figura 10 – DJI ® Ryze Tello Combo Boost	23
Figura 11 – Diagrama de funcionamento do Sistema RPA	24
Figura 12 – Máquina de estado	25
Figura 13 – Pacote Tello Driver	27
Figura 14 – Localização do botão <i>Power</i> da aeronave	28
Figura 15 – Esperando conexão com o computador	28
Figura 16 – Conexão via Wi-Fi com o computador	29
Figura 17 – RPA conectado ao computador	30
Figura 18 – Aeronave capturando imagem e transmitindo pelo ROS	30
Figura 19 – Listagem de Nós de Tópicos em funcionamento	31
Figura 20 – Dados de voo fornecidos pela aeronave	32
Figura 21 – Joystick conectado ao quadrotor	32
Figura 22 – rqt mostrando o ecossistema em modo <i>joystick</i>	33
Figura 23 – Listagem de nós e tópicos do ecossistema com o modo joystick e a teleoperação	34
Figura 24 – Interface de utilização do <i>teleop_key</i>	34
Figura 25 – rqt mostrando o ecossistema com o modo joystick e a teleoperação	35
Figura 26 – Diagrama de utilização da plataforma em laboratório	35
Figura 27 – Teste de aplicação de rotina de decolagem e pouso no mesmo ponto	36
Figura 28 – Planejamento do teste de aplicação de rotina de decolagem, translação e pouso	37
Figura 29 – Execução do teste de aplicação de rotina de decolagem, translação e pouso	37
Figura 30 – Atuação do sensor infravermelho no sistema VPS	38
Figura 31 – Atuação aleatória do sensor infravermelho no sistema VPS	39

LISTA DE TABELAS

Tabela 1 – Especificações DJI Tello	19
---	----

LISTA DE ABREVIATURAS E SIGLAS

ANAC	Agência Nacional de Aviação Civil
ANATEL	Agência Nacional de Telecomunicações
API	<i>Application Programming Interface</i>
CREA-MG	Conselho Regional de Engenharia e Agronomia de Minas Gerais
DECEA	Departamento de Controle do Espaço Aéreo
FIEPR	Federação das Indústrias do Estado do Paraná
ICAO	<i>International Civil Aviation Organization</i>
IFPI	Instituto Federal de Educação, Ciência e Tecnologia do Piauí
IMU	<i>Inertial Measurement Unit</i>
LABIRAS	Laboratório de Inovação e Pesquisa Aplicada
PPC	Projeto Pedagógico do Curso
ROS	<i>Robot Operating System</i>
RPA	<i>Remotely Piloted Aircraft System</i>
SDK	<i>Software Development Kit</i>

LISTA DE SÍMBOLOS

® Marca registrada

% Porcentagem

SUMÁRIO

1	INTRODUÇÃO	12
2	OBJETIVOS	14
2.1	OBJETIVO GERAL	14
2.2	OBJETIVOS ESPECÍFICOS	14
3	REFERENCIAL TEÓRICO	15
3.1	ROBÓTICA MÓVEL AÉREA	15
3.2	QUADROTORES	16
3.3	DJI RYZE TELLO	17
3.4	<i>ROBOT OPERATING SYSTEM</i>	19
3.4.1	Nós	19
3.4.2	Tópicos	20
3.4.3	Aplicações de referência	21
4	METODOLOGIA	23
4.1	ABORDAGEM PRELIMINAR	23
4.2	RATIFICAÇÃO DE RECURSOS	23
4.3	COMUNICAÇÃO COM O HARDWARE	24
4.4	DESENVOLVIMENTO DO SOFTWARE	24
4.5	ELABORAÇÃO DA MÁQUINA DE ESTADO	25
4.6	APLICAÇÃO DA PLATAFORMA ROBÓTICA	25
5	RESULTADOS E DISCUSSÃO	27
5.1	PACOTE DESENVOLVIDO NO ROS 2	27
5.2	CONEXÃO ENTRE OS DISPOSITIVOS	27
5.3	UTILIZAÇÃO DA PLATAFORMA	29
5.3.1	Estrutura do sistema operacional robótico	31
5.3.2	Joystick	32
5.3.3	Teleoperação	33
5.3.4	<i>Scripts</i> de comandos	35
5.4	TESTES DE APLICAÇÃO	36
5.4.1	Decolagem e pouso	36
5.4.2	Translação	36
5.4.3	Estabilização	38

6	CONSIDERAÇÕES FINAIS	40
7	TRABALHOS FUTUROS	41
	REFERÊNCIAS	42

1 INTRODUÇÃO

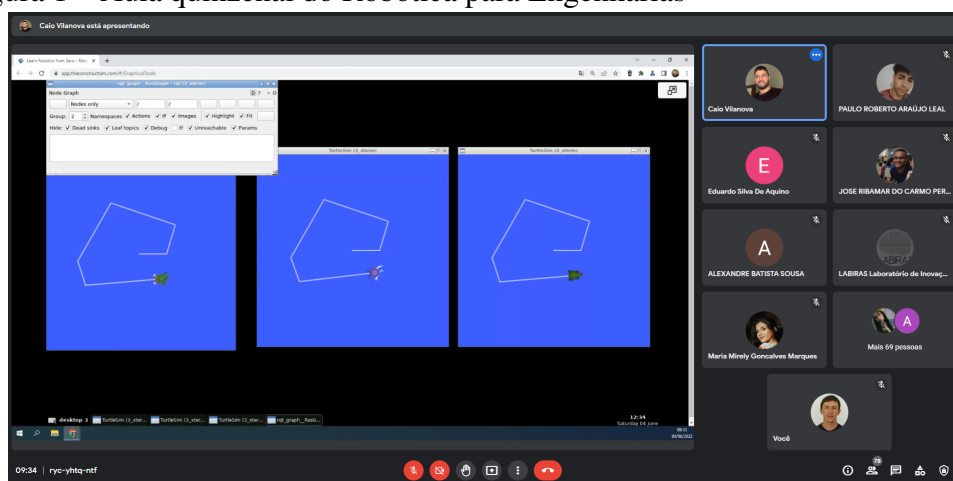
Embora originalmente construídos para fins militares, com implementos de armas e mísseis a serem controlados remotamente, os drones avançaram rapidamente e se tornaram populares como eletrônicos de consumo, especialmente na forma de quadricópteros, para serviços como entrega de mercadorias, auxílio em operações de busca e resgate, filmagem e fotografia, dentre outros (VYAS, 2020).

Conforme o Departamento de Controle do Espaço Aéreo (DECEA), o termo técnico *Remotely Piloted Aircraft System* (RPA), é padronizado internacionalmente pela *International Civil Aviation Organization* (ICAO) para se referir aos sistemas de aeronaves remotamente pilotadas utilizadas com propósitos não recreativos (DECEA, 2019).

Segundo a Federação das Indústrias do Estado do Paraná (FIEPR), com a utilização desse dispositivo robótico no ambiente industrial, se poderia alcançar uma economia de custos logísticos de até 20% e se observar um aumento na disponibilidade operacional de equipamentos e plataformas em até 30% (FIEPR, 2019).

Nesse cenário, entre os dias 28 e 30 setembro de 2021 foi realizado o 49º Congresso Brasileiro de Educação em Engenharia, com o tema central “A formação em Engenharia: Tecnologia, Inovação e Sustentabilidade”. No evento, foram discutidas as competências esperadas dos egressos na atualidade: “[...] o mercado demanda cada vez mais profissionais que desenvolvam as novas tecnologias digitais – como inteligência artificial, *big data*, *machine learning* e **robótica avançada**.” (CREA-MG, 2021, grifo do autor).

Figura 1 – Aula quinzenal do Robótica para Engenharias



Fonte: Elaborado pelo Autor.

Assim, com o objetivo de proporcionar aos alunos e professores dos cursos de Engenharias

uma base teórica de conhecimentos necessários para iniciar na robótica industrial, o projeto intitulado "Robótica Aplicada para Engenharia" foi realizado de forma online no Campus Teresina Central do Instituto Federal de Educação, Ciência e Tecnologia do Piauí (IFPI), por meio do Laboratório de Inovação e Pesquisa Aplicada (LABIRAS).

O curso, com duração de 4 meses, iniciado em 14 de maio de 2022, teve um total de 149 inscritos, sendo 44 desses colaboradores de empresas privadas que atuam diretamente em ambiente industrial. Em virtude disso, além do conteúdo programático do curso ministrado pela equipe de monitores, o projeto acrescentou uma capacitação suplementar em Python, com certificado da *Cisco Academy*®, por meio de parceria com o projeto Piauí Conectado (IFPI, 2022).

Entretanto, não foi possível a realização de atividades em laboratório, usando robôs físicos, pelos alunos do curso durante o período de execução do projeto. Diante desse contexto, esse trabalho propõe a utilização de uma metodologia ativa para possibilitar a prática dos conhecimentos adquiridos em sala de aula ao longo do curso, utilizando drones comerciais como recurso educacional em laboratório.

Essa proposta, após validação, permitirá a abertura de um segundo módulo do curso, de caráter presencial, ampliando a imersão dos discentes do Instituto Federal do Piauí ao campo de pesquisa e desenvolvimento da Robótica Industrial, aproximando-os de mais um ramo da engenharia disponível no mercado de trabalho.

2 OBJETIVOS

2.1 OBJETIVO GERAL

Desenvolver um método de utilização de uma aeronave controlada pelo *Robot Operating System* (ROS) como plataforma para práticas educacionais de robótica móvel em laboratório do curso de Bacharelado em Engenharia Mecânica do Instituto Federal do Piauí (IFPI).

2.2 OBJETIVOS ESPECÍFICOS

- Compreender a cinemática envolvendo a utilização de Aeronaves Remotamente Pilotadas;
- Estruturar o ROS como ferramenta de desenvolvimento de algoritmos para a comunicação com o hardware do dispositivo;
- Efetivar o recebimento de dados dos sensores embarcados no dispositivo;
- Realizar a operação do robô aéreo utilizando o ROS 2;
- Analisar a execução de rotinas na máquina de estado durante a utilização do sistema;
- Documentar os resultados da aplicação.

3 REFERENCIAL TEÓRICO

Tanto em aeronaves de grande porte quanto em sistemas de robótica móvel aérea, os subsistemas de comunicação informam o piloto e a estação terrestre sobre o estado atual do veículo. No entanto, os dispositivos de exibição de bordo podem ser totalmente substituídos por dispositivos de transmissão de rádio para as aeronaves remotamente pilotadas (KHAMVILAI *et al.*, 2021).

Para tanto, a utilização desses equipamentos deve se basear em plataformas confiáveis que não só atendam aos requisitos legais para sua operação mas também estejam embarcados sistemas de comunicação efetivos para evitar perda de controle da aeronave.

3.1 ROBÓTICA MÓVEL AÉREA

A Agência Nacional de Aviação Civil (ANAC) controla por meio do Regulamento Brasileiro de Aviação Civil Especial nº 94/2017 (RBAC-E nº 94/2017), as operações civis de aeronaves não tripuladas, complementando às normas de operação de drones estabelecidas pelo Departamento de Controle do Espaço Aéreo (DECEA) e as normas de homologação dos equipamentos segundo a Agência Nacional de Telecomunicações (ANATEL) (ANAC, 2022b).

Conforme ANAC (2022a), as aeronaves remotamente pilotadas (RPA) estão divididas em três classes, de acordo com o peso máximo de decolagem, no qual deve ser considerado os pesos da bateria ou combustível do equipamento e de carga eventualmente transportada: Classe 1 (Peso máximo de decolagem maior que 150 kg), Classe 2 (Peso máximo de decolagem maior que 25 kg e até 150 kg) e Classe 3 (Peso máximo de decolagem de até 25 kg).

Contudo, as aeronaves remotamente pilotadas de até 250g (pertencentes à Classe 3) estão dispensados de vários requisitos. Para esses equipamentos, as operações são permitidas pelo órgão regulamentador sem a necessidade de registro ou solicitação de voo, sendo as condições deste de total responsabilidade do seu piloto (ANAC, 2022c).

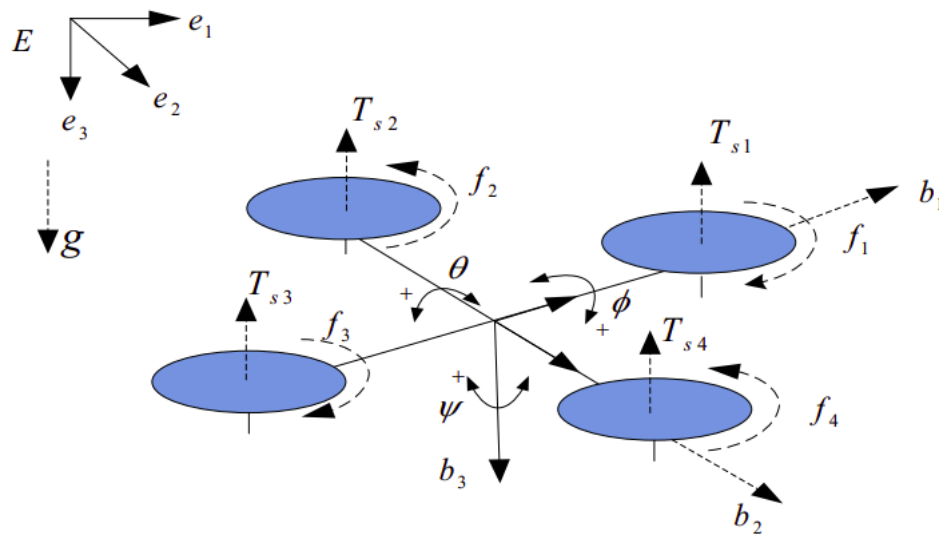
Dessa forma, esse tipo de aeronave se enquadra perfeitamente como meio a ser utilizado para a introdução ao desenvolvimento e utilização desse tipo de dispositivo robótico na prática educacional em ambiente controlado, como é o caso de laboratórios nas universidades.

3.2 QUADROTORES

Os quadrotors têm sido cada vez mais usados como plataforma preferida de veículo aéreo não tripulado (UAV) para aplicações internas. Algumas das características importantes dos quadricópteros em comparação com uma ampla seleção de outros veículos aéreos não tripulados incluem sua manobrabilidade, capacidade de decolagem/aterissagem vertical (VTOL), baixo custo e baixa manutenção (EMRAN; NAJJARAN, 2018).

Um quadrotor tem seis graus de liberdade e apenas quatro atuadores. Seu mecanismo é modelado em uma configuração cruzada com braços simétricos. A simetria permite a centralização da carga útil e do sistema de controle. Os eixos de rotação da hélice são fixos e paralelos entre si e todas as hélices têm pás de passo fixo, sendo que seu fluxo de ar é para baixo para criar um empuxo de elevação (EMRAN; NAJJARAN, 2018).

Figura 2 – Eixos de movimentação da aeronave



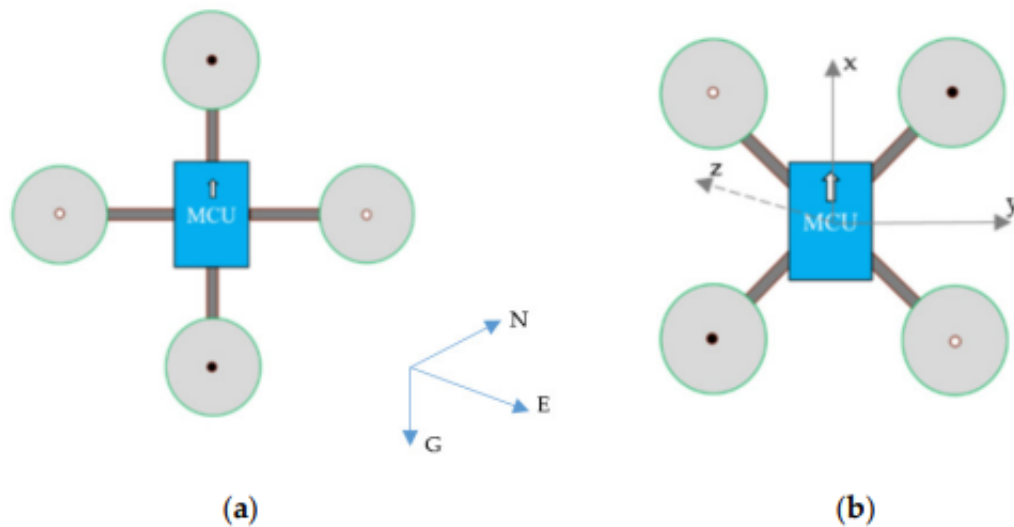
Fonte: Wang e Liu (2018)

No entanto, para equilibrar o quadrotor e eliminar a necessidade de um rotor de cauda, as hélices adjacentes devem girar em sentido contrário. As variáveis de estado do sistema são controladas usando acelerador (empuxo), roll, pitch e yaw que estão diretamente relacionados às velocidades das hélices. Subsequentemente, o quadricóptero pode atingir uma altitude e atitude desejadas (EMRAN; NAJJARAN, 2018).

Para isso, possuem dois tipos comerciais de configuração, conforme mostrado na Figura 3: em formato de cruz ou de xis. Os dois tipos mencionados têm diferentes distribuições de torque (FERDAUS *et al.*, 2020). Além disso, a configuração em xis permite uma melhor acomodação de dispositivos como câmera, Lidar e outros sensores, por evitar que sua área frontal seja bloqueada

pelos rotores (RAO *et al.*, 2022).

Figura 3 – Variações de modelos de quadrotores



Fonte: Rao *et al.* (2022)

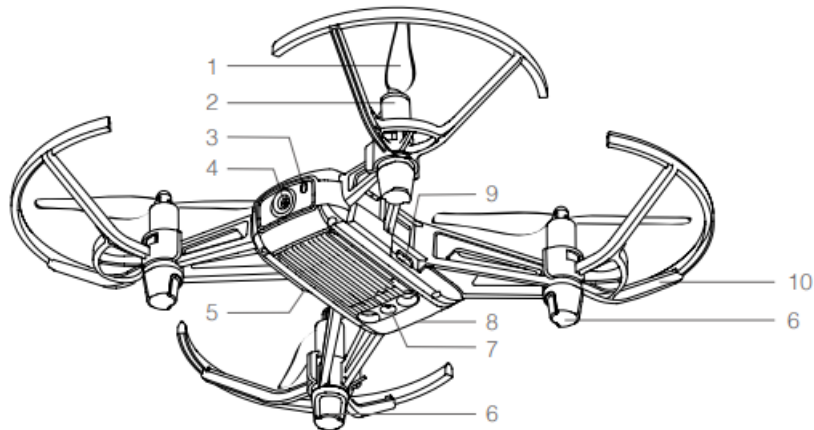
3.3 DJI RYZE TELLO

O Tello se destaca como sendo o primeiro veículo aéreo comercialmente disponível como uma plataforma dedicada ao aprendizado de programação nas linguagens Scratch, Python e Swift, anunciado para esta finalidade na sua versão nomeada Tello EDU. Ainda, apesar desta versão não ser distribuída em território brasileiro, a versão convencional pode ser utilizada com o mesmo propósito, uma vez que a programação de códigos para controle da aeronave pode ser desenvolvida utilizando o *Software Development Kit* (SDK) disponibilizado pelo fabricante: fator que contribui para ampliar seu potencial de utilização na pesquisa e educação (GIERNACKI *et al.*, 2022).

A estrutura geral de acesso ao *hardware* do equipamento é dividida em duas partes. O sistema de controle de voo é comercial e com acesso fechado, capaz de interpretar os dados de sensores e responder utilizando correções pré estabelecidas de forma a proporcionar a operação de maneira segura e constante (KHAMVILAI *et al.*, 2021).

Conforme ilustrado na Figura 4, esse dispositivo é equipado com conjunto motopropulsor formado por hélice-motor (itens 1 e 2 no diagrama, respectivamente) e esse controlador é responsável pelo transporte dos comandos de movimentação a estes componentes, baseado nos valores de entrada recebidos via conexão de rede local sem fio com algum dispositivo executando uma *Application Programming Interface* (API) compatível com o software do fabricante.

Figura 4 – Diagrama de componentes da aeronave

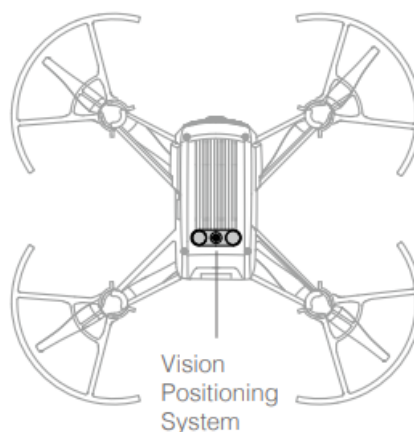


Fonte: Ryze Robotics (2018, adaptado)

Em segundo nível de acesso estão os sensores complementares, capazes de fornecer dados ao usuário informações que auxiliam na operação do aeronave, como porcentagem de bateria, imagem da câmera, informações de *Inertial Measurement Unit* (IMU) para os três eixos de referência, dentre outros.

Além disso, o dispositivo conta com a tecnologia de *Vision Positioning System* (VPS) que ajuda a aeronave a manter sua posição. Assim, o Tello pode pairar no lugar com mais precisão e voar em ambientes internos ou externos em condições sem vento. Os principais componentes do VPS são uma câmera e um módulo infravermelho 3D localizado na parte inferior da aeronave (RYZE ROBOTICS, 2018).

Figura 5 – Sensor de posicionamento da aeronave



Fonte: Ryze Robotics (2018, adaptado)

Na Tabela 1 encontra-se disposta a descrição de outras especificações importantes sobre a aeronave.

Tabela 1 – Especificações DJI Tello

Aeronave (Modelo: TLW004)	
Peso (incluindo protetores de hélice)	87 g
Velocidade máxima	17.8 mph (28.8 km/h)
Tempo Máximo de Voo	13 minutos (0 vento, a constantes 9 mph (15 km/h))
Faixa de temperatura operacional	32 a 104 °F (0 a 40 °C)
Faixa de Frequência Operacional	2,4 a 2,4835 GHz
Transmissor (EIRP)	20 dBm (FCC), 19 dBm (CE), 19 dBm (SRRC)
Câmera	
Tamanho máximo da imagem	2592×1936
Modos de gravação de vídeo	HD: 1280×720 30p
Formato de vídeo	MP4
Bateria de voo	
Capacidade	1100 mAh
Tensão	3.8 V
Tipo de Bateria	LiPo
Energia	4.18 Wh
Peso líquido	25±2 g
Faixa de temperatura de carregamento	41 a 113 °F (5 a 45 °C)
Potência máxima de carregamento	10 W

Fonte: Ryze Robotics (2018, adaptado)

3.4 ROBOT OPERATING SYSTEM

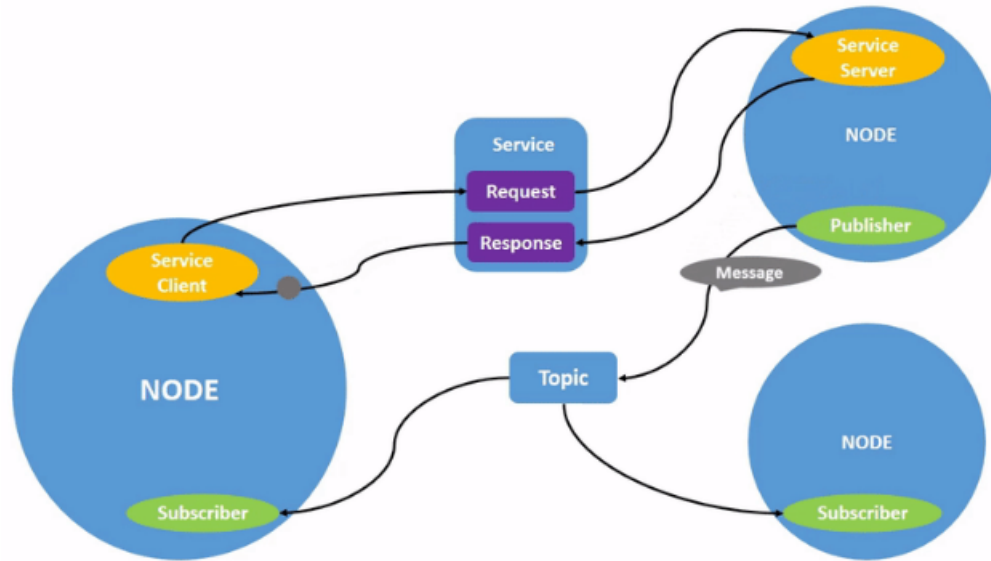
O *Robot Operating System* (ROS) é um SDK de código aberto que fornece bibliotecas e ferramentas que permitem a abstração de *hardware*, acesso a drivers de dispositivo, visualizadores e simuladores, transmissão de mensagens, gerenciamento de pacotes e muito mais (OPEN ROBOTICS, 2022).

E com isso, suportando uma ampla gama de aplicações, desde educação e pesquisa até desenvolvimento e implantação de produtos, o ROS compreende um grande conjunto de componentes de *software* que são comumente usados para desenvolvimento na robótica no contexto da indústria mundial (MACENSKI *et al.*, 2022).

3.4.1 Nós

Cada nó no ROS deve ser responsável por um único propósito (por exemplo, um nó para controlar os motores das rodas, um nó para controlar um telêmetro a laser, etc). Cada nó pode enviar e receber dados para outros nós por meio de tópicos, serviços, ações ou parâmetros (Figura 6). Um sistema robótico completo é composto por muitos nós trabalhando em conjunto. No ROS 2, um único executável (programa C++, programa Python, etc.) pode conter um ou mais nós (ROS, 2022a).

Figura 6 – Funcionamento dos nós na estrutura do ROS 2

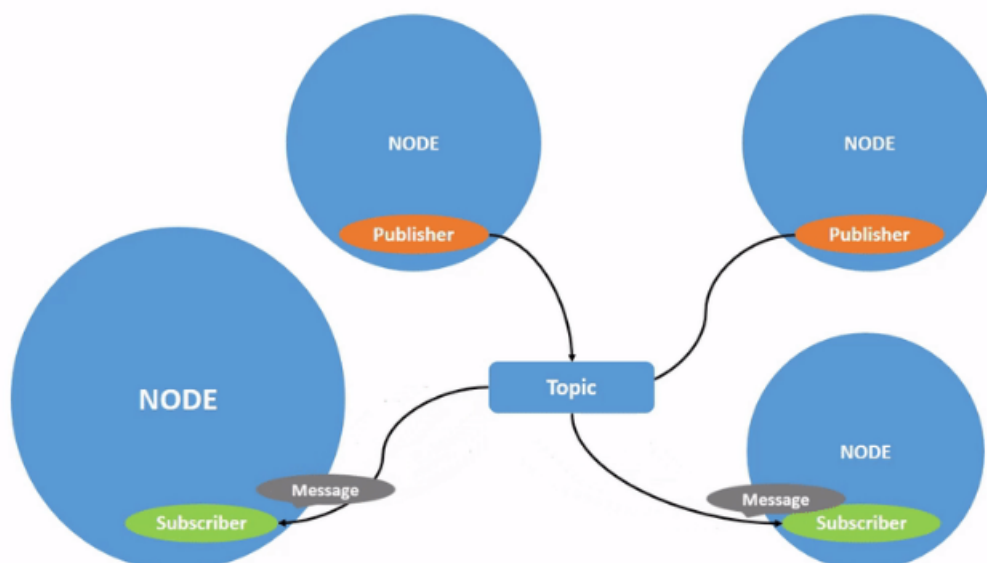


Fonte: ROS (2022a, adaptado)

3.4.2 Tópicos

O ROS 2 divide sistemas complexos em muitos nós modulares (Figura 7). Os tópicos são um elemento essenciais que atuam como um barramento para os nós trocarem mensagens. Um nó pode publicar dados para qualquer número de tópicos e simultaneamente ter assinaturas para qualquer número de tópicos (ROS, 2022b).

Figura 7 – Funcionamento dos tópicos na estrutura do ROS 2

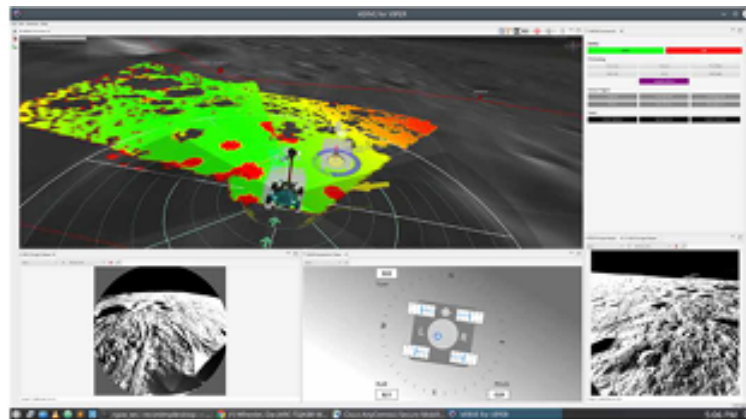


Fonte: ROS (2022b, adaptado)

3.4.3 Aplicações de referência

Como casos de sucesso, essa ferramenta de desenvolvimento é aplicada no gerenciamento de missão de voo em ambiente lunar (Figura 8) no projeto VIPER da *National Aeronautics and Space Administration* (NASA) e também nos produtos da Auterion®, referência em sistemas de veículos aéreos autônomos (MACENSKI *et al.*, 2022).

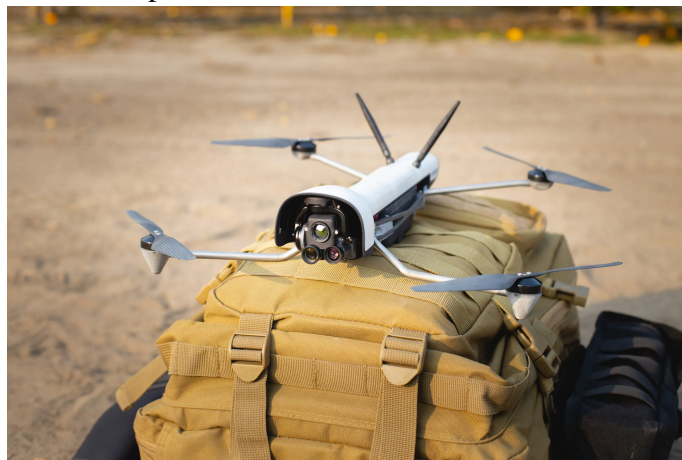
Figura 8 – NASA's VIPER



Fonte: Macenski *et al.* (2022, adaptado)

A Figura 9 mostra o Vesper®, produzido pela Vantage Robotics®, que utiliza como interface de controle o sistema de piloto automático de código aberto chamado Pixhawk 4®, o qual o gerenciamento é compatível com o ROS 2 por meio da utilização do micro-ROS, que providencia a utilização do sistema em microcontroladores.

Figura 9 – Quadrotor Vesper



Fonte: AUTERION (2020b)

A utilização do quadrotor, é resultado da parceria entre a empresa Auterion® e o governo dos Estados Unidos da América (EUA), motivada pela decisão do governo dos EUA de padronizar o PX4 para utilização em pequenos drones em missões de reconhecimento desarmado

(AUTERION, 2020b). A empresa, é responsável por uma plataforma de *software* integrada, baseada no ROS, completa para operações avançadas de voo, que permite a prevenção de colisão, controle de câmera e gimbal, gerenciamento de registros de voo e localização, entre outras implementações (AUTERION, 2020a).

4 METODOLOGIA

Segundo Leão (2016, p. 20) "Qualquer atividade por mais simples que seja necessita de um método para ser realizada. Método é um conjunto de processos para atingir determinados resultados. Emprega-se em qualquer domínio para se alcançar determinado fim ou fins". Assim, para a realização deste trabalho será seguida a sequência de ações listadas nas seções abaixo.

4.1 ABORDAGEM PRELIMINAR

Em primeiro momento, foi executado um estudo relacionado à aplicação da robótica móvel aérea utilizando o ROS. Nesse sentido, foram realizadas pesquisas em repositórios online acessados utilizando o buscador Google Acadêmico, procurando por livros, teses, dissertações e artigos científicos, nos idiomas português e inglês, limitados a publicações de até 5 anos. Além disso, sites técnicos de organizações regulamentadoras tiveram que ser consultados buscando orientações sobre a legislação vigente sobre o tema.

4.2 RATIFICAÇÃO DE RECURSOS

Para realização do trabalho, foram utilizados equipamentos de informática disponíveis no ambiente de laboratório do LABIRAS, acrescido de aeronaves DJI ® Ryze Tello (Figura 10), adquiridas para execução do projeto Robótica Aplicada para Engenharias, aprovado por meio do EDITAL 3/2021 - PROEX/REI/IFPI, de 29 de abril de 2021. Ademais, foi utilizado um controle em fio DualShock ® 4, fornecido pelo autor.

Figura 10 – DJI ® Ryze Tello Combo Boost



Fonte: Elaborado pelo autor.

4.3 COMUNICAÇÃO COM O HARDWARE

A aeronave utilizada, produzida pela Shenzhen Ryze Technology, é um quadrirotor do tipo em xis, que incorpora uma tecnologia de controlador de voo da DJI®, acompanhado de processadores Intel. O sistema RPA utilizando essa aeronave é descrito na Figura 11.

A conexão do entre esse dispositivo e o computador é feita por meio do acesso a uma rede WiFi do tipo IEEE 802.11n, operando na frequência de 2.4 GHz (RUBBESTAD; SÖDERQVIST, 2021), e é bilateral, ou seja, tanto o computador envia mensagens ao quadrotor, quanto este envia mensagens ao computador.

Ainda, o *joystick DualShock*® 4, fabricado pela Sony Group Corporation, se comunica com o sistema de forma unilateral por meio da conexão do tipo *Bluetooth* entre o dispositivo de entrada e o computador.

Figura 11 – Diagrama de funcionamento do Sistema RPA



Fonte: Elaborado pelo autor.

A efetivação da comunicação se deu por meio da identificação das mensagens que permitam o acionamento ou recebimento de dados dos componentes. Além disso, é necessária a observação do comportamento desses componentes durante a sua utilização a fim auxiliar na elaboração de rotinas de controle para a aeronave.

4.4 DESENVOLVIMENTO DO SOFTWARE

Utilizando o conteúdo de algoritmos e estruturas de dados das linguagens aprendidas na disciplina de Introdução à Programação, no Controle de sistemas dinâmicos apresentados na disciplina de Robótica, pertencente ao grupo de optativas de projetos, previstas dentro do Projeto Pedagógico do Curso (PPC) de Engenharia Mecânica no IFPI (LOPES *et al.*, 2018), foi estruturada uma interface de controle operacional para o robô utilizando um computador em rede local dentro do ambiente de laboratório.

Nesse projeto, o ROS é utilizado como meio para a criação de executáveis em pacotes e controle, baseados no SDK fornecido pelo fabricante, visando fornecer, a nível de usuário de engenharia, uma interface de controle para o sistema de RPA.

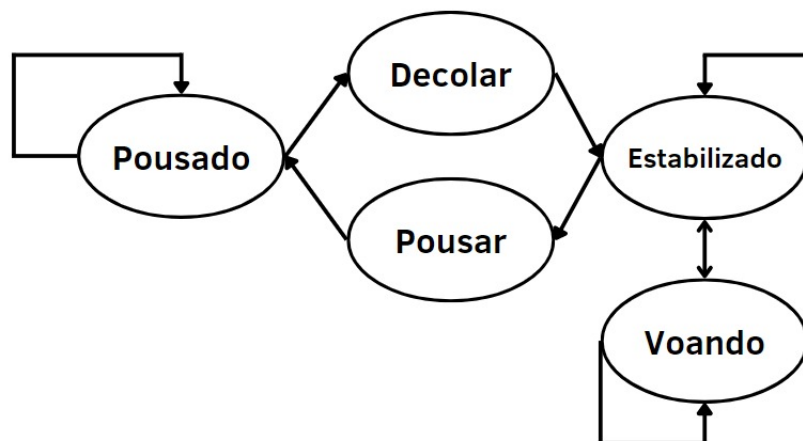
4.5 ELABORAÇÃO DA MÁQUINA DE ESTADO

O trabalho tem como foco o desenvolvimento de três modos de utilização, variando o meio para entrada de dados de operação, com a coleta de parâmetros a partir de uma máquina de estado externa, baseada no ROS, com o robô funcionando como nó individual ao computador:

- Utilizando um *joystick Bluetooth*;
- Utilizando a teleoperação com o teclado do computador;
- Execução no terminal de arquivos de *scripts* no formato *.sh*.

Assim, como ilustrado na Figura 12, a máquina de estado será baseado em cinco estados primários: pousado em espera, decolagem, estabilização, voo, pouso. Dessa forma, a alteração entre esses estados, instáveis e estáveis, permitirá a elaboração de rotinas de movimentação para a aeronave durante a execução de comandos.

Figura 12 – Máquina de estado



Fonte: Elaborado pelo autor.

4.6 APLICAÇÃO DA PLATAFORMA ROBÓTICA

O sistema de aeronave remotamente pilotada foi testado utilizando o ambiente físico do laboratório do LABIRAS, localizado no 5º andar do prédio B, no campus Teresina Central, a fim

de minimizar os riscos e controlar as variáveis do experimento. A prática se deu na execução de rotinas de movimentação padronizadas, para observação e avaliação do comportamento do robô em relação ao controle:

- Decolagem, estabilização e pouso;
- Decolagem, translação e pouso;
- Estabilização com o auxílio do sistema de VPS;

5 RESULTADOS E DISCUSSÃO

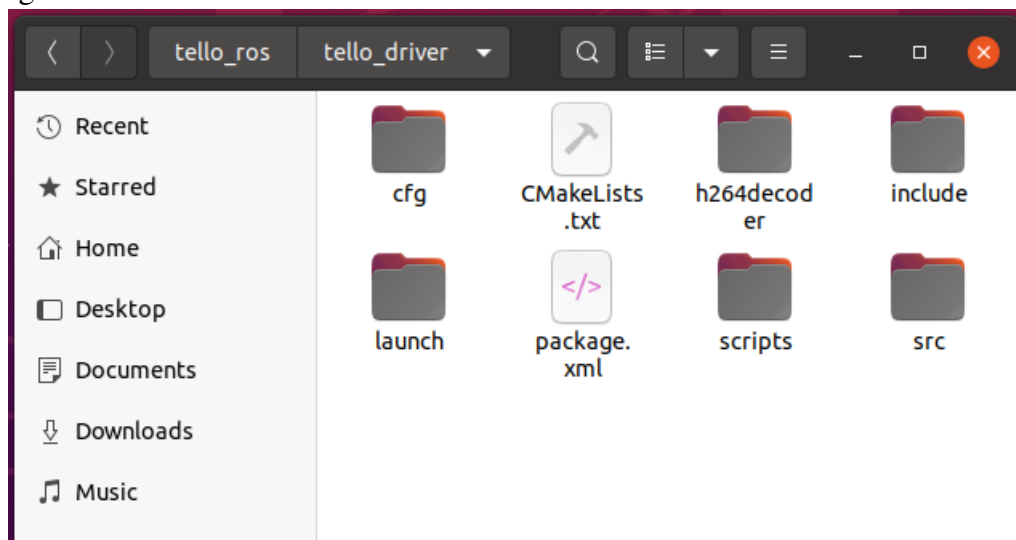
5.1 PACOTE DESENVOLVIDO NO ROS 2

O pacote desenvolvido para controle do veículo aéreo, chamado de Tello Driver, foi resultado não só da utilização de arquivos padrões de um pacote ROS 2, mas também de arquivos de configuração e decodificação, os quais foram inseridos de modo a permitir a comunicação com o hardware do dispositivo.

Assim, a Figura 13 apresenta o seu conteúdo. Na pasta *cfg*, encontra-se o arquivo *camera_info.yaml*, utilizado para equalizar as distorções nas imagens obtidas pela câmera embarcada no Tello, transformando os pixels da imagem capturada nativamente em 960x720 em dados dispostos em matrizes e aplicando neles coeficientes de correção.

Além disso, no diretório *h264decoder*, foi utilizado um decodificador de imagem de vídeo no formato de compressão h264 para que se pudesse utilizar a imagem processada na execução de nós do ROS 2.

Figura 13 – Pacote Tello Driver



Fonte: Elaborado pelo autor.

5.2 CONEXÃO ENTRE OS DISPOSITIVOS

Durante o processo de desenvolvimento, verificou-se a necessidade do *firmware* da aeronave estar atualizado para que seja possível a utilização da conexão utilizando pacotes baseados no SDK disponibilizado pela fabricante. Esse procedimento foi realizado com o drone conectado pelo aplicativo *TELLO*, disponível gratuitamente nas lojas de aplicativos Google Apps

e App Store.

Em seguida, o nome do ponto de acesso inicializado pelo Tello foi alterado, neste caso utilizou-se o *Service Set Identifier* (SSID) "*TELLO-CAIO*". Além disso, no mesmo aplicativo, uma senha pode ser adicionada ao ponto de acesso para proteger a conexão do dispositivo por usuários não autorizados.

Figura 14 – Localização do botão *Power* da aeronave



Fonte: Elaborado pelo autor.

A partir da inicialização do dispositivo, realizada ao pressionar o botão *Power* localizado na sua lateral direita (Figura 14), o led indicador de estado irá permutar entre as cores e estabilizar na cor amarela (Figura 15), indicando que o mesmo encontrar-se-á em modo de espera. Nesse modo, o dispositivo aguarda, sem movimentação, a conexão com outro equipamento de controle.

Figura 15 – Esperando conexão com o computador

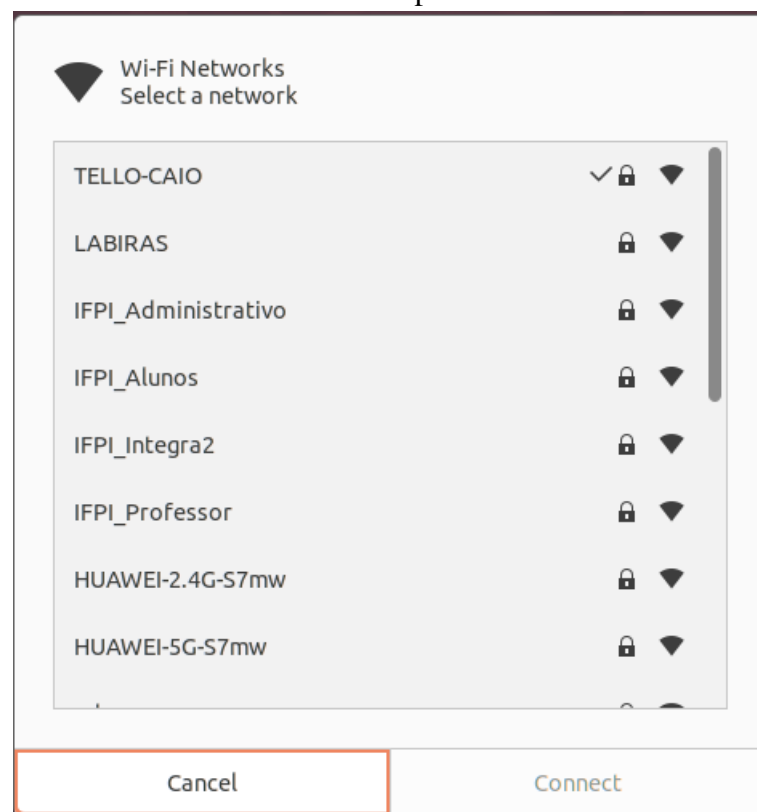


Fonte: Elaborado pelo autor.

Ainda, é importante destacar que verificou-se que não se deve permanecer nesse modo por muito tempo, uma vez que sem movimentação dos rotores que providenciam o empuxo derivado da circulação de uma massa de ar, ocorre o superaquecimento dos componentes internos, o que pode resultar em seu não funcionamento.

A conexão via rede sem fio por meio de Wi-Fi pode ser acessada utilizando o menu superior direito dentro da área de trabalho do sistema operacional linux, na guia de configuração e em seguida na aba de seleção de redes, como mostra a Figura 16. O usuário deve identificar o SSID configurado anteriormente pelo aplicativo na lista de redes disponíveis e clicar nela para que, ao informar a senha, o computador conecte ao ponto de acesso do drone.

Figura 16 – Conexão via Wi-Fi com o computador



Fonte: Elaborado pelo autor.

5.3 UTILIZAÇÃO DA PLATAFORMA

Para que o pacote de controle desenvolvido com base no ROS 2 possa ser utilizado, o arquivo de configuração no formato *.bash* deve ser carregado através da linha de comando *source install/setup.bash*.

Com isso, o arquivo *launch* pode ser executado através de *ros2 launch tello_driver teleop_launch.py*. A partir do momento em que o arquivo de *launch* está em execução, o led

na cor verde (Figura 17) indica a mudança no modo de utilização, no qual o veículo pode ser operado pelo usuário.

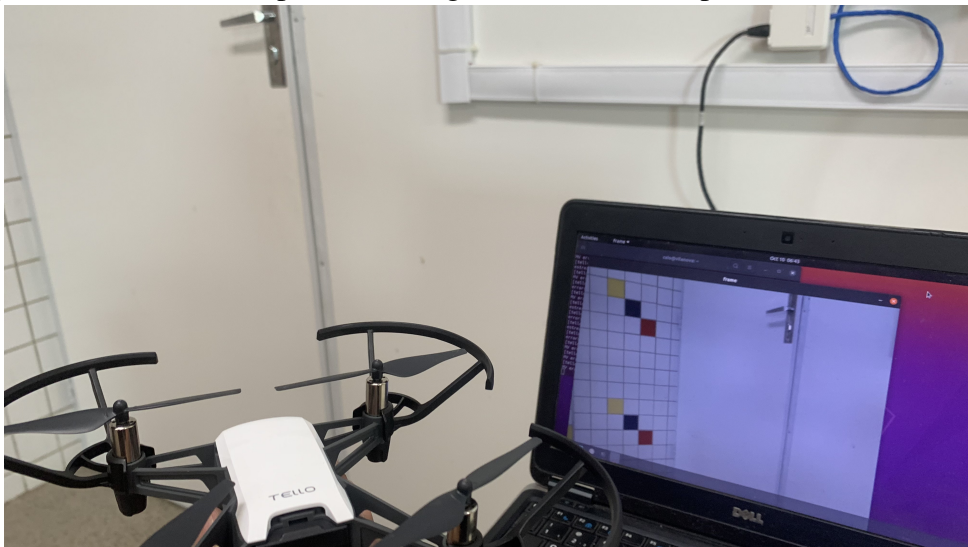
Figura 17 – RPA conectado ao computador



Fonte: Elaborado pelo autor.

Ainda, ao entrar em modo de utilização a interface de visualização da imagem da câmera é inicializada a fim de auxiliar o usuário na operação de controle da aeronave a qual o sistema está conectado (Figura 18).

Figura 18 – Aeronave capturando imagem e transmitindo pelo ROS

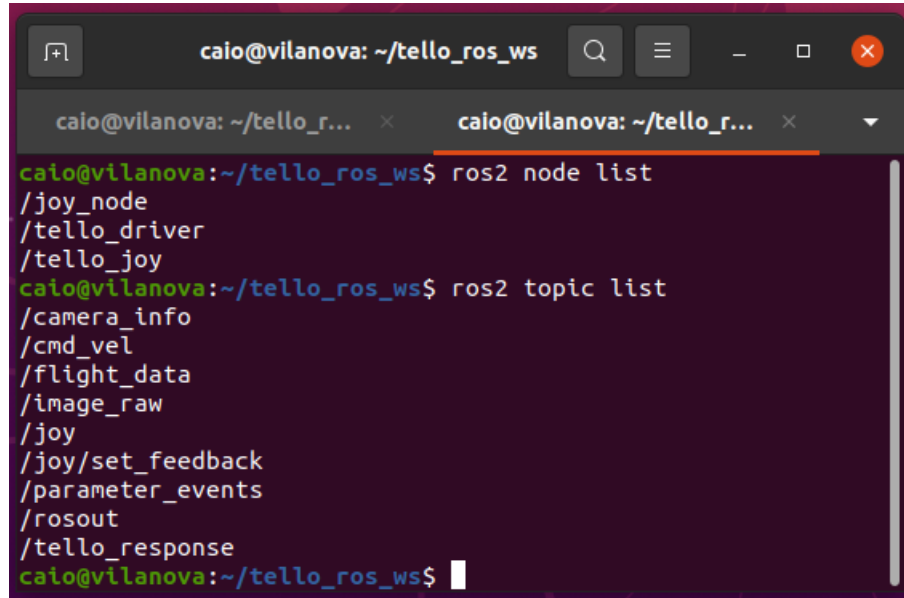


Fonte: Elaborado pelo autor.

5.3.1 Estrutura do sistema operacional robótico

Em uma guia do terminal *bash*, o usuário pode verificar os nós e os tópicos que podem ser acessados no sistema operacional robótico, por meio da utilização de comandos padrões do ROS 2: *ros2 node list* e *ros2 topic list* (Figura 19).

Figura 19 – Listagem de Nós de Tópicos em funcionamento

A screenshot of a terminal window titled 'caio@vilanova: ~/tello_ros_ws'. The terminal shows two commands and their outputs. The first command is 'ros2 node list', which outputs three lines: '/joy_node', '/tello_driver', and '/tello_joy'. The second command is 'ros2 topic list', which outputs ten lines: '/camera_info', '/cmd_vel', '/flight_data', '/image_raw', '/joy', '/joy/set_feedback', '/parameter_events', '/rosout', '/tello_response', and a blank line. The prompt 'caio@vilanova:~/tello_ros_ws\$' is visible at the end of the output lines.

```
caio@vilanova:~/tello_ros_ws$ ros2 node list
/joy_node
/tello_driver
/tello_joy
caio@vilanova:~/tello_ros_ws$ ros2 topic list
/camera_info
/cmd_vel
/flight_data
/image_raw
/joy
/joy/set_feedback
/parameter_events
/rosout
/tello_response
caio@vilanova:~/tello_ros_ws$
```

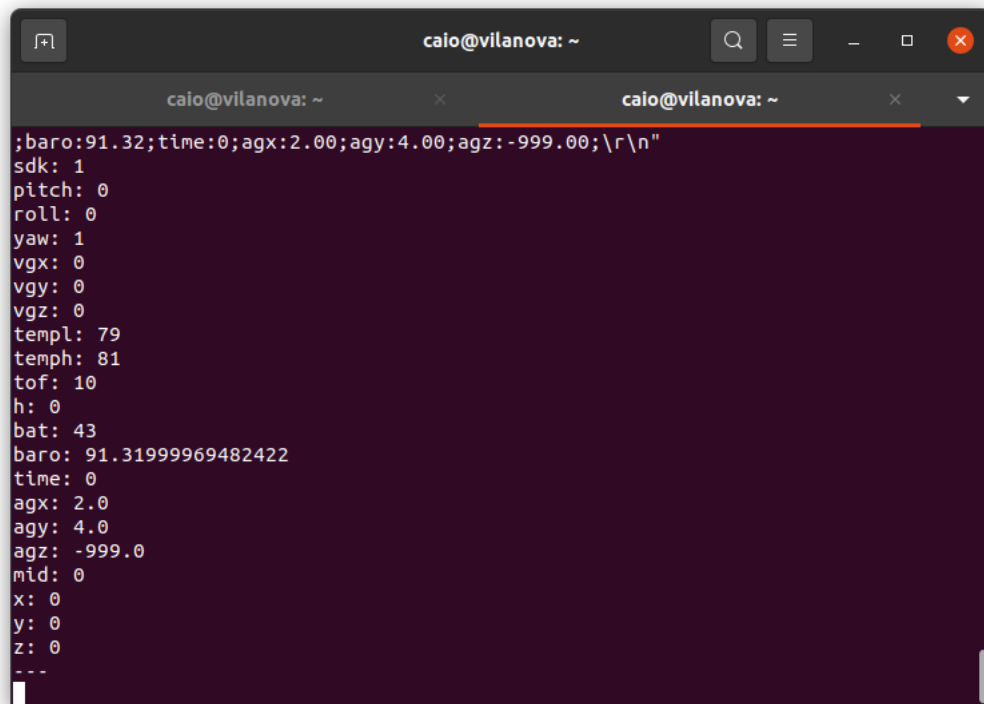
Fonte: Elaborado pelo autor.

Tem-se por padrão como nós disponíveis no ecossistema do ROS 2: *joy_node*, *tello_driver* e *tello_joy*. O *joy_node* torna um *joystick* bluetooth conectado ao computador um dispositivo capaz de controlar robôs. O *tello_driver* é o nó responsável por providenciar a conexão do ROS com o robô. O *tello_joy* traduz comandos do *joystick* para ações do Tello e mensagens *cmd_vel*.

O tópico *camera_info* é responsável pelo transporte dos metadados relacionados à própria câmera, como o tamanho da imagem e os parâmetros de calibração. O *cmd_vel* é essencial para transporte das mensagens de velocidade.

Já o *flight_data* inclui informações de voo como dados de barômetro e acelerômetro, porcentagem e temperatura de bateria (Figura 20). O *image_raw* é a própria imagem capturada pela câmera, como o nome sugere, e cada número exibido corresponde à intensidade do pixel. O *joy* contém o estado atual de cada um dos botões e eixos do joystick. Já o *parameter_events*, *rosout* e o *tello_response* são padrões de execução do ROS 2.

Figura 20 – Dados de voo fornecidos pela aeronave



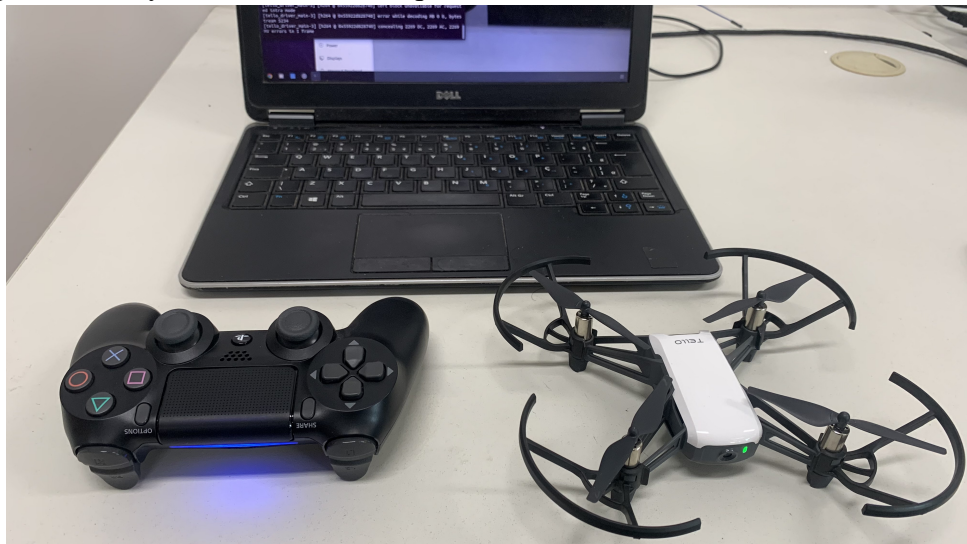
```
caio@vilanova: ~  
;baro:91.32;time:0;agx:2.00;agy:4.00;agz:-999.00;\r\n"  
sdk: 1  
pitch: 0  
roll: 0  
yaw: 1  
vgx: 0  
vgy: 0  
vgz: 0  
templ: 79  
temph: 81  
tof: 10  
h: 0  
bat: 43  
baro: 91.31999969482422  
time: 0  
agx: 2.0  
agy: 4.0  
agz: -999.0  
mid: 0  
x: 0  
y: 0  
z: 0  
--
```

Fonte: Elaborado pelo autor.

5.3.2 Joystick

A utilização da plataforma com o joystick permitiu uma gameificação do processo de operação, ao introduzir uma interface familiar ao público alvo como meio de inserção na prática de aula em laboratório. Na Figura 21, encontra-se com o led azul ligado, indicando a conexão estabelecida com o sistema, juntamente com o led verde na aeronave.

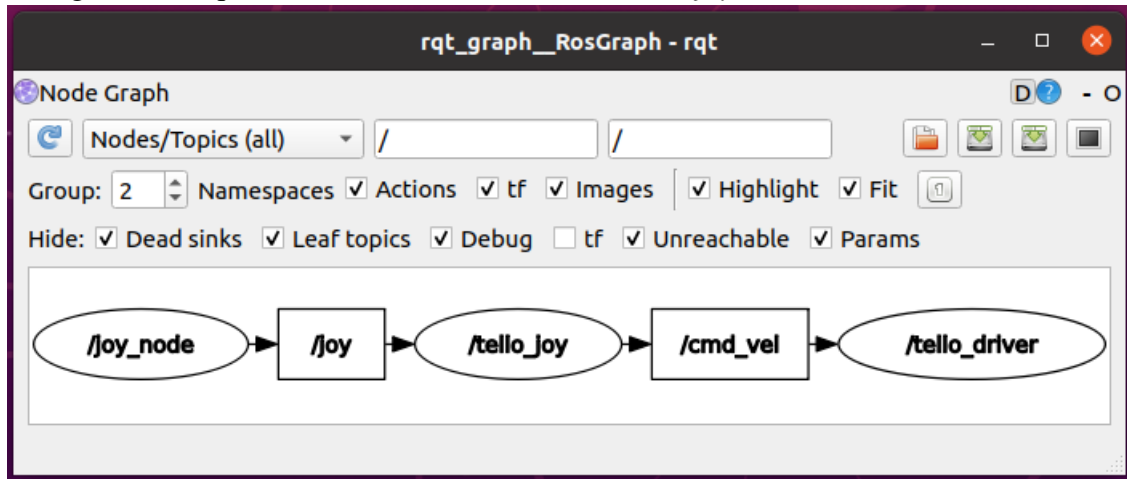
Figura 21 – Joystick conectado ao quadrotor



Fonte: Elaborado pelo autor.

A Figura 22, proporcionada por meio da utilização da ferramenta *rqt_graph*, e ilustra a sequência de comunicação dentro do ecossistema de nós e tópicos para o sistema da RPA com o computador, com o joystick sendo a interface de controle da aeronave disponível para o usuário.

Figura 22 – rqt mostrando o ecossistema em modo *joystick*



Fonte: Elaborado pelo autor.

5.3.3 Teleoperação

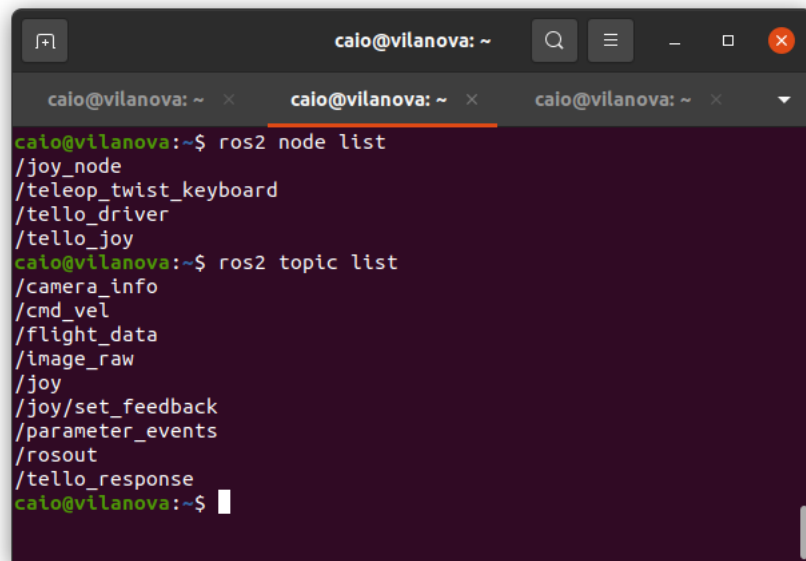
Uma segunda maneira de operação para o robô foi introduzida a partir da integração de um pacote padrão do ROS 2 chamado *teleop_twist_keyboard*. Este pacote inicia um nó que recebe comandos de entrada do teclado do computador os publica como mensagens Twist, por padrão já utilizando o caminho do tópico *cmd_vel*.

A inicialização do nó é feita utilizando o comando `ros2 run teleop_twist_keyboard teleop_twist_keyboard`, em uma nova janela de terminal, com o *launch* que faz a inicialização e a conexão em já em execução. A composição do sistema pode ser verificada Figura 23, com a listagem de nós e tópicos.

Como observado na Figura 24, o usuário é capaz de utilizar dois tipos de movimentação: em torno do veículo e de modo holonômico. A movimentação em torno do veículo permitiu uma movimentação mais agressiva, porém mais imprecisa. Nela, as teclas "i", "j", "l" e "k" correspondem respectivamente aos comandos de ir para frente, trás, esquerda, direita e estabilizar.

Além disso, foi possível a movimentação de modo holonômico, no qual a movimentação ocorre mantendo os eixos de referência no plano cartesiano, proporcionando uma pilotagem mais estável. Nesse modo, as teclas "I", "J", "L" e "K" correspondem respectivamente aos

Figura 23 – Listagem de nós e tópicos do ecossistema com o modo joystick e a teleoperação



```

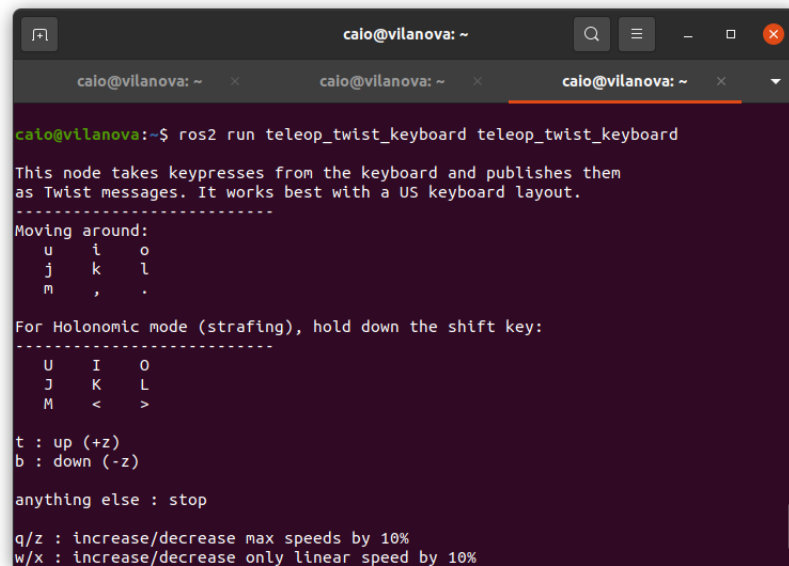
caio@vilanova: ~$ ros2 node list
/joy_node
/teleop_twist_keyboard
/tello_driver
/tello_joy
caio@vilanova:~$ ros2 topic list
/camera_info
/cmd_vel
/flight_data
/image_raw
/joy
/joy/set_feedback
/parameter_events
/rosout
/tello_response
caio@vilanova:~$

```

Fonte: Elaborado pelo autor.

comandos de ir para frente, trás, esquerda, direita e estabilizar.

Figura 24 – Interface de utilização do *teleop_key*



```

caio@vilanova:~$ ros2 run teleop_twist_keyboard teleop_twist_keyboard
This node takes keypresses from the keyboard and publishes them
as Twist messages. It works best with a US keyboard layout.
-----
Moving around:
  u   i   o
  j   k   l
  m   ,   .

For Holonomic mode (strafing), hold down the shift key:
-----
  U   I   O
  J   K   L
  M   <   >

t : up (+z)
b : down (-z)

anything else : stop

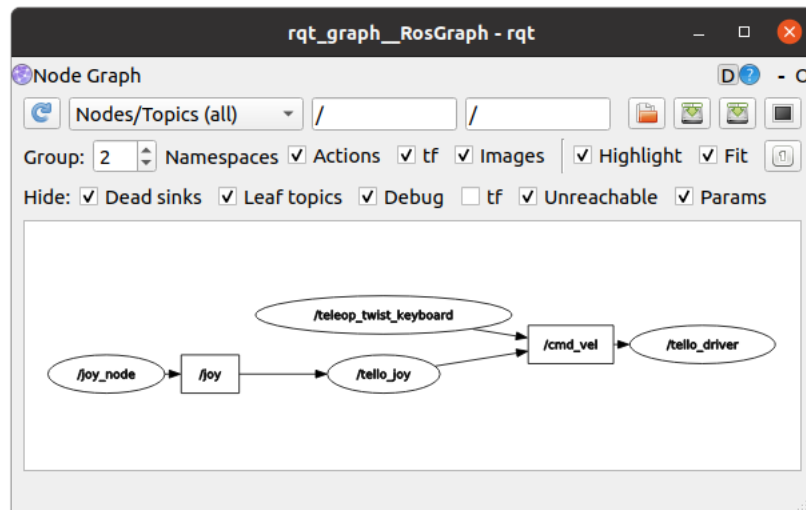
q/z : increase/decrease max speeds by 10%
w/x : increase/decrease only linear speed by 10%

```

Fonte: Elaborado pelo autor.

Usando novamente o *rqt_graph*, na Figura 25 podemos identificar a nova configuração do sistema de controle da RPA pelo ROS, com as entradas de comando disponíveis tanto para um joystick conectado ao computador, quanto a teleoperação utilizando o teclado do mesmo.

Figura 25 – rqt mostrando o ecossistema com o modo joystick e a teleoperação



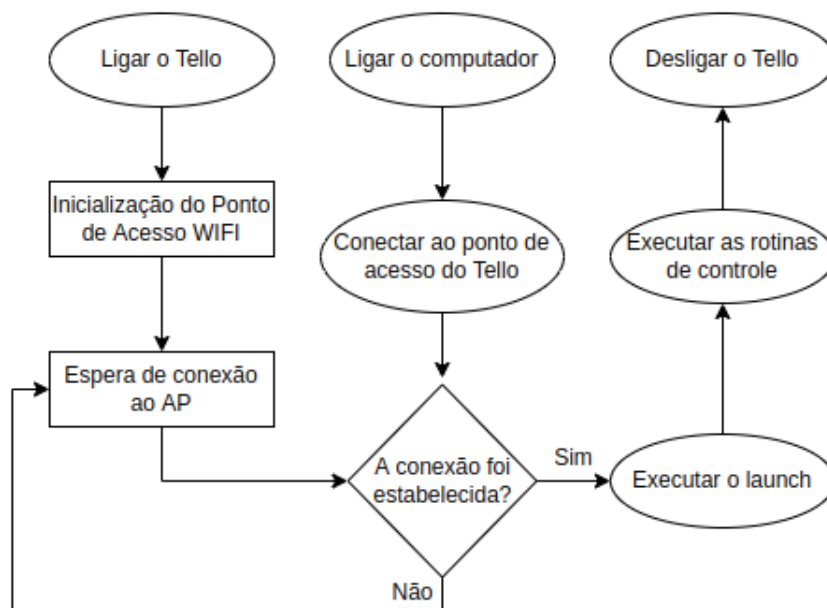
Fonte: Elaborado pelo autor.

5.3.4 Scripts de comandos

Com a validação dos pacotes desenvolvidos no ROS 2, bem como a conexão da aeronave com o sistema operacional robótico, foi possível padronizar o modo de utilização do conjunto como prática laboratorial.

Como ilustra a Figura 26, como ponto de partida o usuário deve ligar os dois dispositivos: a aeronave e o computador. Em seguida, o sistema passa pelos procedimentos que visam estabelecer a conexão, permitindo após isso a introdução do sistema de controle. Por fim, quando necessária a finalização da prática o usuário terá apenas que desligar o dispositivo.

Figura 26 – Diagrama de utilização da plataforma em laboratório



Fonte: Elaborado pelo autor.

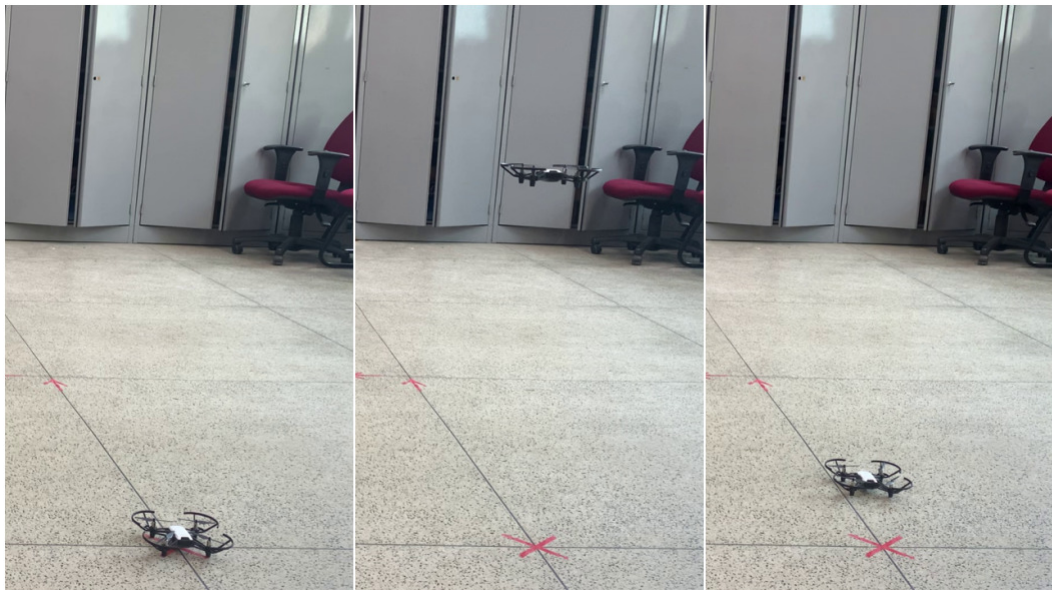
E com o intuito de simplificar a utilização do sistema, foram elaboradas, na forma de *scripts* executáveis, rotinas de utilização responsáveis por permitir ao usuário executar as ações recorrentes sem a necessidade de digitar várias linhas de código repetidamente. Isso foi realizado criando arquivos de dados no formato *.sh*, e introduzindo neles os comandos utilizados para controlar o sistema RPA. No entanto, foi verificada, ainda, a necessidade de se fornecer a eles permissões de execução dentro do sistema do linux.

5.4 TESTES DE APLICAÇÃO

5.4.1 Decolagem e pouso

A Figura 27 mostra a realização do teste no qual a aeronave decola e pouso. É possível verificar a interferência da massa de ar lateral que provoca uma ligeira translação do veículo no ar, fazendo com que ao pousar ela se encontre em posição diferente da qual foi lançada.

Figura 27 – Teste de aplicação de rotina de decolagem e pouso no mesmo ponto

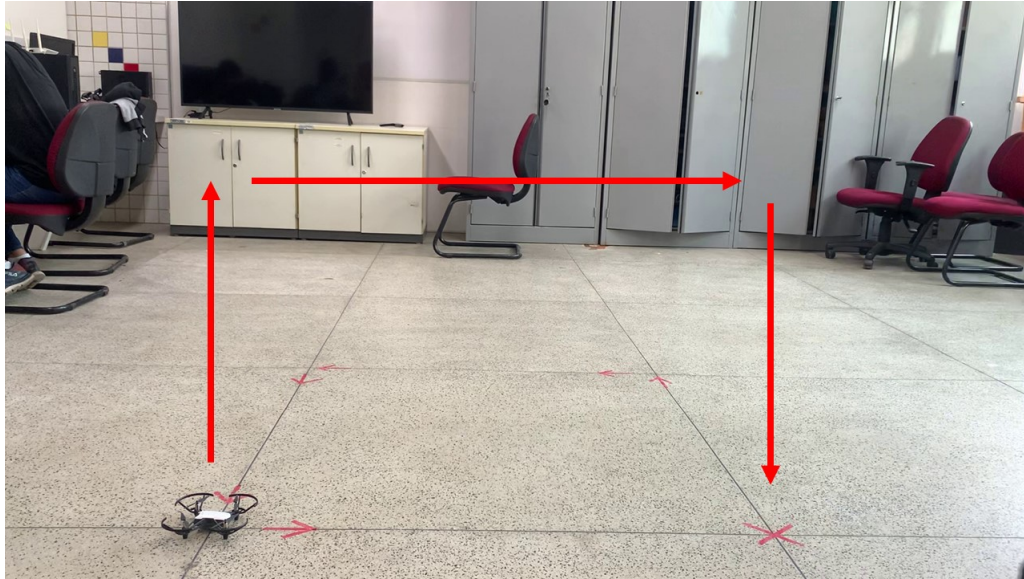


Fonte: Elaborado pelo autor.

5.4.2 Translação

A Figura 28 demonstra a realização do teste no qual a aeronave decola, translada em um eixo e pouso, dentro do ambiente do laboratório, mantendo-se estabilizada durante 10 segundos durante a realização de cada passo de movimentação requisitada.

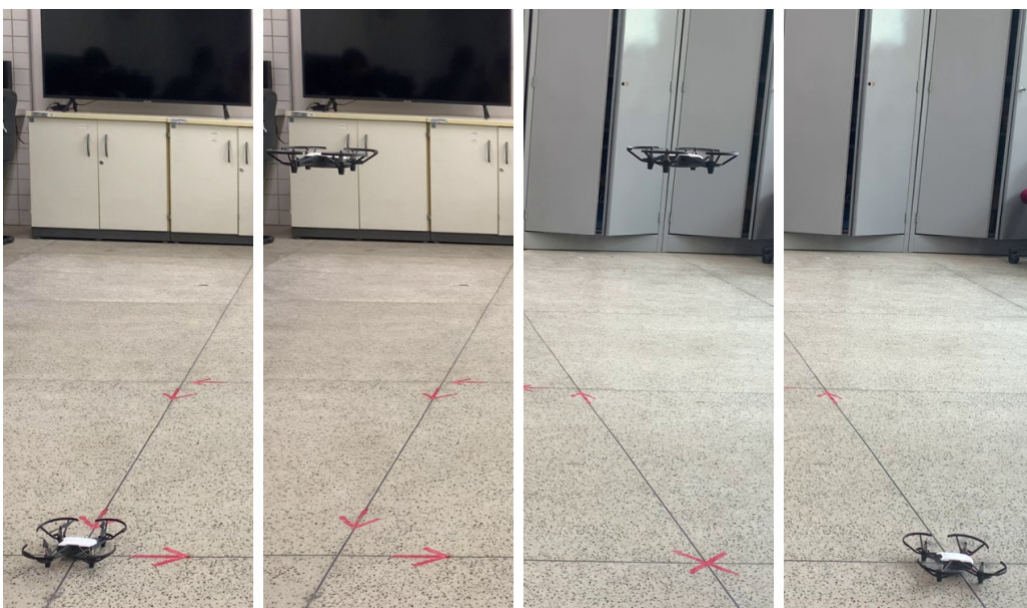
Figura 28 – Planejamento do teste de aplicação de rotina de decolagem, translação e pouso



Fonte: Elaborado pelo autor.

Durante a realização do teste, como mostra a Figura 29, o robô conseguiu realizar a movimentação planejada, sem perdas de informações e concluindo a rota planejada no *script* de comando. Além disso, verificou-se uma menor interferência da massa de ar lateral agiu sobre o veículo com ele em movimento, fazendo com que conseguisse pousar com maior precisão no ponto planejado.

Figura 29 – Execução do teste de aplicação de rotina de decolagem, translação e pouso



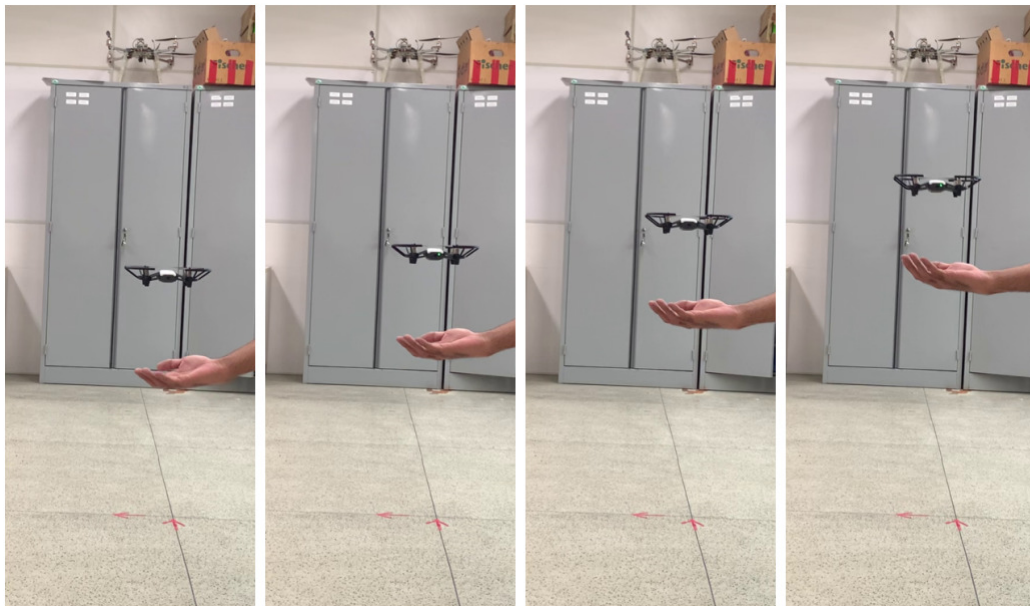
Fonte: Elaborado pelo autor.

5.4.3 Estabilização

A plataforma se comportou de maneira satisfatória operando em modo estabilizado na maior parte do período de utilização. A correção de posicionamento é feita de modo suave pela ação dos rotores operados pelo controlador de voo DJI. Além disso, o sistema de sensores infravermelho que incorpora o VPS atua de modo a manter uma diferença de altitude com objetos abaixo da aeronave, de modo a manter a segurança em voo.

Assim, ao induzir a detecção de um corpo em modo de voo pairado, ela mantém a mesma distância em altitude do objeto, tracionando de modo avançar no deixo z na medida em que se tenta aproximar o corpo ao sensor, como observado na Figura 30, e voltando a manter-se em modo estabilizado após retornar ao estado de segurança.

Figura 30 – Atuação do sensor infravermelho no sistema VPS

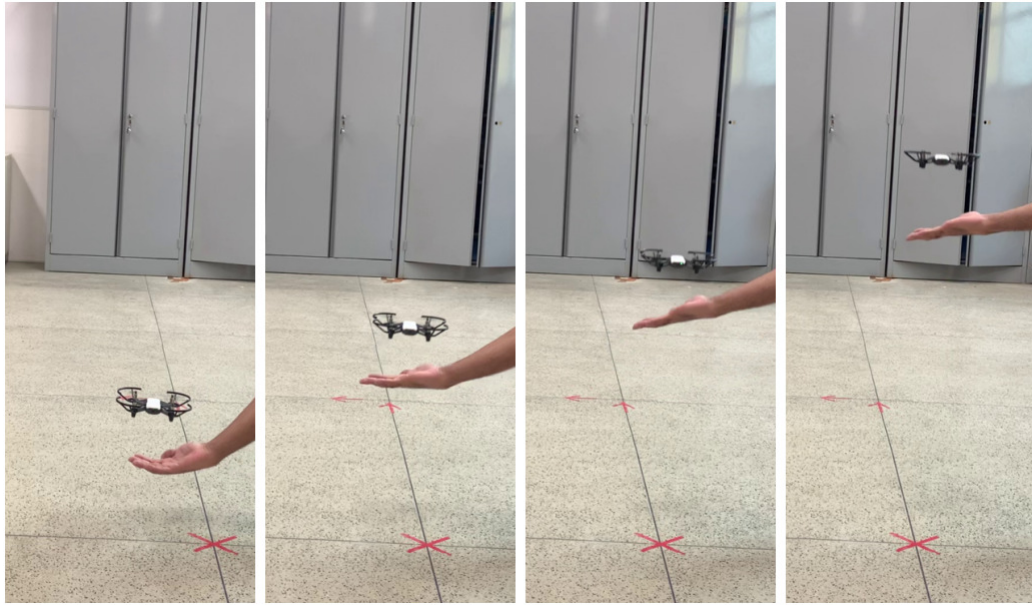


Fonte: Elaborado pelo autor.

No entanto, ao realizar uma aproximação mais agressiva, induzindo a atuação do robô com a detecção do corpo pelo sistema, em alguns casos ocorreu a variação de posição em mais de um eixo de forma aleatória (Figura 31).

O mesmo ocorreu ao proporcionar correntes de ar fortes no corpo do dispositivo, o que se deve principalmente a baixa tração desenvolvida pelos rotores e a falta de sensores de posicionamento global.

Figura 31 – Atuação aleatória do sensor infravermelho no sistema VPS



Fonte: Elaborado pelo autor.

Contudo, apesar de algumas ocorrências dessa movimentação aleatória, logo em seguida o dispositivo se comporta de forma a estabilizar-se de forma pairada, aguardando um comando do usuário para sair dessa posição ou uma nova detecção no sensor infravermelho.

6 CONSIDERAÇÕES FINAIS

O presente trabalho apresentou o desenvolvimento de um método de utilização de uma aeronave controlada pelo *Robot Operating System* (ROS) como plataforma para práticas educacionais de robótica móvel em laboratório do curso de Engenharia Mecânica, conforme previsão do Projeto Pedagógico do curso.

Foi possível apresentar a cinemática envolvendo a utilização de Aeronaves Remotamente Pilotadas, bem como a legislação vigente a qual se regulamenta a operação desses dispositivos e apontando fatores os quais favorecem a escolha do referido equipamento para uso em práticas universitárias.

Além disso, houve a estruturação eficaz do ROS como ferramenta de desenvolvimento dos algoritmos utilizados para a comunicação com o hardware do dispositivo escolhido, permitindo não só a fácil operação pelo usuário não familiarizado com a plataforma, mas também a obtenção de informações de maneira remota em tempo real.

No entanto, ao analisar a execução de rotinas na máquina de estado durante a utilização do sistema, verificou-se a tendência de armazenamento de pacotes de informações recebidas pelo dispositivo em voo, para scripts de repetições com um número maior de entradas de código.

Ainda, os resultados da aplicação da aeronave em voo no ambiente do laboratório permitiram verificar efeitos da ação da gravidade e inércia, interferindo no posicionamento global do Tello durante a execução de movimentos, principalmente no que se refere a tendência de manter o movimento retilíneo após parar de mandar o comando de operação, além da dificuldade de se manter em modo pairado quanto menor a altitude, o que se deve a ação do efeito solo.

7 TRABALHOS FUTUROS

Diante do trabalho apresentado, no que se refere a verificada viabilidade da utilização da aeronave, aplicada a ela um sistema operacional robótico de código aberto, diversas são as possibilidades de linha de pesquisa para utilização da plataforma para desenvolvimento de estudos em laboratório, aplicados ao curso de Engenharia Mecânica.

Como exemplo, a sua utilização para para a aplicação de mecanismos de inteligência artificial, voltadas principalmente para processos de *machine learning* que possam possibilitar a aplicação de veículos robóticos aéreos na logística e execução de tarefas em armazéns e indústrias, bem como auxílio na fiscalização e acompanhamento operacional de obras e eventos ambientais catastróficos.

Além disso, participação em eventos de competição, nacionais e internacionais, na forma de seguidores de linha, robótica cooperativa, localização e resgate, entre outros.

Desta forma, o pesquisador na área terá uma experiência de aplicação prática com um dispositivo em ascensão no mercado, implicando a sua capacitação em relação ao mercado de trabalho e respectivo desenvolvimento tecnológico almejado na indústria de robótica avançada.

REFERÊNCIAS

- Auterion Enterprise PX4. **AUTERION**, 2020. Disponível em: <https://auterion.com/drone-manufacturers/enterprise-px4/>. Acesso em: 16 out. 2022.
- Classes de Drones. **ANAC**, 2022. Disponível em: <https://www.gov.br/anac/pt-br/assuntos/drones/classes-de-drones>. Acesso em: 02 ago. 2022.
- DRONES. **ANAC**, 2022. Disponível em: <https://www.gov.br/anac/pt-br/assuntos/drones>. Acesso em: 01 ago. 2022.
- Drones Classe 3. **ANAC**, 2022. Disponível em: <https://www.gov.br/anac/pt-br/assuntos/drones/aeromodelos-ou-rpa-com-peso-maximo-de-decolagem-de-ate-250g>. Acesso em: 02 ago. 2022.
- Drones para logística 4.0. **FIEPR**, 2019. Disponível em: <https://www.fiepr.org.br/boletins-setoriais/13/especial/drones-para-logistica-40-2-32029-391096.shtml>. Acesso em: 21 ago. 2022.
- EMRAN, B. J.; NAJJARAN, H. A review of quadrotor: An underactuated mechanical system. **Annual Reviews in Control**, v. 46, p. 165–180, 2018. ISSN 1367-5788. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1367578818300932>.
- FERDAUS, M. M.; ANAVATTI, S. G.; PRATAMA, M.; GARRATT, M. A. Towards the use of fuzzy logic systems in rotary wing unmanned aerial vehicle: a review. **Artificial Intelligence Review**, Springer, v. 53, p. 257–290, 1 2020. ISSN 15737462.
- GIERNACKI, W.; RAO, J.; SLADIC, S.; BONDYRA, A.; RETINGER, M.; ESPINOZA-FRAIRE, T. Dji tello quadrotor as a platform for research and education in mobile robotics and control engineering. In: IEEE. **2022 International Conference on Unmanned Aircraft Systems (ICUAS)**. [S.l.], 2022. p. 735–744.
- KHAMVILAI, T.; DUNHAM, J.; FERON, E.; JOHNSON, E. N. Avionics of aerial robots. **Current Robotics Reports**, v. 2, p. 113–124, 2021.
- LEÃO, L. M. **Metodologia do estudo e pesquisa: facilitando a vida dos estudantes, professores e pesquisadores**. [S.l.]: Editora Vozes, 2016.
- LOPES, R. de A.; FORTES, A. F. C.; PEDROSA, A. Í. R.; ARAÚJO, A. G. F. de; ALVES, R. M. **PROJETO PEDAGÓGICO DO CURSO DE ENGENHARIA MECÂNICA NA MODALIDADE PRESENCIAL CAMPUS TERESINA CENTRAL**. 2018. Disponível em: <https://www.ifpi.edu.br/cursos/documentos-dos-cursos/ppc/PPCEMETSa.pdf>. Acesso em: 23 ago. 2022.
- MACENSKI, S.; FOOTE, T.; GERKEY, B.; LALANCETTE, C.; WOODALL, W. Robot operating system 2: Design, architecture, and uses in the wild. **Science Robotics**, v. 7, n. 66, p. eabm6074, 2022. Disponível em: <https://www.science.org/doi/abs/10.1126/scirobotics.abm6074>.
- Nova engenharia passa por inteligência artificial e robótica. **CREA-MG**, 2021. Disponível em: <https://www.crea-mg.org.br/comunicacao/sala-de-imprensa/release/Nova%20engenharia%20passa%20por%20intelig%C3%Aancia%20artificial%20e%20rob%C3%B3tica>. Acesso em: 21 ago. 2022.

Qual a diferença entre drone, VANT e RPAS?. **DECEA**, 2019. Disponível em: <https://ajuda.decea.mil.br/base-de-conhecimento/qual-a-diferenca-entre-drone-vant-e-rpas/>. Acesso em: 04 ago. 2022.

RAO, J.; LI, B.; ZHANG, Z.; CHEN, D.; GIERNACKI, W. Position control of quadrotor uav based on cascade fuzzy neural network. **Energies**, v. 15, 2022. ISSN 19961073.

ROS Wiki: Documentation. **OPEN ROBOTICS**, 2022. Disponível em: <https://www.ros.org/blog/why-ros/>. Acesso em: 15 ago. 2022.

RUBBESTAD, G.; SÖDERQVIST, W. **Hacking a Wi-Fi based drone**. 2021. Disponível em: <https://www.diva-portal.org/smash/record.jsf?pid=diva2:1586253>. Acesso em: 25 ago. 2022.

Tello User Manual. **Ryze Robotics**, 2018. Disponível em: https://dl-cdn.ryzeroobotics.com/downloads/Tello/20180404/Tello_User_Manual_V1.2_EN.pdf. Acesso em: 25 ago. 2022.

Teresina Central abre inscrições para curso de Robótica para Engenheiras. **IFPI**, 2022. Disponível em: <https://www.ifpi.edu.br/teresinacentral/noticias/teresina-central-abre-inscricoes-para-curso-de-robotica-para-engenheiras>. Acesso em: 01 ago. 2022.

The DIU standardizes on PX4 and MAVLink for small commercial drones. **AUTERION**, 2020. Disponível em: <https://auterion.com/the-diu-standardizes-on-px4-and-mavlink-for-small-commercial-drones/>. Acesso em: 16 out. 2022.

Understanding nodes. **ROS**, 2022. Disponível em: <https://docs.ros.org/en/foxy/Tutorials/Beginner-CLI-Tools/Understanding-ROS2-Nodes/Understanding-ROS2-Nodes.html>. Acesso em: 12 ago. 2022.

Understanding topics. **ROS**, 2022. Disponível em: <https://docs.ros.org/en/foxy/Tutorials/Beginner-CLI-Tools/Understanding-ROS2-Topics/Understanding-ROS2-Topics.html>. Acesso em: 12 ago. 2022.

VYAS, K. **A Brief History of Drones: The Remote Controlled Unmanned Aerial Vehicles (UAVs)**. 2020. Disponível em: <https://interestingengineering.com/a-brief-history-of-drones-the-remote-controlled-unmanned-aerial-vehicles-uavs>. Acesso em: 01 ago. 2022.

WANG, R.; LIU, J. Trajectory tracking control of a 6-dof quadrotor uav with input saturation via backstepping. **Journal of the Franklin Institute**, Elsevier Ltd, v. 355, p. 3288–3309, 5 2018. ISSN 00160032.