UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
PROJETO DE GRADUAÇÃO

Ricardo da Mota Salvador

Módulo de Voo para Quadrimotor Baseado em Visão Computacional e ROS para Cumprimento de Tarefas Autônomas

Vitória-ES

Junho/2018

Ricardo da Mota Salvador

Módulo de Voo para Quadrimotor Baseado em Visão Computacional e ROS para Cumprimento de Tarefas Autônomas

Parte manuscrita do Projeto de Graduação do aluno Ricardo da Mota Salvador, apresentado ao Departamento de Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Engenheiro Eletricista.

Vitória-ES

Junho/2018

Ricardo da Mota Salvador

Módulo de Voo para Quadrimotor Baseado em Visão Computacional e ROS para Cumprimento de Tarefas Autônomas

Parte manuscrita do Projeto de Graduação do aluno Ricardo da Mota Salvador, apresentado ao Departamento de Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Engenheiro Eletricista.

Aprovado em 25, de Junho de 2018.

COMISSÃO EXAMINADORA:

Profa. Dra. Raquel Frizera Vassallo
Universidade Federal do Espírito Santo
Orientadora

Prof. M.Sc. Leonardo de Assis Silva
Instituto Federal do Espírito Santo
Coorientador

Prof. M.Sc. Fabrício Bortolini de Sá
Instituto Federal do Espírito Santo
Examinador

M.Sc. Héctor Ignacio Perez Azpúrua
Universidade Federal de Minas Gerais
Examinador

Vitória-ES

Junho/2018

AGRADECIMENTOS

Esse é o momento em que respira-se fundo e todas as memórias são trazidas de volta, alinhadas cronologicamente, e analisadas como se eu estivesse em um plano superior, como se todo os anos da faculdade estivessem colocados na minha mão e eu, assim, possa contemplar cada instante.

Os anos durante o curso de Engenharia Elétrica não foram fáceis. Não posso reclamar, não teria como ser diferente. Entretanto, pra quem passou fazendo contas impossíveis durante tanto tempo, não é complicado somar todos os momentos bons e os ruins e perceber que não está na mesma ordem de grandeza tudo de especial e gratificante que aconteceu desde que me matriculei na UFES.

Dos agradecimentos deixo os primeiros para minha família, e essa é a única parte deste texto que consigo mensurar o grau de importância. Minha família foi o mais importante. Cada um, com suas características e capacidades, foi capaz de propiciar, ao final, um ambiente em que encontrei sabedoria pra tudo, limite quando precisei, força nos momentos mais intensos e carinho, carinho ilimitado.

Agradeço também a Raquel Frizera Vassallo, que além de figurar como a melhor professora que tive, também faz as vezes de uma pessoa da minha família, pelo menos assim considero. Essa é a melhor descrição, talvez a única possível. Obrigado Raquel.

Obrigado a meus hermanos Fifi e Messi, vulgo Felippe Mendonça e Leonardo Silva, por tantos empreendimentos de sucesso. Vida longa à 3M hahaha.

Tenho que registrar aqui também minha gratidão a todos do laboratório, com quem tive o prazer de trabalhar e o privilégio de chamar de amigos. À geração mais recente deixo a observação dos inúmeros ”fires”, foram sextas memoráveis (não pra todos kkk). Aos ”blindados” Pedro Catelan e Walastan Melo agradeço por toda a parceria e sapiência, vocês são demais.

Por fim, dedico e agradeço a todos que fizeram parte, direta ou indiretamente, dessa etapa da minha vida.

RESUMO

Norteada por uma necessidade de maior desenvolvimento no ramo da robótica aérea, esta pesquisa visa contribuição em situações onde é inviável ou muito custoso o envio de comandos de maneira manual para um veículo aéreo não tripulado como, por exemplo, missões onde há necessidade de se seguir um alvo móvel, como outro robô, decolagem ou aterrizagem em locais específicos, identificação de alvos em regiões de grandes extensões ou mesmo operações de resgate. Outra aplicação possível para o trabalho se dá em situações onde deseja-se diminuir a interferência humana no comando dos equipamentos, se aproximando do assunto abordado na concepção de Cidades Digitais e IoT.

Portanto, o presente trabalho tem como objetivo o desenvolvimento de um controle com realimentação visual para quadricópteros, possibilitando autonomia de movimentos e a tomada de decisões a partir da detecção, durante o voo, de padrões visuais e extração de informações como posição e orientação deles. O veículo deverá ser capaz de capturar, processar e tomar atitudes de acordo com o padrão observado através das imagens capturadas.

Apesar de objetivo claro, o projeto possui desenvolvimento não tão óbvio, sendo dividido em três frentes de pesquisa principais: a captura e reconhecimento de padrões, o processamento das informações obtidas para a tomada de decisões, e a fusão das duas primeiras etapas com o controlador de voo nativo do equipamento.

Palavras-chave: Visão Computacional, Controle com Realimentação Visual, ROS, Veículo Aéreo Não Tripulado.

LISTA DE FIGURAS

Figura 1 – O novo quadricóptero do laboratório. 11
Figura 2 – Forças presentes na asa em movimento. 15
Figura 3 – Componentes de movimento do VANT (*roll*, *pitch* e *yaw*). 16
Figura 4 – Representação do registro da cena no modelo de câmera pinhole (princípio da câmara escura). 17
Figura 5 – Diagrama de blocos do sistema de controle com realimentação visual. . 19
Figura 6 – Etapas de montagem. 21
Figura 7 – Estrutura e modelagem. 21
Figura 8 – Distribuição de esforços no trem de pouso. 22
Figura 9 – Modelos de quadrimotores desenvolvidos. 22
Figura 10 – Equipamentos embarcados. 23
Figura 11 – O novo quadricóptero do laboratório. 24
Figura 12 – Trecho do esquema de funcionamento do sistema - ROS. 25
Figura 13 – Esquema completo de funcionamento do sistema - ROS. 26
Figura 14 – Padrões utilizados no trabalho. 29
Figura 15 – RaspiCam conectada à Raspiberry Pi. 30
Figura 16 – Gimbal utilizada. 31
Figura 17 – Dimensões do padrão adotado. 32
Figura 18 – Ilustração do cenário real da situação abordada com cada referencial da cena. 35
Figura 19 – Transformações para mudança de referencial do sistema A para o sistema de coordenadas C da câmera de um quadrimotor. 36
Figura 20 – Ilustração do controle de posição e orientação final do quadrimotor. . . 37
Figura 21 – Máquina de estados da tarefa autônoma. 39
Figura 22 – (i) No primeiro momento, o padrão ainda não está no campo de visão da câmera. (ii) Ao ser detectado pela primeira vez, (iii) o VANT tenta se alinhar ao padrão e passa a segui-lo. 40
Figura 23 – Distância definida pelo usuário. 41
Figura 24 – Padões utilizados no trabalho. 41
Figura 25 – Inicialmente, (i) o VANT está seguindo o padrão A. (ii) Após detectar o padrão B, (iii) o VANT deixa de se movimentar em busca do marcador A e da início ao pouso. 42
Figura 26 – Publicações de posição iniciais. 44
Figura 27 – Detecção do marcador visual. O campo **id** indica qual o padrão está sendo detectado, seguido de **pose**, que contém informações espaciais do marcador. 45

Figura 28 – Variação de intenção de posicionamento conforme a posição do padrão na imagem. 45
Figura 29 – Perda do padrão (a última posição continua sendo enviada). 46
Figura 30 – Diagnóstico instantâneo geral do VANT. 47
Figura 31 – Odometria do robô terrestre, posição do VANT e trajetória esperada para a rota circular. 50
Figura 32 – Odometria do robô terrestre, posição do VANT e trajetória esperada para a rota em oito. 50
Figura 33 – Odometria do robô terrestre, posição do VANT e trajetória esperada para a rota em reta. 51
Figura 34 – Odometria do robô terrestre, posição do VANT e trajetória esperada para a rota em zigzag. 52
Figura 35 – Rota esperada, em verde; rota do VANT, dada pelos dados de GPS, em vermelho; e rota estimada pelas imagens da câmera, em azul. 53
Figura 36 – Comportamento do ângulo de *yaw* obtido a partir das informações extraídas do VANT durante a trajetória de Círculo e ZigZag. 54
Figura 37 – Trajetória percorrida. 56
Figura 38 – Relação entre altura do veiculo e FOV do sistema de visão. 57
Figura 39 – Instante anterior à primeira detecção e instante da primeira detecção. . 57
Figura 40 – Comportamento do ângulo de YAW durante o experimento. 58
Figura 41 – Dinâmica do experimento na missão de pouso. (i) Na primeira volta, o marcador de pouso B está oculto. (ii) Na segunda volta, o marcador B é revelado. 59

LISTA DE TABELAS

Tabela 1 – Taxa de detecção do padrão na imagem e tempo médio de processamento para resoluções de $480p$ e $720p$. 49
Tabela 2 – Erro de trajetória a partir dos dados do GPS para o experimento de rota circular. 51
Tabela 3 – Distâncias euclidianas entre posição real e esperada do VANT nas três componentes e, por último, tridimensionalmente. 52

SUMÁRIO

1 INTRODUÇÃO . . . 10
1.1 Justificativa . . . 12
1.2 Objetivos . . . 13
1.2.1 Objetivo Geral . . . 13
1.2.2 Objetivos Específicos . . . 14

2 EMBASAMENTO TEÓRICO . . . 15
2.1 VANTs . . . 15
2.2 Modelo de câmera . . . 16
2.3 Controle com Realimentação Visual . . . 18

3 DESCRIÇÃO DA PLATAFORMA . . . 20
3.1 Estrutura Física do Veículo . . . 20
3.2 Sistema de Software Embarcado . . . 23
3.2.1 ROS . . . 24
3.2.2 MAVLink . . . 27

4 AQUISIÇÃO E PROCESSAMENTO DE IMAGENS . . . 28
4.1 Captura das imagens . . . 29
4.2 Processamento das imagens . . . 31

5 PROCESSAMENTO DOS DADOS . . . 34
5.1 Atualização de posicionamento do VANT a partir de dados do ar_track_alvar . . . 34
5.2 Missões . . . 38
5.2.1 Missão de busca inicial . . . 39
5.2.2 Missão de seguimento de padrão . . . 40
5.2.3 Missão de pouso . . . 42

6 EXPERIMENTOS E RESULTADOS . . . 43
6.1 Ativação do sistema . . . 43
6.2 Avaliação do sistema de captura de imagens e detecção dos marcadores . . . 43
6.3 Avaliação da intenção de posicionamento . . . 44
6.4 Análise da perda momentânea do padrão e eventuais outliers . . . 44
6.5 Tomada de decisão conforme o padrão . . . 46
6.6 Seguimento de marcador móvel com realimentação visual . . . 48

6.7 **Tarefa (as três missões)** **55**

7 **CONCLUSÃO E TRABALHOS FUTUROS** **60**

REFERÊNCIAS BIBLIOGRÁFICAS **62**

1 INTRODUÇÃO

Veículos Aéreos Não Tripulados, ou VANTs, têm sido muito utilizados nos últimos anos para diversas aplicações, principalmente devido à sua visão privilegiada e à sua versatilidade quanto aos ambientes de atuação.

O emprego da tecnologia abrange as mais diversas áreas, sendo que as aplicações mais comuns são: patrulhamento aéreo (e.g. utilização para equipes de resgate) (RESTAS, 2015; APVRILLE; TANZI; DUGELAY, 2014); inspeção de equipamentos e estruturas (e.g. geradores eólicos, e linhas de transmissão) (MÁTHÉ; BUŞONIU, 2015); monitoramento rural ou urbano (JENSEN; ZELLER; APAN, 2011); fiscalização ambiental (FARIA; COSTA, 2015); e auxílio de equipes em terra ou veículos terrestres em situações onde eles possuem limitações de acesso devido à presença de obstáculos, às características do terreno, ao limite visual em terra, ou ao risco envolvido na aproximação do local de interesse (SA et al., 2014).

Tais aplicações, geralmente, requerem informações visuais para sua realização, exigindo do VANT uma câmera embarcada e, nesses casos, além do interesse nas imagens capturadas para a realização da tarefa, a informação visual pode ser utilizada para o controle do próprio veículo. As imagens podem servir de guia para a navegação do VANT (e.g. seguimento de rios, ruas, e linhas de transmissão), para detectar marcadores que possam indicar outras tarefas a serem realizadas, ou para encontrar locais de pouso. Ou seja, mesmo que a aplicação não demande a aquisição de imagens, a informação visual poderá ser importante para o VANT, principalmente quando a aplicação precise ser executada de forma completamente autônoma.

Além disso, dependendo das especificações de cada tarefa quanto a ambiente, tempo de voo e capacidade de carregamento, um tipo diferente de VANT pode ser demandado. Atualmente, os VANTs mais utilizados são os planadores e os veículos aéreos de quatro ou mais rotores (quadricópteros, hexacópteros ou octocópteros), mais conhecidos como drones. Os planadores apresentam a vantagem de percorrerem longas distâncias, uma vez que dependem menos de bateria para voarem. Já os quadricópteros são mais estáveis, apresentam a possibilidade de decolar e aterrissar em ambientes de pequenas dimensões, o que facilita a realização de voos e experimentos, principalmente quando se necessitam imagens aéreas estacionárias. Sua principal desvantagem é a sua autonomia de voo.

Nesse contexto, o grupo de pesquisa coordenado pela Profa. Dra. Raquel Frizera Vassallo, orientadora deste trabalho, já vem desenvolvendo alguns trabalhos com quadricópteros utilizando visão computacional. Uma dissertação de mestrado com cooperação de robôs

Figura 1 – O novo quadricóptero do laboratório.

Fonte: Produção do próprio autor.

foi finalizada em 2014 (SA et al., 2014), e outra em 2016 (AMORIM et al., 2015), cujo objetivo era sobrevoar regiões e gerar mosaicos com as imagens aéreas para monitoramento ambiental pelo Corpo de Bombeiros de Vitória.

Recentemente, o grupo concluiu a montagem de um novo quadricóptero, mostrado na Figura 1, capaz de voar em ambientes externos e suportar ventos de até 50 km/h.

Além dos sensores tradicionais (acelerômetro, giroscópio, barômetro, GPS, ultrassom), foi instalado nesse VANT um suporte para câmera digital do tipo gimbal, que compensa o movimento do robô aéreo e estabiliza a câmera para captura de imagens com melhor qualidade e menos inclinação.

Assim, este trabalho tem como objetivo desenvolver ferramentas de controle baseado em realimentação visual para o quadricóptero mostrado na Figura 1, tendo como principais funcionalidades: o seguimento de um alvo para que o VANT possa se manter sempre a uma distância predeterminada, e a aterrizagem de forma autônoma de acordo com um marcador visual.

O processamento das imagens pelo VANT também deverá ser realizado rapidamente para que seja possível seguir alvos rápidos, visto que um processamento lento implica em uma atuação tardia, o que não permitirá acompanhar alvos velozes, que fugirão do campo de visão do drone em algum momento.

Para isso, será utilizado um framework para robótica chamado ROS (Robot Operating

System) (ROS, 2017). Ele se encontra mundialmente em simulações e controle de diversos tipos de robôs. Uma grande vantagem dele é a quantidade de pacotes (conjunto de funções pré-programadas) disponíveis e já consolidados, sendo alguns voltados para visão computacional (e.g. *ar_track_alvar*). Dessa forma, é possível desenvolver as aplicações: detecção de marcadores, cálculo de homografias necessárias para seguimento dos marcadores e controle baseado em realimentação visual do veículo.

1.1 Justificativa

Como já comentado, a quantidade de aplicações utilizando drones vem crescendo consideravelmente nos últimos tempos. Esse aumento se deve às possibilidades que essa plataforma aérea proporciona para as pesquisas em robótica. Mesmo que aviões e helicópteros também possuem uma visão privilegiada e mecanismos suficientes para a realização de diversas tarefas, o tamanho e preço reduzidos dos VANTs e o fato de não haver tripulantes a bordo, tornam os VANTs mais adequados para tarefas em ambientes hostis onde os métodos já existentes têm dificuldade ou simplesmente não são capazes de atuar.

Entretanto, muitas vezes as possibilidades de atuação destes equipamentos são limitadas quando o equipamento demanda controle manual humano. Para contornar essa questão, muitos trabalhos propõem módulos autônomos de voo, de forma que o próprio veículo tenha capacidade de comando e tomada de decisão a partir de estímulos do ambiente. Isso expande as áreas de emprego dessa tecnologia, justificando o aumento do foco no assunto observado nos últimos tempos, tanto no âmbito acadêmico e empresarial quanto no social.

Na maioria dos casos em que é desejado um controle autônomo para VANTs, as informações visuais são importantes fontes de dados, os quais são processados de forma a se extrair característica pretendidas da cena capazes de proporcionar uma tomada de decisão pelo próprio equipamento. Em (ZAKARIA et al., 2016), por exemplo, é feito um controle servo visual por meio de fluxo ótico, permitindo o seguimento de objetos alvos. Entretanto, o volume de dados para processamento, geralmente, demanda uma base de apoio em terra.

Uma abordagem diferente é a detecção de marcadores que possam indicar tarefas a serem realizadas (CARVALHO et al., 2016) ou atitudes a serem tomadas (NITSCHKE, 2014). Nesses trabalhos, utiliza-se o cálculo de homografia a partir dos marcadores, retirando-se informações espaciais para planejar a movimentação do veículo. Carvalho et al. (2016) apresenta uma simulação para pouso conforme a detecção da marca. Já Nitschke (2014) realiza um controle com sistema multimarcas em um ambiente interno, com processamento em base terrestre.

Ter a possibilidade de um controle autônomo torna o VANT um equipamento mais versátil, facilitando sua inserção em projetos maiores e mais complexos, como no caso de cidades digitais (BARRETO; SOUZA, 2017) e IoT (MOTLAGH; BAGAA; TALEB, 2017). O veículo deixa de necessitar de comandos diretos ou indiretos de humanos e passa a ser uma ferramenta autossuficiente na execução de tarefas.

Além disso, é importante comentar sobre o fator ambiente. Em (BUTZKE et al., 2016), é proposto um controle autônomo, valendo-se de marcadores fixos e cooperação entre robô aéreo e terrestre. Os resultados dessa pesquisa são interessantes, entretanto, contemplam somente ambientes internos. Butzke et al. (2016) explicitaram em seu trabalho que o sistema proposto foi desenvolvido com o foco em ambientes internos. Isso evidencia a necessidade de abordagens específicas quanto ao controle para grandes áreas e/ou sob influência de condições adversas do próprio ambiente.

Neste contexto, este trabalho propõe um módulo de controle de voo multi-tarefa capaz de realizar missões autônomas (e.g. seguir alvo móvel e pousar), gerando comandos de posição e orientação final para o veículo em relação a um referencial local fixo. Esta característica permite o uso do equipamento em ambientes e referenciais diversos, facilitando a cooperação com outros robôs ou sua integração a um sistema maior, como uma cidade inteligente. Também é considerado objetivo do trabalho que o módulo de voo seja capaz de lidar com imperfeições intrínsecas ao sistema, como a baixa precisão do GPS em determinadas situações, e influências externas do ambiente, como a intensa exposição de luz sobre o padrão em dias ensolarados e/ou a incidência de ventos moderados sobre o veículo. Desta forma, a plataforma estará apta a atuar em ambientes externos.

1.2 Objetivos

1.2.1 Objetivo Geral

Este trabalho tem por objetivo elaborar um sistema que possibilite o voo e a tomada de decisões de forma totalmente autônoma de um quadricóptero por meio de informações visuais coletadas e processadas no próprio veículo. A ideia é acrescentar ao quadricóptero um modo de voo que o permita se locomover em busca de um alvo específico ou a detecção de padrões visuais para a tomada de decisões, como o pouso ou ações predeterminadas, sem a necessidade de comando externo.

1.2.2 Objetivos Específicos

- Revisão bibliográfica acerca das ferramentas de visão computacional disponíveis e análise das que melhor se adequam à robótica aérea para verificar as possíveis vertentes do projeto;
- Estudo do funcionamento do ROS;
- Obtenção de imagens estáveis e adequadas, contendo a maior qualidade de informação possível para a extração das características relevantes;
- Desenvolvimento do algoritmo que processará as imagens capturadas;
- Desenvolvimento do algoritmo que processará os dados de posição e detecção dos padrões para o controle do veículo;
- Estudo e adequação do pacote MAVROS ao sistema;
- Implantação do programa elaborado em um microcomputador compatível com a capacidade de carga do VANT para processamento embarcado;
- Montagem da infraestrutura, contemplando a interligação do VANT ao microcomputador a ser utilizado abordo;
- Utilização do sistema em um quadricóptero real.

2 EMBASAMENTO TEÓRICO

Será apresentada neste capítulo a base teórica necessária para o desenvolvimentos das etapas do projeto proposto.

2.1 VANTs

VANTs são equipamentos aéreos e, portanto, devem seguir a dinâmica e propriedades de corpos em movimento imersos em fluidos (e.g. ar). Independente do tipo de estrutura do veículo, suas asas devem ser capazes de proporcionar sustentação levando em consideração o movimento e aerodinâmica envolvidos.

Mais precisamente, quando o fluido passa pelas asas, são criadas forças através dela (empuxo, sustentação e arrasto) de acordo com a Figura 2.

Figura 2 – Forças presentes na asa em movimento.

Fonte: Bastos (2011).

A tração impulsiona a aeronave, o arrasto é a componente da força aerodinâmica na mesma direção e sentido do vento relativo, e a sustentação é a componente da força aerodinâmica perpendicular ao vento relativo, que compensa a força peso (SAMPAIO, 2006).

A mecânica envolvida é mais intuitiva e fica mais explícita nos veículos de asas fixas, sendo que os de asas rotativas utilizam do mesmo princípio, salvas algumas peculiaridades. Nesses últimos, é possível observar a produção tanto de força de sustentação quanto de tração a partir do movimento das hélices. Isso se deve à diferença de pressão gerada pelo ar passando através do perfil da asa, levando-se em consideração seu ângulo de ataque e velocidade de rotação. (HOFFMANN et al., 2007).

Nos veículos desse tipo, o ângulo de ataque das hélices é fixo, portanto, a sustentação e variação da direção do veículo estão atreladas à velocidade de rotação referente aos seus motores, o que gera momento linear em relação ao centro de gravidade do equipamento. Por fim, o VANT tem sua movimentação resumida em três componentes, movimento de inclinação ($PITCH$), rotação ($ROLL$) e guinada (YAW), como mostrado na Figura 3. (HOFFMANN et al., 2007).

Figura 3 – Componentes de movimento do VANT (*roll*, *pitch* e *yaw*).

Fonte: Cuong (2017).

Dessa forma, uma tarefa de movimentação, partindo da posição atual do drone para uma posição desejada, tem por base o controle dessas três componentes até que o VANT alcance a posição final.

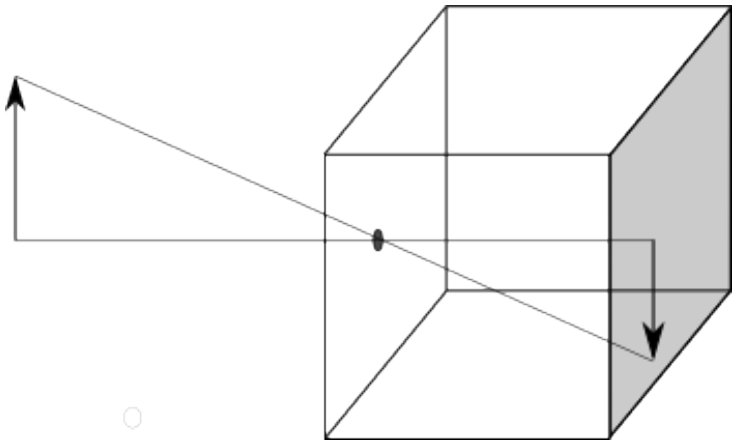
2.2 Modelo de câmera

A câmera é o cerne do controle servo visual que será embarcado no VANT. Além disso, o sistema evolverá a estimação da correspondência de pontos tridimensionais do ambiente a partir de pontos nas imagens capturadas pela câmera. Ou seja, será necessário entender e modelar matematicamente o funcionamento da mesma.

O modelo de câmera que será utilizado neste trabalho será o modelo de câmera Pinhole (HARTLEY; ZISSERMAN, 2003). Ele representa a câmera como um modelo de projeção pelo princípio da câmara escura. Nesse princípio, os raios de luz passam por uma pequena abertura de uma caixa escura, incidindo de maneira invertida na parte interna oposta dela, assim como mostrado na Figura 4.

Esse modelo é amplamente difundido, visto sua maneira simples de equacionamento da teoria envolvida para realizar a transformação das informações visuais da cena em pixels,

Figura 4 – Representação do registro da cena no modelo de câmera pinhole (princípio da câmara escura).



Fonte: Produção do próprio autor.

e pode ser conferido na Equação 2.1.

$$Z \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} fs_x & s_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix}}_{\text{parâmetros intrínsecos } (\mathbf{K})} \underbrace{\begin{bmatrix} \mathbf{R}_{3\times3} & \mathbf{t}_{3\times1} \end{bmatrix}}_{\text{parâmetros extrínsecos}} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.1}$$

Como pode ser observado na Equação 2.1, a transformação de um ponto tridimensional da cena em um pixel na imagem é feita em duas etapas. A primeira, dada pelos parâmetros extrínsecos, relaciona o referencial da câmera e o referencial global adotado, sendo descrita como uma rotação $\mathbf{R}$ e uma translação $\mathbf{t}$. Já a segunda, compreende a forma com que esse modelo associa unidades métricas da cena com pixels na imagem gerada pela matriz de parâmetros intrínsecos $\mathbf{K}$.

De forma resumida, a câmera projeta pontos tridimensionais presentes na cena observada no plano da imagem, perdendo a coordenada Z. Apesar de perder essa informação de profundidade, o sistema ainda é capaz de recuperar o ponto tridimensional a menos de um fator de escala. Uma maneira de resolver isso pode ser a estimação da coordenada Z pela fusão dessa informação com as de outros sensores embarcados no VANT (e.g. ultrassom, gps) e/ou pelo processo de homografia, tendo como informações já conhecidas as medidas do alvo.

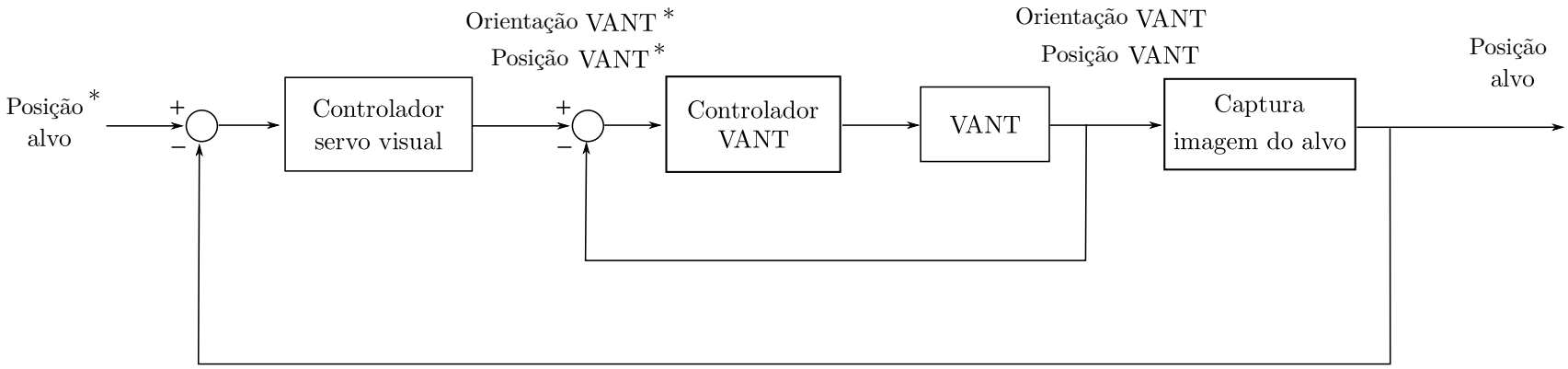
2.3 Controle com Realimentação Visual

A partir das imagens obtidas pela câmera, o sistema a ser desenvolvido (Controle com Realimentação Visual) deve identificar o marcador visual pretendido, estimando sua posição e orientação.

No contexto de seguimento de alvo móvel, o módulo tem a função de reconhecer o alvo na imagem e calcular os esforços de controle, ou seja, posição e orientação finais do VANT de maneira que o alvo seja novamente representado em um local pré-definido na imagem (e.g. centro) no próximo ciclo de captura, configurando uma malha de controle da movimentação do veículo com base na realimentação visual, como apresentado na Figura 5 por meio de um diagrama de blocos do sistema.

Como pode-se notar na Figura 5, os valores de posição e orientação finais do VANT são enviados para o controlador nativo do VANT para que ele comande os motores de forma desejada.

Figura 5 – Diagrama de blocos do sistema de controle com realimentação visual.



Fonte: Produção do próprio autor.

3 DESCRIÇÃO DA PLATAFORMA

Este trabalho foi desenvolvido no laboratório de pesquisa da UFES, Cisne B - Viros, onde são realizados projetos voltados à área de Visão Computacional - Espaço Inteligente e Robótica Aérea.

No ramo da Robótica Aérea, é possível destacar alguns trabalhos como os desenvolvidos por Sa et al. (2014) e Amorim et al. (2015). Neles, percebe-se que quando se objetiva realizar pesquisas de inovação e desenvolvimento científico, soluções comerciais de VANTs muitas vezes não são as mais indicadas. Isso se deve ao fato de que essas soluções limitam a customização e programação da plataforma pelo usuário.

Quando se trata de desenvolvimento científico, principalmente na área de VANTs, é crucial ter autonomia na criação do equipamento e na inserção de ferramentas (sensores, funcionalidades, modos de operação), possibilitando ao usuário fazer ajustes e alterações específicas nas mais diversas partes do veículo.

Por isso, durante a segunda iniciação científica do autor, um equipamento próprio para as aplicações do laboratório foi projetado e montado, desde a estrutura física até a eletrônica. Algumas etapas de montagem podem ser conferidas na Figura 6.

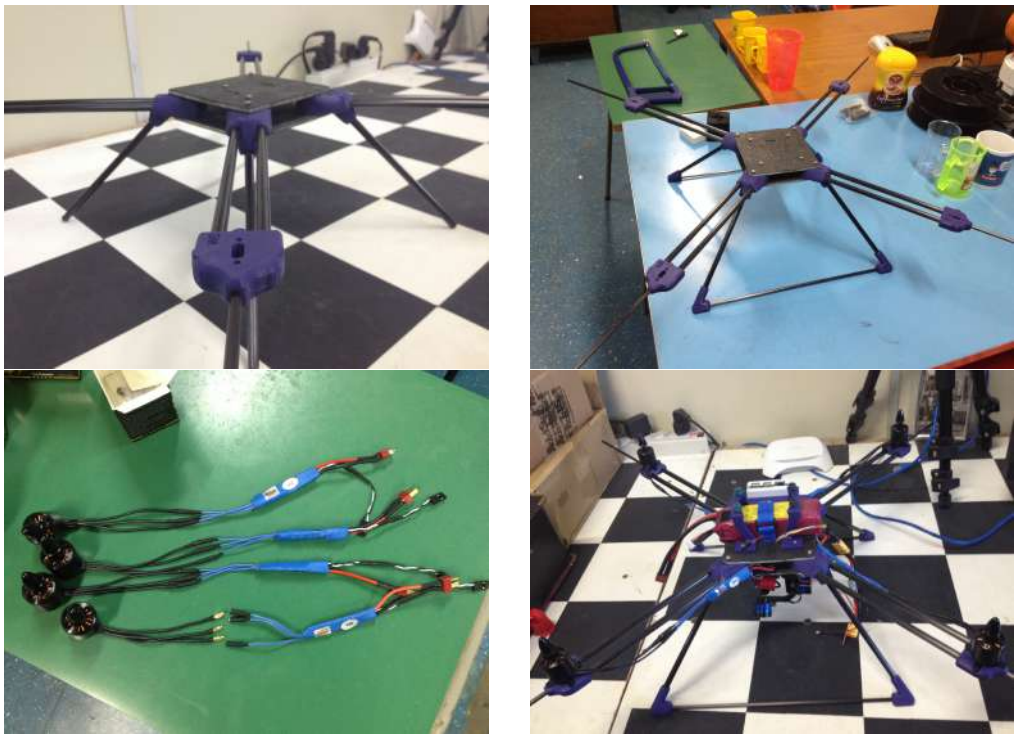
3.1 Estrutura Física do Veículo

O VANT montado e utilizado neste trabalho é do tipo quadrimotor, com estrutura feita em fibra de carbono e peças em PLA, modeladas e impressas no Grupo Viros, como observado na Figura 7.

Algumas características construtivas devem ser salientadas, como uso de um par de tubos de fibra de carbono paralelo em cada braço do veículo, atribuindo resistência à torção nesse membro. Isso faz com que toda forca exercida pela hélice seja de fato orientada verticalmente para baixo.

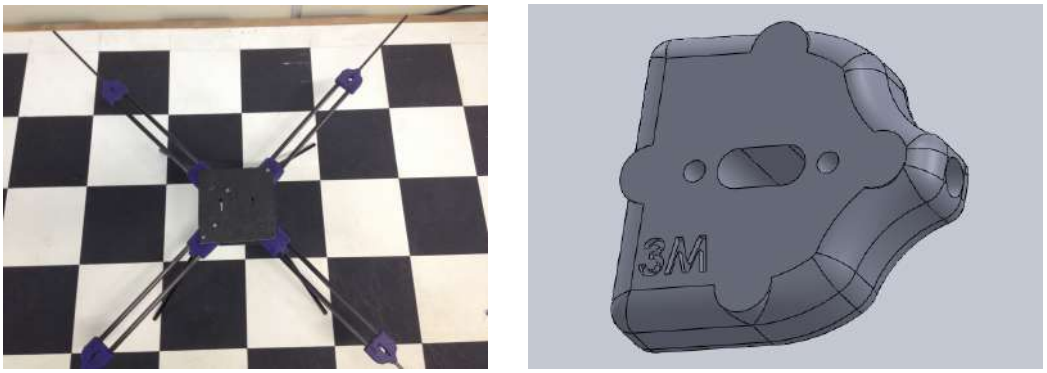
Outra característica a ser exaltada é a construção de um trem de pouso baseado em helicópteros. Dessa forma, o VANT ganha em resistência ao impacto, já que os esforços no contato com o solo são direcionadas e absorvidos pela parte de maior resistência do veículo, o centro, como observado na Figura 8.

Figura 6 – Etapas de montagem.



Fonte: Produção do próprio autor.

Figura 7 – Estrutura e modelagem.



Fonte: Produção do próprio autor.

Essa estrutura de trem de pouso viabiliza a utilização de um estabilizador de câmera, visto que libera o espaço adequado para fixação do equipamento entre o solo e o corpo central do veículo.

Com a possibilidade de modelar e imprimir a maioria das peças do projeto, foi viável otimizar a parte estrutural como um todo, já que regiões que apresentavam uma grande incidência de falhas ou quebras em projetos anteriores foram reforçadas. Partes que não sofrem com a incidência de esforços grandes foram reajustadas, por exemplo, reduzindo-se

Figura 8 – Distribuição de esforços no trem de pouso.

Fonte: Produção do próprio autor.

a densidade das mesmas.

Os avanços trazidos por esse novo modelo de VANT, visto na Figura 9a, portanto, ultrapassam os quesitos estruturais. Além de peças mais bem modeladas aerodinamicamente e resistentes, o novo modelo também teve melhora quanto à sua massa, totalizando 2,3kg, cerca de 200g a menos que o modelo anterior, visto na Figura 9b.

Figura 9 – Modelos de quadrimotores desenvolvidos.

(a) Modelo Atual

(b) Modelo Anterior

Fonte: Produção do próprio autor.

Na Figura 9a está apresentado o veículo atual desenvolvido. Ele é controlado por uma plataforma de hardware aberto chamada Pixhawk (PIXHAWK, 2017). Neste sistema, também estão embarcados módulos de magnetômetro, acelerômetro, barômetro e GPS (*Global Positioning System*) que auxiliam no controle adequado do VANT.

O quadrimotor também possui embarcado um microcomputador (Raspiberry Pi 2 Model

B+), responsável pela realimentação visual do VANT. A aquisição das imagens é realizada com uma RaspiCam v1.3 conectada à Raspiberry Pi. Como estabilizador da câmera, para uma melhor aquisição das imagens, foi utilizada uma Gimbal da Tarot. Esses equipamentos podem ser vistos na Figura 10.

Figura 10 – Equipamentos embarcados.

(a) Raspiberry Pi 2 Model B+

(b) RaspiCam v1.3

(c) Gimbal da Tarot

Fonte: Produção do próprio autor.

Os quatro motores *brushless* DC de 880 K_v são alimentados por uma bateria de 4 células com uma carga de 5200 mAh, que permite ao VANT uma autonomia média de 15 minutos. Essa bateria também alimenta a Pixhawk, a Raspiberry Pi e a Gimbal. A Figura 11 apresenta novamente o quadrimotor utilizado, construído para este projeto.

3.2 Sistema de Software Embarcado

Para realizar esse processo de realimentação visual e comunicar com a Pixhawk, o micro-computador funciona a base de Linux (Ubuntu ARM) e utiliza o *framework* ROS Indigo via MAVROS (protocolo MAVLink).

Figura 11 – O novo quadricóptero do laboratório.

Fonte: Produção do próprio autor.

Em síntese, o ROS é responsável por realizar a realimentação visual pretendida, operando como o controlador de alto nível do sistema. O Pixhawk é o controlador nativo do equipamento, trabalhando com o controle de baixo nível.

A comunicação entre esses dois níveis de controle é dado via protocolo MAVLink, valendo-se do pacote oferecido pelo ROS, o MAVROS.

3.2.1 ROS

O ROS é um pseudo sistema operacional para robôs, criado em 2007 pelo Laboratório de Inteligência Artificial de Stanford (SAIL) como suporte para o projeto Stanford AI Robot [1] . Nos anos seguintes, o desenvolvimento já contava com o apoio de mais de 20 instituições de pesquisa, de maneira que o software começa a tomar forma de um sistema de código aberto. Em 2013, o ROS se transforma em Open Source Robotics Foundation [2], oferecendo softwares de código aberto gratuitos, tanto para aplicação em pesquisa, quanto uso comercial.

Atualmente, o ROS é conhecido mundialmente como uma importante ferramenta na criação de softwares para robôs. Isso se deve ao fato de que a *framework* conta com características específicas para aplicações de robótica, como mecanismos de comunicação bem desenvolvidos (com integração a bibliotecas comumente utilizadas), gerenciamento de pacotes, simuladores, além de abstração de hardware, o que possibilita que a troca e reutilização de trabalhos ocorra com facilidade (QUIGLEY et al., 2009).

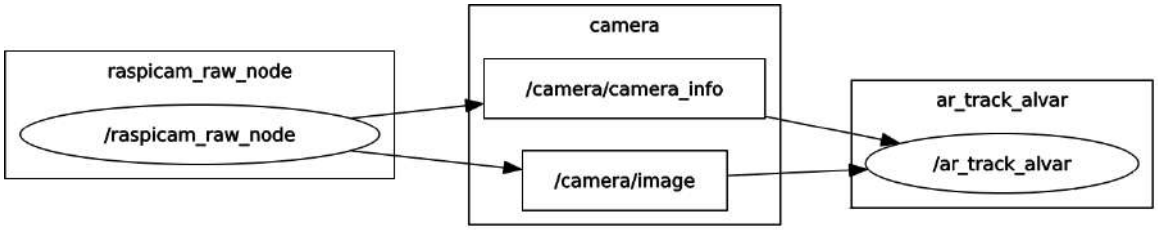
[1] http://stair.stanford.edu/
[2] https://www.osrfoundation.org/

O ROS funciona, resumidamente, em função de nós: pequenas tarefas executadas de forma independente entre si. Essas tarefas podem consumir informações e/ou publicá-las, sendo dito, então, que o nó pode ser configurado como *subscriber* e/ou *publisher*, respectivamente.

Esses nós se comunicam entre si por meio de tópicos. Em um tópico, um *publisher* publica seus dados, e *subscribers* se inscrevem para receber as informações contidas no tópico de interesse. Em um tópico podem haver múltiplos *subscribers* e *publishers*.

É importante ressaltar que os tópicos devem receber mensagens com uma estrutura padronizada do nó *publisher*, de maneira que os nós *subscribers* consigam consumir as informações desejadas de maneira adequada. Um exemplo dessa dinâmica é mostrado na Figura 12.

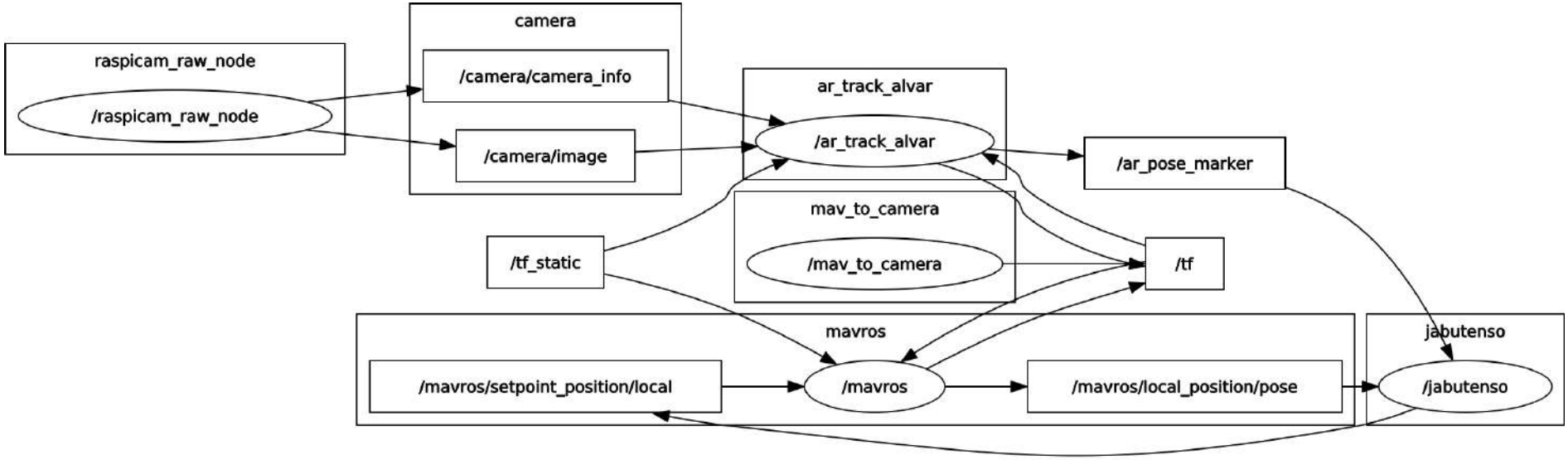
Figura 12 – Trecho do esquema de funcionamento do sistema - ROS.



Fonte: Produção do próprio autor.

Nesse exemplo, o nó que gerencia a Raspicam é responsável por gerar a imagem comprimida (imagem raw) e publicá-la no tópico **camera**. De forma análoga, o nó de detecção de padrão **ar_track_alvar** necessita de imagens para realizar suas tarefas, portanto, se inscreve no tópico citado. O esquema completo de funcionamento do sistema proposto é dado na Figura 13.

Figura 13 – Esquema completo de funcionamento do sistema - ROS.



Fonte: Produção do próprio autor.

3.2.2 MAVLink

O protocolo de comunicação utilizado no trabalho segue o padrão original usado pelo Pixhawk, o MAVLink (Micro Air Vehicle Link). Esse protocolo foi criado por Lorenz Meler em 2009 e é amplamente utilizado para a comunicação entre bases de controle/controle manual e VANT (MEIER et al., 2013)

A escolha por esse protocolo é baseada em dois fatos relevantes. O primeiro, é que o equipamento (Pixhawk) é configurado originalmente para funcionar com esse protocolo. O outro ponto importante a ser destacado, é que o MAVLink é projetado para veículos aéreos de pequeno porte e, portanto, tem como exigência de funcionamento um baixo custo operacional para troca de mensagens. Isso é possível devido às características simples na morfologia das mensagens, tornando-o um protocolo leve e rápido (ATOEV et al., 2017).

Outra vantagem do MAVLink para esse projeto é o fato de o ROS já possuir um pacote específico para esse tipo de comunicação, de maneira que toda a tomada de decisão e atitude tomada pelo controlador de alto nível seja levada de maneira compreensível e rápida para o controlador de baixo nível da arquitetura proposta. O uso do MAVLink como meio de comunicação entre os controladores possibilita ainda que o usuário possa retomar o controle do veículo (controle manual) quando achar necessário, visto que o controle manual continua conectado (e é fator imprescindível para a ativação de modos autônomos de voô) ao Pixhawk usando o mesmo protocolo.

4 AQUISIÇÃO E PROCESSAMENTO DE IMAGENS

Mesmo hoje, não é incomum nos depararmos com equipamentos robóticos realizando tarefas repetitivas, com pouca autonomia de decisão e baixa resiliência a estímulos externos. Esse cenário vem se alterando com o avanço tecnológico, mas ainda existem fatores, como a precisão do GPS (quando possuir) e a complexidade de se orientar em novos ambientes (problemas de SLAM), que impedem ou dificultam a flexibilização das aplicações de VANTs autônomos, por exemplo.

A utilização de câmeras, sendo sensores de visão, pode solucionar, ou pelo menos minimizar, grande parte desses problemas. O motivo está na riqueza de informações contidas em uma imagem, permitindo ao VANT se localizar mais facilmente no ambiente e adotar características de controle adequadas para cada situação.

Utilizar a visão como elemento de realimentação, adiciona ao sistema uma maior flexibilidade quanto à movimentação, por ser capaz de extrair a posição do robô em relação ao objeto de interesse e então encontrar a melhor maneira de chegar até ele. Como consequência, esse VANT pode valer-se de sistemas auxiliares (sensores de ultrassom, barômetro, GPS, etc.) de menor precisão, reduzindo a complexidade e custo da plataforma, uma vez que a precisão final de movimentação do veículo estará fortemente vinculada ao sistema de aquisição e processamento de imagens.

Entretanto, o ciclo de processamento e comando necessita ser rápido para que contribua de forma significativa e segura para a movimentação final do veículo. Neste trabalho, por exemplo, uma das missões do VANT é o seguimento de um alvo móvel. Uma taxa elevada de atualizações é crucial para o correto cumprimento da tarefa.

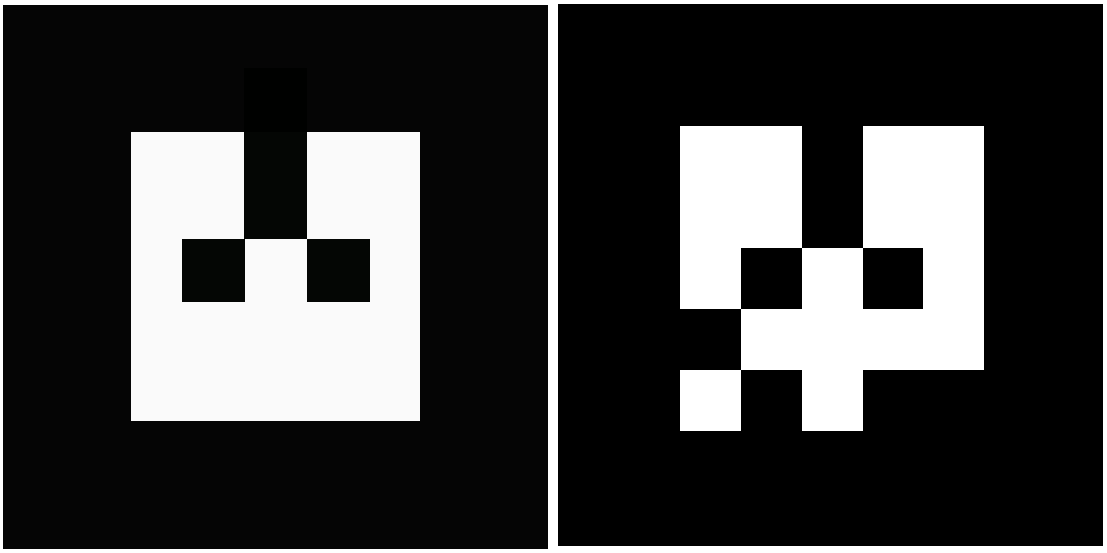
Em oposição a essa taxa de comando desejada está a complexidade natural de se lidar com tamanho fluxo de dados obtidos por câmeras. Analisar uma cena em busca de objetos ou características específicas pode ser custoso, principalmente objetivando taxas elevadas de atualização.

Uma solução possível e adequada para essa questão é a utilização de padrões pré determinados ou marcadores visuais na dinâmica da cena (OH et al., 2011). A Figura 14 apresenta os marcadores utilizados neste trabalho.

Missões de segmento de alvo móvel são especialmente favoráveis à aplicação desses marca-

Figura 14 – Padrões utilizados no trabalho.

(a) Padrão A (b) Padrão B



Fonte: Produção do próprio autor.

dores visuais visto que existem algoritmos de busca otimizados para a identificação de tais padrões em imagens, por exemplo, o *ar_track_alvar* (algoritmo utilizado neste trabalho) (AR_TAG, 2017) ou o ArUco (ARUCO, 2018).

Dessa forma, é possível identificar o alvo de maneira mais eficiente e menos custosa computacionalmente, assim como pretendido para esse tipo de tarefa. Consequentemente, isso viabiliza o processamento *onboard* desses dados, agregando ainda mais flexibilidade ao equipamento, visto que não precisaria, necessariamente, de links com bases terrestres.

Esse capítulo apresenta a etapa inicial do processo de realimentação visual, objetivando esclarecer os detalhes e peculiaridades inerentes. Para tornar a explicação mais concisa, o tema foi dividido em dois tópicos. O primeiro trata da aquisição de imagens, com especificações e mecanismos adotados; o segundo elucida as técnicas e ferramentas utilizadas para extração das informações pretendidas a partir das imagens capturadas.

4.1 Captura das imagens

Imagem é uma representação bidimensional de uma cena, seja por uma pintura ou fotografia. É dita imagem digital a imagem de uma cena capturada por lentes e conduzidas a sensores onde a luz é transformada em sinais elétricos. O sinal analógico é convertido a digital ponto a ponto em uma matriz que é chamada, ao final, de imagem digital.

O sistema de captura de imagens é parte crucial no trabalho. Imagens obtidas de maneira inadequada podem dificultar ou inviabilizar o estágio de processamento seguinte.

No sistema em questão, o sensor de visão responsável pela captura das imagens é a RaspiCam. Ela é uma câmera proprietária e específica para a Raspiberry Pi e foi configurada para obter imagens a uma taxa de 30 fps. A Figura 15 apresenta a RaspiCam conectada à Raspiberry Pi, ambas utilizadas neste trabalho.

Figura 15 – RaspiCam conectada à Raspiberry Pi.

Fonte: Produção do próprio autor.

Como pode-se notar na Figura 15, as duas são conectadas por um cabo flat serial. Esse cabo é curto e possui uma malha de aterramento para a mitigação das interferências eletromagnéticas dos sistemas de potência do veículo. A corrente de partida dos motores é um exemplo disso, podendo totalizar 80A.

Outro fator que influencia diretamente no desempenho da captura das imagens é a vibração e dinâmica de movimentos do equipamento. A rotação dos motores faz com que toda estrutura do veículo vibre e, portanto, uma câmera fixada diretamente a ele fica sujeita a todo tipo de interferência dessa natureza. Uma solução para essa questão é o emprego de um estabilizador de câmera (Gimbal), mostrada na Figura 16.

Ela é formada por dois motores que movimentam a câmera, tentando manter constantes duas das suas três componentes de orientação ($ROLL$ e $PITCH$). Além disso, ela absorve parte da vibração do equipamento por meio de seu sistema de fixação ao VANT, composto por borrachas anti-vibração.

Entretanto, a adição da Gimbal ao sistema acaba alterando a relação entre os referenciais

Figura 16 – Gimbal utilizada.

Fonte: Produção do próprio autor.

do VANT e da câmera, ou seja, muda a maneira como os dados obtidos por imagem e pelos sensores se associam. Esse assunto será discutido com mais detalhes no Capítulo 5.

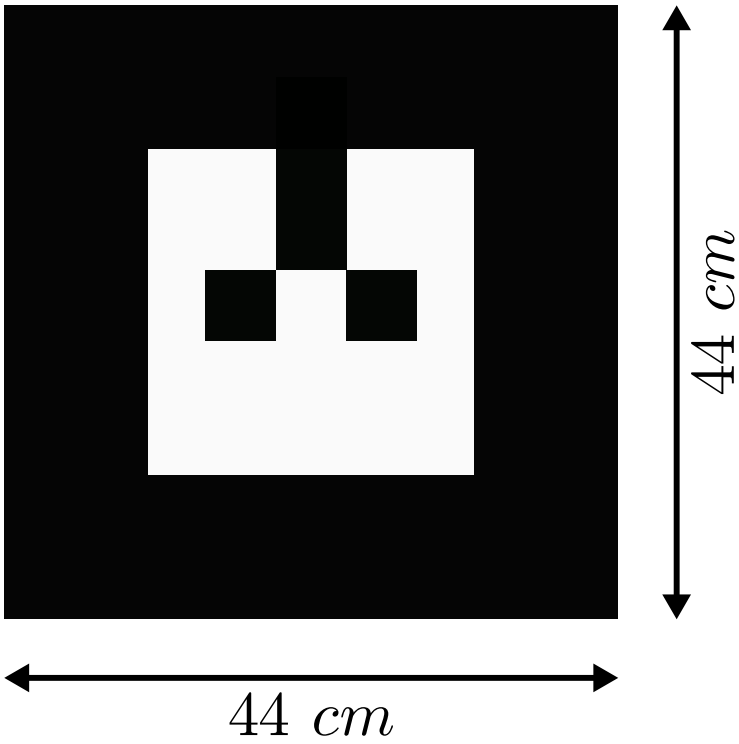
4.2 Processamento das imagens

Depois de capturadas pela RaspiCam, as imagens são processadas na Raspiberry Pi, extraindo-se as informações da cena em questão. Essas informações são, então, interpretadas por uma série de algoritmos que resultam em um comando de atitude para o VANT, que é enviado à Pixhawk.

O processamento inicial das imagem é dado pela busca de marcadores visuais, em que cada padrão específico deve ser interpretado de uma forma pelo algoritmo. Portanto, antes mesmo de se obter as informações espaciais e de disposição da cena, é analisado se há um padrão presente e, se sim, é verificado qual marcador é. A partir de então, o algoritmo pode seguir por ramos de análise diferentes a serem definidos pelas informações obtidas nessa etapa.

Os marcadores adotados nessa etapa são *AR Tags*, padrões amplamente difundidos na área de realidade aumentada (CHI; KANG; WANG, 2013). Eles são usados como referência para extração de informação no ambiente, principalmente, quando combinadas com informações prévias, como a dimensão do marcador. Neste trabalho, são usados marcadores com dimensão de $44 \times 44\ cm^2$. A Figura 17 apresenta um dos padrões utilizados e suas dimensões.

Figura 17 – Dimensões do padrão adotado.



Fonte: Produção do próprio autor.

A maioria dos programas voltados ao reconhecimento dos *AR Tags* têm por base o tamanho e a distorção do marcador na imagem, de maneira que, através de transformações projetivas, o programa retorne as características espaciais e geométricas do padrão presente na cena.

É válido ressaltar que a maioria dos padrões deste tipo são quadrados, facilitando a detecção de suas bordas a partir de uma única imagem/perspectiva da câmera. Sabendo disso, o algoritmo (*ar_track_alvar*) já é capaz de extrair a homografia da cena, levando em consideração o plano do marcador. Como cada marcador é diferente, há a possibilidade de detecção multimarcas.

Com o processo de homografia, é possível estimar as posições e orientações relativas na cena, entretanto, é perdida a informação de tamanho real dos objetos envolvidos. A homografia por si só gera uma informação incompleta para o propósito apresentado, ou seja, a menos de um fator de escala.

Esse problema pode ser solucionado acrescentando informações de entrada para o algoritmo, como a calibração da câmera usada (parâmetros intrínsecos e extrínsecos) e a dimensão do padrão. Dessa forma, a posição do marcador pode ser extraída em unidades de distância do mundo real, e não mais em termos de imagem (pixel).

Para essa tarefa, o ROS já conta com pacotes específicos, capazes de reunir todos os dados e variáveis de entrada, disponibilizando um resultado adequado e em tempo hábil para um processo de realimentação baseado em visão.

Neste trabalho, é utilizado o pacote *ar_track_alvar* (ROS, 2017), que é formado por várias bibliotecas de código aberto voltadas ao reconhecimento e seguimento de *AR Tags*. Ele conta com ferramentas de geração e customização de *AR Tags*, podendo anexar informações e identificadores às marcas, identificar e seguir os marcadores (tracking), etc. (ROS, 2017). No sistema de processamento, o *ar_track_alvar* recebe as imagens vindas da câmera e realiza sua busca por padrões (*AR Tags*).

Um dos chamativos que popularizou esse pacote, além de poder detectar a posição e orientação do padrão em relação à câmera, é a possibilidade de atribuir informações específicas a cada marcador, assim como atribuir hierarquias de comandos a partir da marca detectada (BENAVIDEZ et al., 2014).

Caso não seja identificado um marcador, o sistema não informa um novo dado de orientação e posição do padrão ao processo descrito no Capítulo 5. Assim, o microcomputador continuará informando à Pixhawk que o VANT deve ir para o último local onde o padrão foi detectado ou manter a execução da última tomada de atitude solicitada.

5 PROCESSAMENTO DOS DADOS

As informações contidas na cena finalmente são interpretadas, direcionando o algoritmo por entre seus ramos de possibilidades, culminando no envio de comandos de decisão ou atitude ao VANT. Dessa forma, encerra-se um ciclo de realimentação e dá-se início a outro.

No trabalho desenvolvido, grande parte dos esforços estão direcionados ao correto seguimento do padrão móvel por parte do VANT. Por essa importância, as peculiaridades e tomada de decisões durante o desenvolvimento da solução desta etapa serão descritas em mais detalhes na Seção 5.1.

Esse capítulo se encerra com a explicação do nível superior do processamento de dados, mais voltado à tomada de decisão a partir da sequência de fatos e marcadores apresentados durante o funcionamento do sistema.

É possível dividir a tarefa proposta por esse trabalho em frações menores, que caracterizam as etapas que o VANT vai percorrer até o cumprimento total do objetivo. Essas etapas são definidas aqui como missões, sendo que a tarefa é composta por três delas, as quais são detalhadas e explicadas na Seção 5.2.

5.1 Atualização de posicionamento do VANT a partir de dados do *ar_track_alvar*

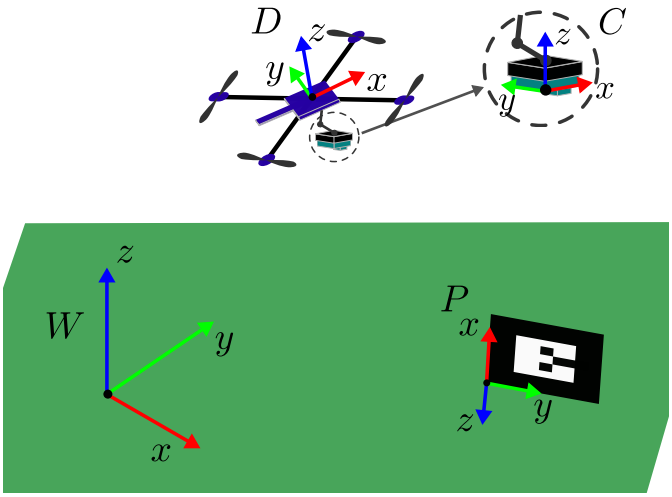
Depois de detectar o padrão, a Raspiberry Pi deve enviar à Pixhawk a posição e orientação que o VANT deve adquirir para se manter sempre sobre o padrão com mesma orientação que ele.

Assim que o padrão é detectado, a informação retornada pelo *ar_track_alvar* deve sofrer algumas transformações antes de ser enviada para a Pixhawk.

A Figura 18 apresenta a razão para isso: os dados do padrão P e do VANT D não se encontram num mesmo referencial.

Enquanto a orientação e posição D do VANT estão num referencial global W definido no instante da decolagem, os dados P do padrão estão representados no referencial C da câmera.

Figura 18 – Ilustração do cenário real da situação abordada com cada referencial da cena.



Fonte: Produção do próprio autor.

Para que se possa trabalhar com esses dois dados corretamente, ambos precisam ser comparados em um mesmo referencial (W por conveniência). Isso pode ser feito transformando os dados do padrão de C para W por meio de uma matriz de rotação R_{WC} e uma matriz de translação t_{WC}.

Considerando-se que o referencial da câmera C possui a mesma posição que o referencial do VANT D, uma vez que a distância entre C e D é muito menor que a distância de D a W, considera-se que t_{WC} é simplesmente a posição do VANT no referencial W.

Já a rotação poderia ser dada pela orientação de D em relação a W caso a câmera estivesse sempre alinhada com o VANT. Entretanto, a Gimbal foi utilizada para que não se perdesse o padrão de vista quando o VANT se movimentasse. Portanto, a orientação da câmera não pode ser considerada coincidente com a orientação do VANT.

A câmera é sempre mantida com vista para baixo. Dessa forma, C apresenta os eixos X e Y no mesmo plano XY do referencial W, compartilhando o mesmo eixo Z. A diferença entre esses dois referenciais, em termos de rotação, se resume ao ângulo de *yaw*. O ângulo de *yaw* θ_D do VANT define essa rotação, ou seja, R_{WC} é dada pela matriz de rotação

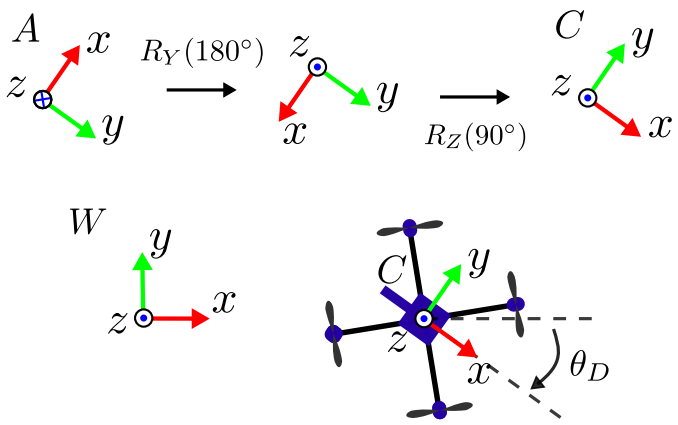
$$R_{WC} = \begin{bmatrix} \cos(\theta_D) & -\sin(\theta_D) & 0 \\ \sin(\theta_D) & \cos(\theta_D) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{5.1}$$

Assim, é possível descobrir a posição do padrão $\boldsymbol{X}_W = (x_W, y_W, z_W)^T$ dada em W a partir de C ($\boldsymbol{X}_C$) por meio da relação

$$\boldsymbol{X}_W = R_{WC}\boldsymbol{X}_C + t_{WC}. \tag{5.2}$$

Um último detalhe sobre a informação de posição do padrão é que o pacote *ar_track_alvar* considera um sistema de coordenadas A parecido com o de um avião na captura das imagens, que é diferente do de um quadrimotor. Para resolver isso e ter de fato os dados do padrão em C, duas rotações consecutivas devem ser realizadas, uma de 180° no eixo Y e outra de 90° no eixo Z. Essas transformações estão ilustradas na Figura 19.

Figura 19 – Transformações para mudança de referencial do sistema A para o sistema de coordenadas C da câmera de um quadrimotor.



Fonte: Produção do próprio autor.

Essas transformações juntas podem ser escritas como

$$\boldsymbol{X}_C = R_Z(90°)R_Y(180°)\boldsymbol{X}_A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \boldsymbol{X}_A, \tag{5.3}$$

onde, $\boldsymbol{X}_A$ é a posição do padrão no referencial do *ar_track_alvar*.

Dessa forma, é possível informar à Pixhawk uma posição $\boldsymbol{Q}_W$ para onde o VANT deve ir

a partir da posição atual do padrão, sendo

$$\boldsymbol{Q}_W = \boldsymbol{X}_W + \begin{bmatrix} 0 \\ 0 \\ z_o \end{bmatrix}, \tag{5.4}$$

onde, z_o é a altura que o VANT sempre deve estar do padrão enquanto realiza o seguimento.

Já o controle de orientação final compreende uma característica própria dos quadrimotores, em que o controle depende apenas da rotação no eixo Z (YAW). Portanto, das três componentes de orientação obtidas ($ROLL$, $PITCH$ e YAW) apenas o YAW é levado em consideração.

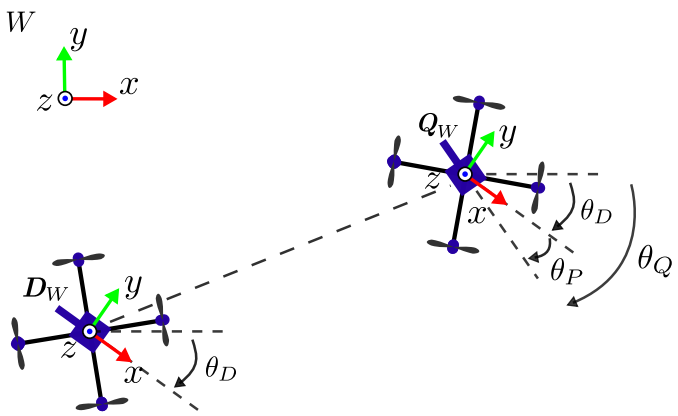
Apesar da Gimbal fazer com que a câmera não possa ser considerada parte fixa do VANT, sua utilização ainda mantém as informações de posição e YAW idênticas entre o VANT e a câmera.

Assim, tendo o *yaw* do VANT/câmera (θ_D em W) e o YAW do padrão (θ_P em C), pode-se calcular a orientação final do VANT θ_Q por meio da relação

$$\theta_Q = \theta_D + \theta_P, \tag{5.5}$$

assim como apresentado na Figura 20.

Figura 20 – Ilustração do controle de posição e orientação final do quadrimotor.



Fonte: Produção do próprio autor.

Um tratamento final no ângulo de *YAW* θ_Q deve ser feito devido à própria maneira com que o controlador do veículo lida com os ângulos recebidos, sendo que um giro completo compreende angulação de $-180°$ a $180°$. Como a angulação final θ_Q é uma composição entre variáveis, o valor resultante pode extrapolar estes limites gerando ações de controle desordenadas e perigosas à estabilidade do equipamento.

Isso pode ser solucionado ajustando os ângulos por meio de identidade trigonométrica de maneira a obedecerem à faixa aceitável, permitindo, inclusive, otimizar a decisão do melhor sentido de giro a ser adotado.

Ao fim de todo esse processo, a posição $\boldsymbol{Q}_W$ e orientação θ_Q são enviados à Pixhawk em forma de posição e orientação final para que o VANT possa realizar o seguimento.

5.2 Missões

No cumprimento de uma tarefa, o veículo pode sofrer com influência de estímulos externos, por isso, é importante que o equipamento seja capaz de se adequar às diferentes situações a ele apresentadas, assumindo um comportamento específico para cada caso.

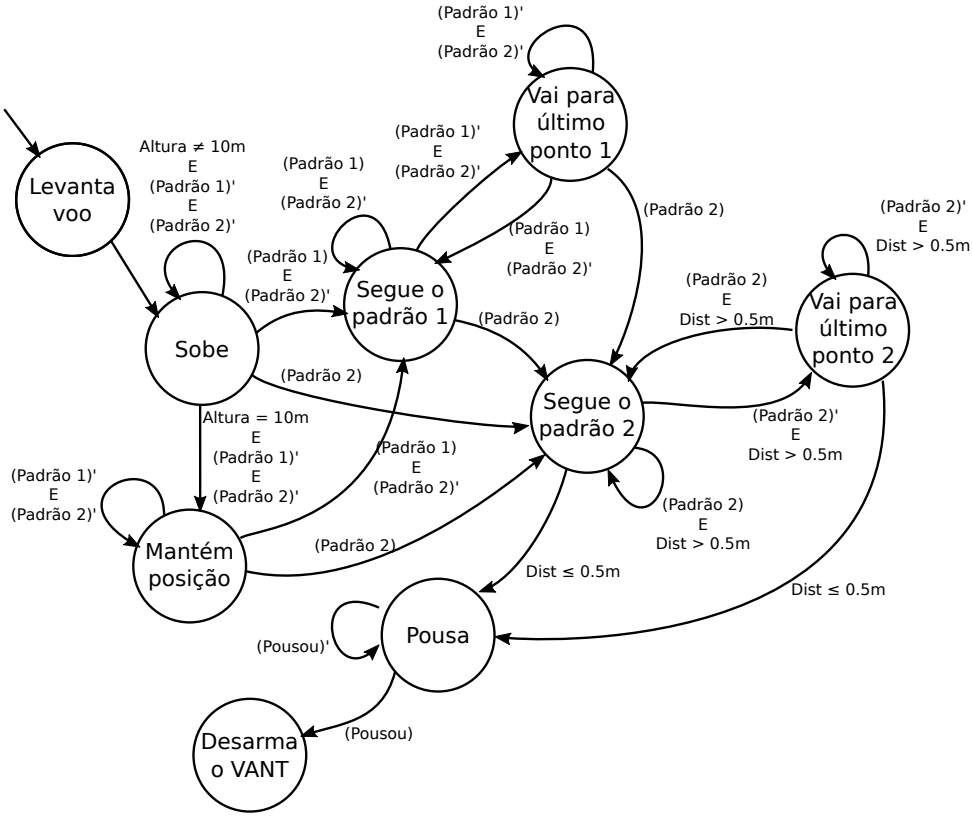
Até então, este capítulo manteve o foco em como o controle de posição e orientação final do VANT pode ser feito a partir da identificação e seguimento de marcadores visuais. Entretanto, no cumprimento de uma tarefa completa, de forma autônoma, existem mais etapas a se cumprir que não seja perseguir um alvo.

O conjunto de missões de uma tarefa (à base de realimentação visual) pode conter instantes em que sequer há a detecção do padrão. Isso ocorre na etapa de decolagem, por exemplo, em que o VANT levanta voo, com baixo campo de visão devido à sua baixa altitude. Isso implica em chances de não encontrar um marcador visual. Nesse momento, deve entrar em ação uma rotina de busca.

Nesse tipo de caso, é esperado que o sistema seja capaz de oferecer ao veículo métodos e alternativas para que ele possa encontrar os padrões desejados.

A tarefa objetivada por esse trabalho pode ser dividida em três missões, que vão operar separadamente conforme forem solicitadas e serão explicadas a seguir. Vale ressaltar que a ativação de cada missão dependerá do transcorrer do algoritmo, variando com a apresentação dos estímulos externos, como a apresentação ou não do marcador na cena e a sequência cronológica desses acontecimentos. A figura 21 traz a máquina de estados para a tarefa.

Figura 21 – Máquina de estados da tarefa autônoma.



Fonte: Produção do próprio autor.

5.2.1 Missão de busca inicial

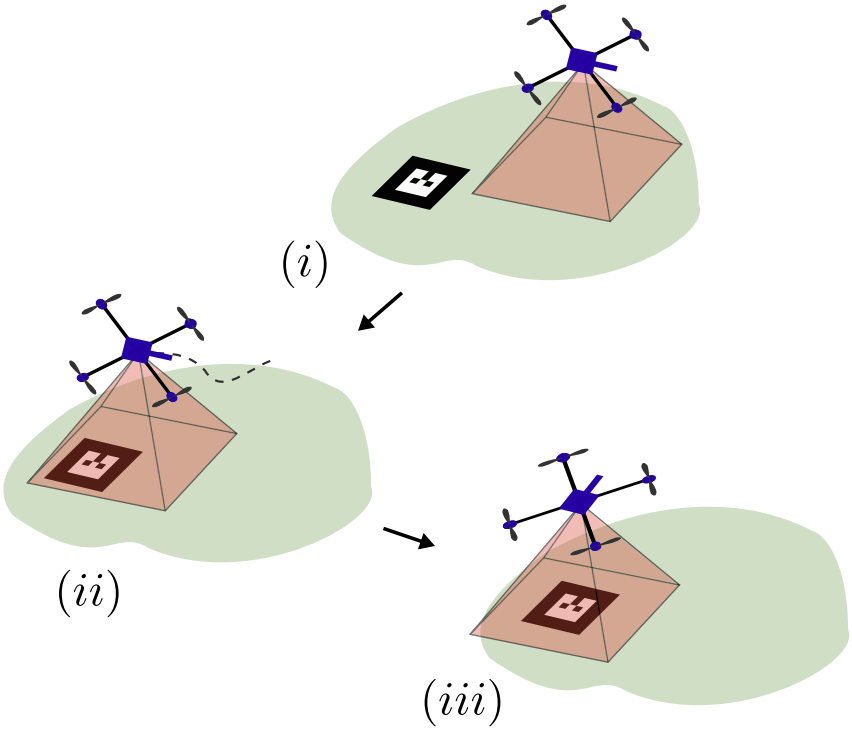
Após os procedimentos de inicialização do sistema, inicia-se o algoritmo de controle baseado em visão. Na rotina principal, de seguimento, o VANT identifica o padrão na imagem e começa a perseguir o marcador. Entretanto, nos momentos iniciais, em que o veículo está em solo, ainda não há detecção de qualquer marcador. Assim, o equipamento tenderia a permanecer parado.

É nesse contexto que a missão de busca inicial atua, tendo como premissa o fato de que nenhuma marca foi detectada ainda durante a tarefa. O VANT recebe rotas pré determinadas (como uma subida reta ou espiral, por exemplo, a ser determinado previamente pelo usuário) valendo-se do fato de que essas trajetórias aumentam a área de visão da câmera em relação ao solo e, assim, permite ao VANT encontrar seu marcador desejado. É importante ter critério na escolha das rotas pré determinadas da missão de busca, para minimizar a chance de colisão. Afinal, no sistema proposto, não há sensores de proteção de entorno do VANT.

A intenção é que, durante a rotina de busca inicial, o marcador possa entrar no campo de visão do VANT e, dessa maneira, seja ativada a próxima missão, dada pelo seguimento do

marcador. Iniciada essa nova missão, a rotina de busca não será mais executada novamente. Esse processo pode ser conferido na Figura 22.

Figura 22 – (i) No primeiro momento, o padrão ainda não está no campo de visão da câmera. (ii) Ao ser detectado pela primeira vez, (iii) o VANT tenta se alinhar ao padrão e passa a segui-lo.



Fonte: Produção do próprio autor.

Caso o VANT cumpra a rota pré determinada e não detecte o marcador, é possível determinar uma ação, configurada pelo usuário, como: pouso mantendo as coordenadas horizontais; retorno e pouso em seu local de decolagem; e manter sua posição final, assumindo que o marcador pode estar em movimento e vá entrar em algum momento em seu campo de visão.

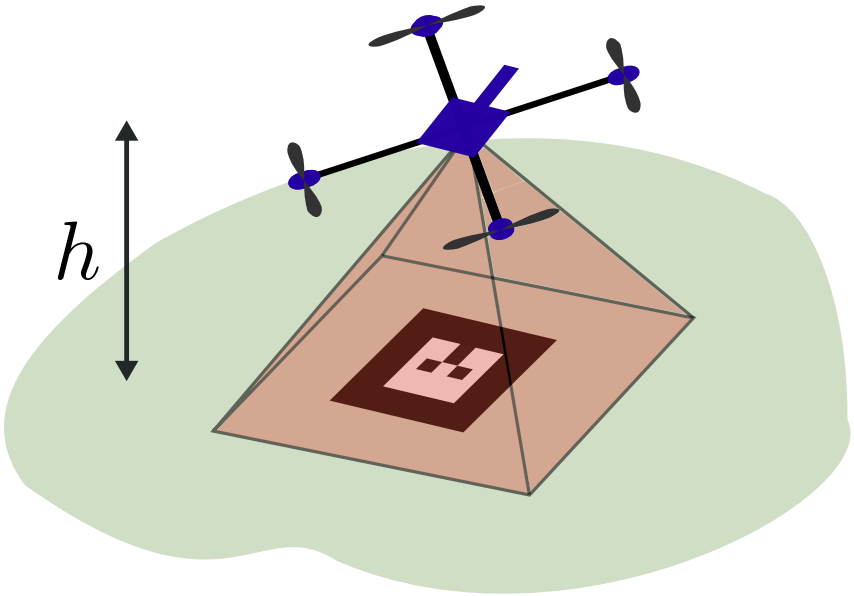
5.2.2 Missão de seguimento de padrão

Ao se detectar pela primeira vez o padrão adequado, apresentado na Figura 24a, o veículo passa a segui-lo. O controlador tenta manter o VANT acima do marcador, permanecendo a uma distância vertical determinada pelo usuário, como mostrado na Figura 23.

Essa missão é executada até que o VANT receba o estímulo de parada, conforme explicado na Seção 5.2.3, dado pela identificação do marcador visual apresentado na Figura 24b.

A partir de então, o algoritmo realiza uma análise simples, para garantir que a outra marca encontrada seja realmente a responsável pela ativação da última missão. Essa verificação

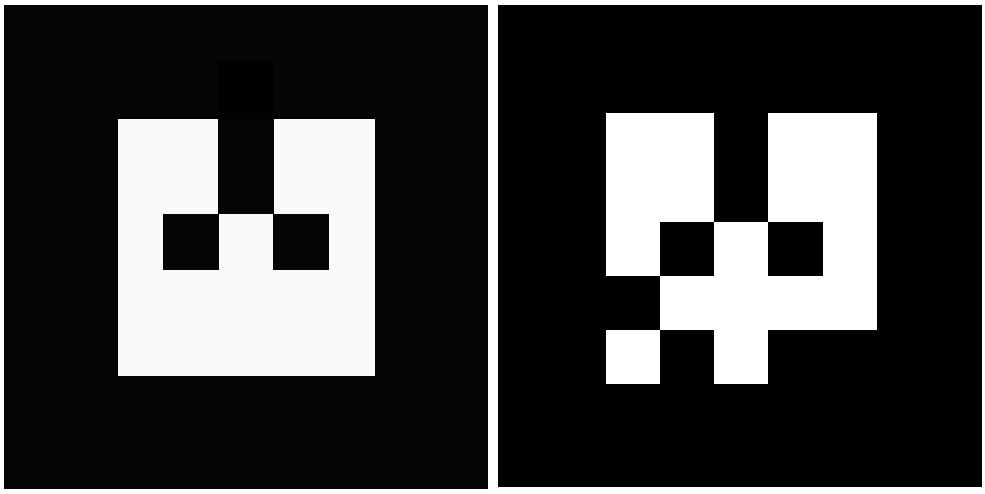
Figura 23 – Distância definida pelo usuário.



Fonte: Produção do próprio autor.

Figura 24 – Padões utilizados no trabalho.

(a) Padrão A (b) Padrão B



Fonte: Produção do próprio autor.

ocorre nas três primeiras capturas do padrão designado ao pouso. Então, caso a posição das três marcas sejam convergentes (no referencial global) o sistema confere a real presença do padrão em questão. Dessa maneira, falhas por meio de eventuais *outliers* são evitadas, elevando a confiabilidade do trabalho.

Com a confirmação da marca B, a missão de pouso é ativada e, as outras missões, desabilitadas.

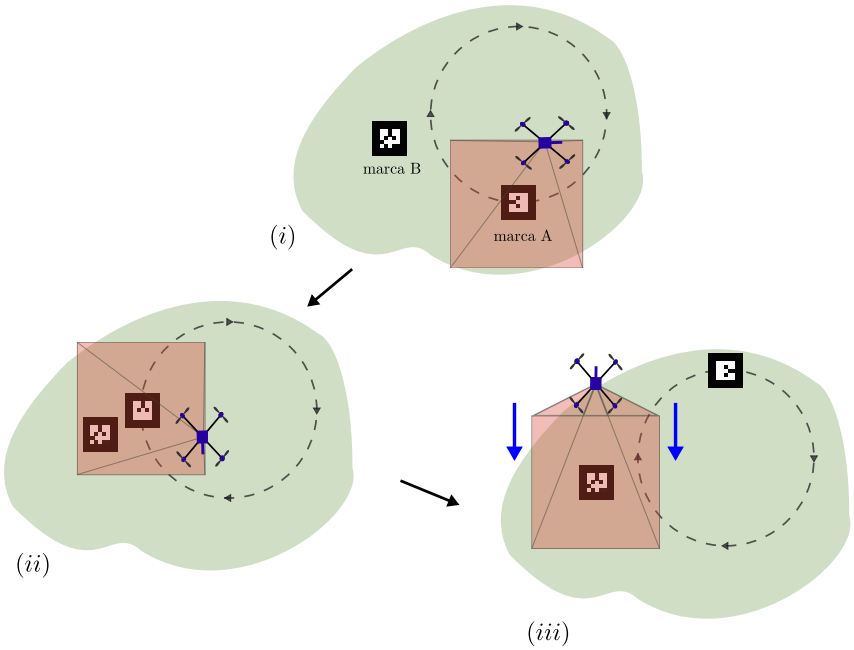
5.2.3 Missão de pouso

Após a detecção e confirmação da marca B, o sistema proposto passa a ignorar a eventual existência de qualquer outro marcador na imagem, buscando apenas por essa marca. Nessa missão, o VANT se desloca para a posição da marca B e pousa, finalizando a tarefa. Para que todo o processo possa ser realizado novamente, o sistema deve ser reiniciado. Vale ressaltar que durante todas as missões, a qualquer momento, o usuário pode retomar o comando do veículo ativando o modo de voo MANUAL no rádio controle.

Esse processo é semelhante à missão de seguimento de padrão (Seção 5.2.2): o VANT identifica e passa seguir a marca B, mantendo a distância vertical escolhida pelo usuário. Entretanto, essa missão também contabiliza a distância horizontal (plano XY) do VANT ao marcador, de maneira que quando ela se torna menor que um raio de 0,5 m, o VANT é autorizado a pousar.

Quando recebe o comando de pouso todo o processo ocorre de maneira automática e já não considera mais a existência de marcadores na cena, sendo a trajetória de descida retilínea a partir do ponto em que foi autorizado. Todo o processo da missão de pouso pode ser conferido na Figura 25.

Figura 25 – Inicialmente, (i) o VANT está seguindo o padrão A. (ii) Após detectar o padrão B, (iii) o VANT deixa de se movimentar em busca do marcador A e da início ao pouso.



Fonte: Produção do próprio autor.

6 EXPERIMENTOS E RESULTADOS

Após a montagem da plataforma e a elaboração de toda a lógica de comando e tomada de decisão, o sistema foi exposto a uma série de pequenos testes iniciais. esses testes visavam compreender o comportamento/tendência do sistema quando exposto a algumas situações controladas, como a perda momentânea do marcador ou a apresentação do marcador designado ao pouso.

Esses testes checam cada etapa de funcionamento descrita nos capítulos anteriores deste trabalho. Todos eles puderam ser realizados em solo e são descritos a seguir.

6.1 Ativação do sistema

Para avaliar a configuração inicial do sistema e confirmar que a comunicação entre os blocos de controle da arquitetura operam adequadamente, avaliou-se a resposta inicial de posicionamento pretendido após a ativação. Nesse caso, era esperado pela lógica de comando que a posição desejada (não executada sem o aval do operador) fosse fixa, desde que não haja marcadores na cena.

Como pode ser observado na Figura 26, o sistema está ativo e mantém o envio constante de uma posição fixa ao controlador final (Pixhawk). O recebimento constante de atualizações é requisito básico para a ativação de modos autônomos de voo no Pixhawk.

6.2 Avaliação do sistema de captura de imagens e detecção dos marcadores

É pretendido que o sistema seja sensível à presença de marcadores na cena, portanto, nesse teste foi avaliado se havia mudança de comportamento perante a apresentação de um marcador ao sistema.

Na Figura 27 é possível observar que o sistema detectou o marcador e informou seus dados, como posição e orientação. Além disso, pode-se notar que o sistema detecta a identificação de cada marcador (valor de **id** na Figura 27), funcionalidade crucial para o escalonamento das missões e tomada de decisão. Neste trabalho, o padrão com **id** igual a 0 foi chamado de marcador A e o de **id** igual a 1, de marcador B. Vale ressaltar que as unidades de medida seguem o padrão internacional.

Figura 26 – Publicações de posição iniciais.

```
---
header:
  seq: 665
  stamp:
    secs: 1528818666
    nsecs: 548073133
  frame_id: ''
pose:
  position:
    x: 0.0
    y: 0.0
    z: 10.0
  orientation:
    x: 0.0
    y: 0.0
    z: 0.0
    w: 0.0
---
header:
  seq: 666
  stamp:
    secs: 1528818666
    nsecs: 548073133
  frame_id: ''
pose:
  position:
    x: 0.0
    y: 0.0
    z: 10.0
  orientation:
    x: 0.0
    y: 0.0
    z: 0.0
    w: 0.0
---
```

```
---
header:
  seq: 667
  stamp:
    secs: 1528818666
    nsecs: 548073133
  frame_id: ''
pose:
  position:
    x: 0.0
    y: 0.0
    z: 10.0
  orientation:
    x: 0.0
    y: 0.0
    z: 0.0
    w: 0.0
---
  header:
  seq: 668
  stamp:
    secs: 1528818666
    nsecs: 548073133
  frame_id: ''
pose:
  position:
    x: 0.0
    y: 0.0
    z: 10.0
  orientation:
    x: 0.0
    y: 0.0
    z: 0.0
    w: 0.0
---
```

Fonte: Produção do próprio autor.

6.3 Avaliação da intenção de posicionamento

Para analisar a intenção de posicionamento, foi apresentado ao sistema de captura de imagens o marcador relativo ao seguimento (marca A). A análise dessa etapa provém da resposta final do controlador, ou seja, da intenção de movimentação.

Para tanto, o marcador foi inserido no campo de visão da câmera e então deslocado na cena, de maneira que ainda pudesse ser capturado. A resposta do algoritmo acompanhou essa variação, alterando também a intenção de posicionamento, como visto na Figura 28.

6.4 Análise da perda momentânea do padrão e eventuais outliers

Esse teste consiste em analisar o comportamento do VANT ao se retirar o padrão de cena ou obstruir a visão da câmera. A Figura 29 apresenta o resultado desse teste.

A Figura D exemplifica a intenção de movimentação no exato momento em que o padrão pára de ser captado durante a missão de seguimento. Como pode ser visto, o sistema

Figura 27 – Detecção do marcador visual. O campo **id** indica qual o padrão está sendo detectado, seguido de **pose**, que contém informações espaciais do marcador.

```
markers:
  -
    header:
      seq: 0
      stamp:
        secs: 1528814927
        nsecs:  99122737
      frame_id: /camera
    id: 1
    confidence: 0
    pose:
      header:
        seq: 0
        stamp:
          secs: 0
          nsecs:         0
        frame_id: ''
      pose:
        position:
          x: 0.123269152623
          y: 0.0750668912839
          z: 1.88340728125
        orientation:
          x: 0.99666524042
          y: -0.000518507645484
          z: -0.0356743948353
          w: 0.0733857427658
---
```

Fonte: Produção do próprio autor.

Figura 28 – Variação de intenção de posicionamento conforme a posição do padrão na imagem.

(a) VANT alinhado com o padrão

```
markers:
  -
    header:
      seq: 0
      stamp:
        secs: 1528815400
        nsecs:  65944327
      frame_id: /camera
    id: 0
    confidence: 0
    pose:
      header:
        seq: 0
        stamp:
          secs: 0
          nsecs:         0
        frame_id: ''
      pose:
        position:
          x: 0.00420078696961
          y: 0.0060863127698
          z: 1.24945469169
        orientation:
          x: 0.0185197327197
          y: 0.998163800019
          z: 0.0487853084821
          w: 0.0307577877605
---
```

(b) VANT desalinhado com o padrão

```
markers:
  -
    header:
      seq: 0
      stamp:
        secs: 1528815562
        nsecs: 655168849
      frame_id: /camera
    id: 0
    confidence: 0
    pose:
      header:
        seq: 0
        stamp:
          secs: 0
          nsecs:         0
        frame_id: ''
      pose:
        position:
          x: 0.299103088252
          y: 0.226072340066
          z: 1.24684995767
        orientation:
          x: 0.0340408638523
          y: 0.997089213281
          z: 0.0631174671457
          w: 0.0258941245985
---
```

Fonte: Produção do próprio autor.

Figura 29 – Perda do padrão (a última posição continua sendo enviada).

```
header:
  seq: 5256
  stamp:
    secs: 1528816125
    nsecs: 972570873
  frame_id: ''
pose:
  position:
    x: -1.00725388554
    y: 3.9152027109
    z: 11.1853653771
  orientation:
    x: -0.0
    y: 0.0
    z: 0.886366471445
    w: -0.462984317551
---
header:
  seq: 5257
  stamp:
    secs: 1528816126
    nsecs:  22561967
  frame_id: ''
pose:
  position:
    x: -1.29758294009
    y: 3.87317419432
    z: 10.6252029577
  orientation:
    x: -0.0
    y: 0.0
    z: 0.0994361405099
    w: -0.995043945743
---
```

```
---
header:
  seq: 5258
  stamp:
    secs: 1528816126
    nsecs:  72572696
  frame_id: ''
pose:
  position:
    x: -1.29758294009
    y: 3.87317419432
    z: 10.6252029577
  orientation:
    x: -0.0
    y: 0.0
    z: 0.0994361405099
    w: -0.995043945743
---
header:
  seq: 5259
  stamp:
    secs: 1528816126
    nsecs:  72572696
  frame_id: ''
pose:
  position:
    x: -1.29758294009
    y: 3.87317419432
    z: 10.6252029577
  orientation:
    x: -0.0
    y: 0.0
    z: 0.0994361405099
    w: -0.995043945743
---
```

reage adequadamente conforme planejado no algoritmo, mantendo sua última posição pretendida.

Nessas ocasiões, é possível que alguns *outliers* apareçam. Os *outliers* são raros e, devido à taxa de atualização, acabam não influenciando perceptivelmente no resultado esperado. Situações em que necessitam de maior grau de confiabilidade recebem uma rotina de confirmação, como já explicado na Seção 5.2.3.

6.5 Tomada de decisão conforme o padrão

Uma das principais funcionalidades do trabalho é a possibilidade de tomar decisões de acordo com os padrões detectados e com a ordem que aparecem na cena.

Para avaliar seu correto funcionamento ainda em solo, o marcador referente ao pouso automático (missão de pouso) foi inserido no campo de visão da câmera. A Figura 30 representa uma estrutura de diagnóstico instantânea geral do VANT, incluindo seus estados de voo atual. Nela, é possível ver o exato momento em que o marcador de pouso entrou

na cena, quando o campo de $STATE$ passa de $OFFBOARD$[1] para $AUTO_LAND$[2].

Figura 30 – Diagnóstico instantâneo geral do VANT.

```
---
header:
  seq: 860
  stamp:
    secs: 1528816748
    nsecs: 588868761
  frame_id: ''
connected: True
armed: False
guided: True
mode: OFFBOARD
---
header:
  seq: 861
  stamp:
    secs: 1528816749
    nsecs: 600033396
  frame_id: ''
connected: True
armed: False
guided: True
mode: OFFBOARD
```

```
---
header:
  seq: 862
  stamp:
    secs: 1528816750
    nsecs: 582647718
  frame_id: ''
connected: True
armed: False
guided: True
mode: AUTO.LAND
---
header:
  seq: 863
  stamp:
    secs: 1528816751
    nsecs: 593270635
  frame_id: ''
connected: True
armed: False
guided: True
mode: AUTO.LAND
```

Fonte: Produção do próprio autor.

Ao verificar que o sistema desenvolvido foi capaz de reagir bem aos testes iniciais, experimentos em voo puderam ser realizados. Para tanto, foi planejada uma série de testes que visaram explorar as principais características e contribuições do trabalho.

Devido à sua grande importância, a missão de seguimento de alvo móvel foi exposta a uma série de testes exclusivos de comportamento e desempenho. Nesses testes ainda, foram elucidados as melhores práticas de configuração de acordo com as especificações da tarefa, como a melhor altura relativa a ser mantida entre o VANT e o marcador (configurada pelo usuário).

Durante essas etapas de testes, dados do voo e de controle do VANT foram coletados, para que pudessem ser expostos e analisados adequadamente. Dessa forma foi possível avaliar as respostas do sistema mais a fundo, detalhando o real desempenho do trabalho desenvolvido.

A outra batelada de experimentos visou avaliar o desempenho do sistema completo, em que o VANT foi designado a cumprir uma tarefa inteira, incluindo decolagem, seguimento de marcador e pouso autônomos. Essa etapa teve por finalidade a compreensão do campo de abrangência do trabalho, com a exemplificação de uma aplicação, em que as três missões

[1] Modo em que o Pixhawk aceita comandos externos, como os provenientes do sistema de visão.
[2] Modo de voo designado ao pouso automático.

propostas pudessem ser solicitadas e executadas, objetivando o cumprimento da tarefa pretendida.

6.6 Seguimento de marcador móvel com realimentação visual

Para avaliar e validar o sistema de seguimento de padrão móvel, foram realizados testes em ambiente externos, onde o equipamento esteve sob influência de variação de iluminação e vento. Para o experimento em questão [3], foi posto um marcador visual (marca A) por cima de um robô terrestre (MobileRobots T M Pioneer 3-AT) e esse foi programado para realizar três rotas: (i) uma rota circular, com 3,80 metros de raio e velocidade linear de 600 mm/s; (ii) uma rota em ZigZag, com velocidade angular de 30 graus/s e linear de 700 mm/s; (iii) e uma rota em forma de 8, com velocidade linear máxima de 600 mm/s. As velocidades angulares e lineares instantâneas são determinadas de acordo com as características e limitações do Pioneer.

O desempenho do sistema de seguimento de padrões móveis é dependente de variáveis, como o tempo dispendido para processamento das imagens, a altura relativa do VANT ao marcador, a velocidade do marcador na imagem e o campo de visão da câmera.

Para ser capaz de seguir alvos ágeis é crucial que o ciclo de processamento das imagens seja o mais breve possível. Além disso, o campo de visão da câmera/altura de voo também influencia na velocidade máxima que o alvo pode ter para que o sistema de seguimento funcione adequadamente e sem perdê-lo.

Nesse contexto, dois fatores cruciais podem ser destacados, o tamanho do marcador e a resolução das imagens adquiridas. Por exemplo, um padrão pequeno possui detalhes menores, o que pode complicar sua detecção pela câmera do VANT quando ele voa em altitudes maiores. Em contrapartida, marcadores grandes demais podem ser inconvenientes em determinadas situações, como a passagem por corredores e locais estreitos. Para esse trabalho foram escolhidos marcadores com dimensões idênticas à superfície do robô terrestre, $44{,}0 \times 44{,}0cm^2$.

Para encontrar a resolução de imagem mais adequada, o veículo foi posto sobre o padrão A por cerca de um minuto a diferentes alturas, enquanto foram avaliados dois valores de resolução. A Tabela 1 contém a taxa de detecção do padrão na imagem (assumindo que o padrão sempre está contido na imagem) e o tempo médio de processamento das imagens para ambos os casos. Esse período também é compreendido como o tempo de atualização

[3] Esse teste foi gravado e pode ser conferido em: <https://www.youtube.com/watch?v=hB3IHAB07I0&t=0s>.

do sistema.

Tabela 1 – Taxa de detecção do padrão na imagem e tempo médio de processamento para resoluções de $480p$ e $720p$.

$480p$ (640×480)					
	10m	15m	16m	20m	24m
Detecção (%)	100	91,4	52,5	0,0	0,0
Tempo de Processamento (ms)			104,4		

$720p$ (1024×720)					
	10m	15m	16m	20m	24m
Detecção (%)	100	100	100	100	82,7
Tempo de Processamento (ms)			382,2		

Baseado nos resultados expostos na Tabela 1, foi adotada a resolução de 480p para os experimentos desse trabalho, mantendo uma altura relativa de 10 m. Esse valor possibilitou um menor tempo de processamento, mantendo, a 10m, a mesma taxa de detecção que a referente à configuração de maior resolução.

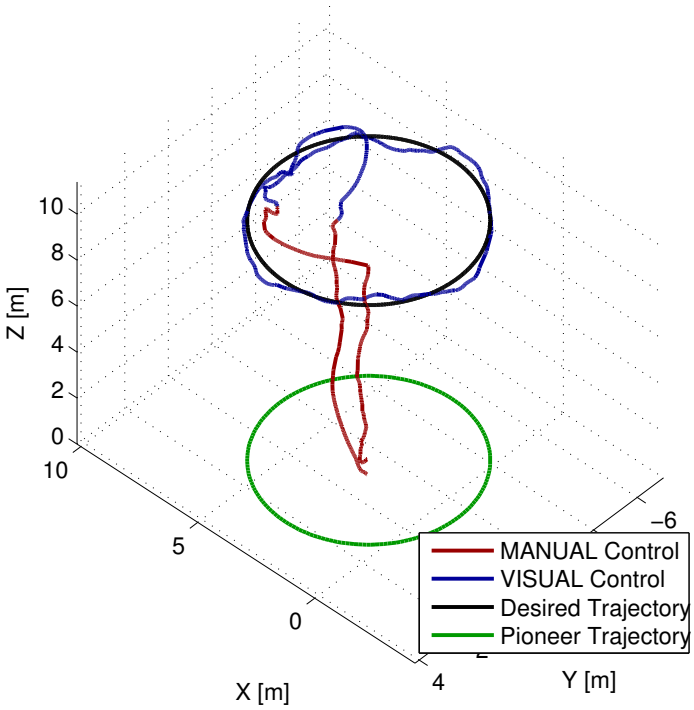
Com as características do marcador e sistema de detecção já definidas, os testes puderam ser iniciados. Primeiramente, o veículo é ligado e salva sua posição mantendo-a como referência para o restante do voo, em seguida, em modo manual, decola e permanece em voo a espera de um marcador. Ao se detectar um padrão, o sistema de seguimento por realimentação visual é ativado e passa a ter controle sobre a movimentação do VANT. O objetivo é que a distância horizontal para o marcador se aproxime de zero, enquanto a vertical deve ser mantida em 10m (z_o), que é definida pelo usuário.

As Figuras 31, 32, 33, 34 representam a odometria do robô terrestre, a posição do VANT e a trajetória esperada para o experimento de rotas diversas. A informação de posição é colorida de acordo com o modo de voo ativo no momento: Modo Manual, em vermelho; Modo Visual (controle baseado em realimentação visual), em azul; sendo possível observar, então, que o veículo foi capaz de seguir o alvo em todas as situações, mesmo sob influência de vento e variação de iluminação.

É possível notar na Figura 31, também, a presença de imperfeições na trajetória realizada em relação à objetivada. A Tabela 2 quantificam erros verticais e horizontais.

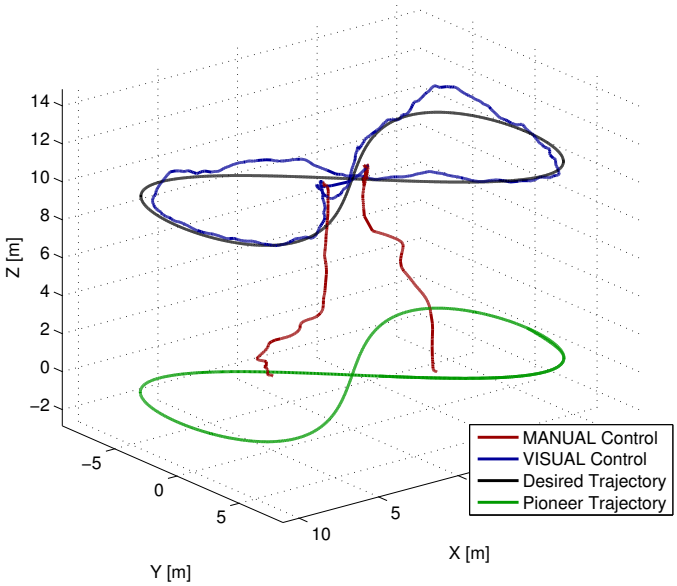
Porém, é importante salientar que a movimentação do VANT tem por base as informações de várias fontes, incluindo sensores inerciais, ultrassom e principalmente do seu GPS, que por sua vez insere um erro de medida ao sistema. O erro máximo inerente ao último sensor

Figura 31 – Odometria do robô terrestre, posição do VANT e trajetória esperada para a rota circular.



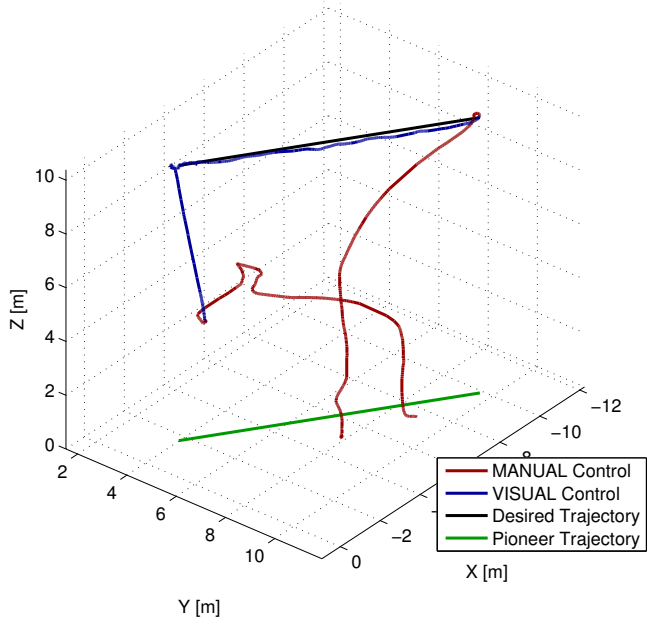
Fonte: Produção do próprio autor.

Figura 32 – Odometria do robô terrestre, posição do VANT e trajetória esperada para a rota em oito.



Fonte: Produção do próprio autor.

Figura 33 – Odometria do robô terrestre, posição do VANT e trajetória esperada para a rota em reta.



Fonte: Produção do próprio autor.

Tabela 2 – Erro de trajetória a partir dos dados do GPS para o experimento de rota circular.

Error	Erro Médio(cm)	Desvio Padrão(cm)	Erro Máximo(cm)
Horizontal	1,03	14,48	34,49
Vertical	42,44	38,37	117,11

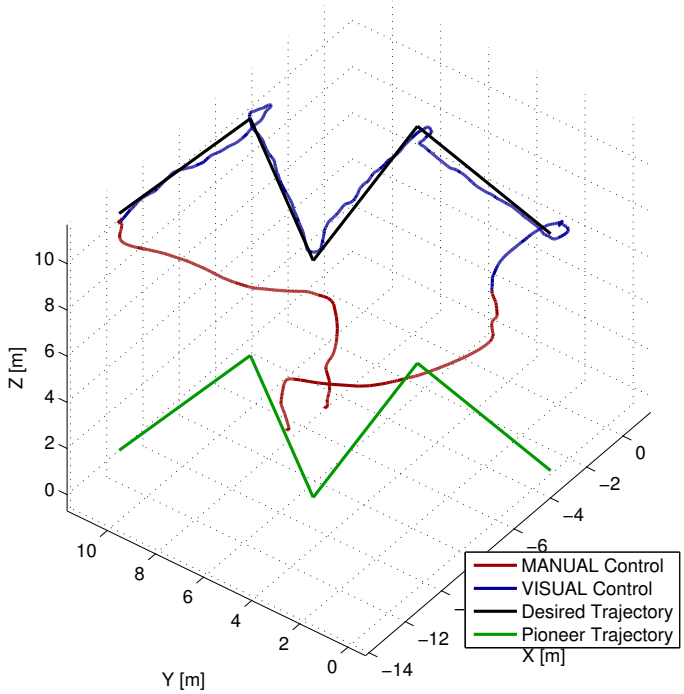
em questão pode chegar a $2m$ horizontalmente (NEO-6, 2011). Esse fato influencia na análise dos resultados coletados, não correspondendo ao real desempenho do sistema.

Para confirmar essa hipótese, o experimento foi gravado por uma câmera no solo, centrada na trajetória do robô terrestre e voltada para cima, de maneira a capturar toda a movimentação do VANT. A Figura 35 traz a rota esperada, em verde; a rota do VANT, dada pela informação de GPS, em vermelho; e a rota calculada pela informação visual extraída das imagens da câmera em solo, em azul.

Como pode ser observado na Figura 35, a real trajetória do VANT apresenta menos variações e ruídos quando comparado com a obtida via informações do GPS.

Uma questão relevante a ser discutida é o fato de que apesar da imprecisão do GPS, o veículo é capaz de seguir a marca com sucesso. Isso pode ser explicado pelo seguinte: quando o GPS traz um erro de medida na posição do VANT em W (sistema de coordenada

Figura 34 – Odometria do robô terrestre, posição do VANT e trajetória esperada para a rota em zigzag.



Fonte: Produção do próprio autor.

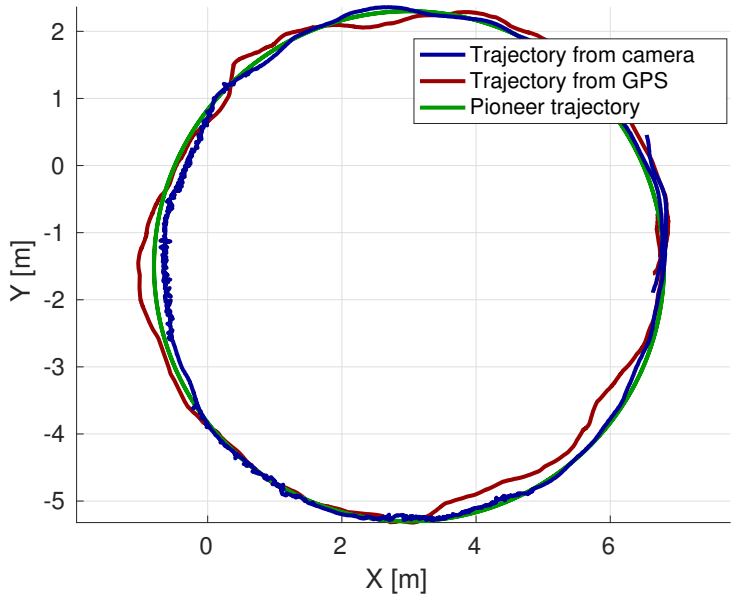
global), esse erro é equivalentemente trazido para o sistema de detecção do marcador. Isto é, o controle é realizado de maneira relativa entre o veículo aéreo e o marcador, reduzindo, assim, a interferência oriunda pela baixa precisão dos outros sensores, como o GPS.

Além disso, o sistema de controle baseado em realimentação visual também foi avaliado. Isso foi possível analisando a diferença entre a posição do marcador e da câmera segundo as informações obtidas pelo *ar_track_alvar* (erro de *tracking*). Com essas informações, foi possível calcular a distância euclidiana entre a posição atual do padrão na imagem e a posição esperada (zero de distância horizontal e 10m de altura, eixo Z). O erro euclidiano é calculado separadamente e exposto na Tabela 3.

Tabela 3 – Distâncias euclidianas entre posição real e esperada do VANT nas três componentes e, por último, tridimensionalmente.

	Distância Média (cm)			
Tragetória	X	Y	Z	3D
ZigZag	43,12	1,18	6,71	65,22
Círculo	61,43	10,70	0,68	66,70
Oito	55,74	3,13	3,19	61,33

Figura 35 – Rota esperada, em verde; rota do VANT, dada pelos dados de GPS, em vermelho; e rota estimada pelas imagens da câmera, em azul.



Fonte: Produção do próprio autor.

A Tabela 3 traz, assim como no caso do erro de trajetória, a constatação de que o erro de *tracking* também é pequeno. A distância entre o VANT e o marcador foi, em média, de 0,61m em X, 0,11m em Y e 10,01m em Z, que é próximo ao esperado. Também, nesse caso, é possível observar que a maior distância relativa está no eixo X, visto que a maior componente da velocidade do padrão nos experimentos realizados, na perspectiva da câmera, está nesse mesmo eixo. Isso acontece pelo fato de que o veículo aéreo tende a se alinhar com a orientação do marcador, ajustando o eixo X do sistema de visão (câmera) com o do robô terrestre.

No experimento com trajetória em ZigZag, o erro de *tracking* é devido aos movimentos de translação, pois quando o robô terrestre gira em seu próprio eixo, só há movimento rotacional. Nesse caso, o marcador permanece no centro da imagem, resumindo o *tracking* a apenas minimização das diferenças entre os ângulos dos veículos.

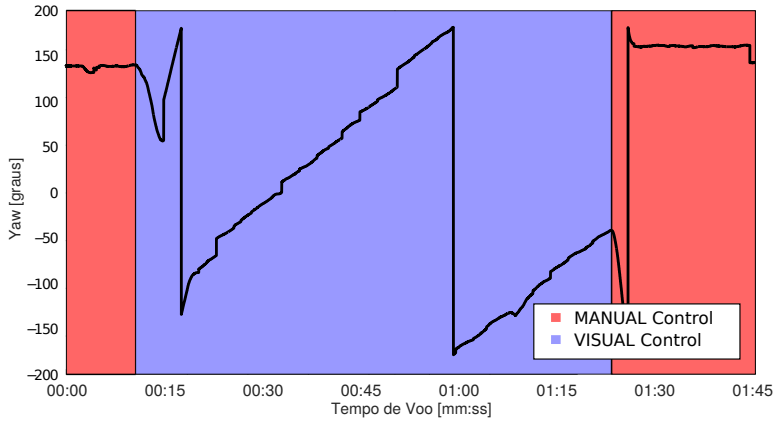
Além disso, podemos sugerir que esses erros obtidos são resultado das limitações oriundas do próprio sistema de visão, causando um atraso estimado de 104,4 ms entre a captura da imagem e o comando dado pela Raspiberry Pi (Tabela 1).

O comportamento do ângulo de *yaw* obtido a partir das informações extraídas do VANT durante a trajetória de Círculo e ZigZag é representado na Figura 36. Esses dados são coloridos de acordo com o modo de controle: manual, em verde e realimentação visual, em

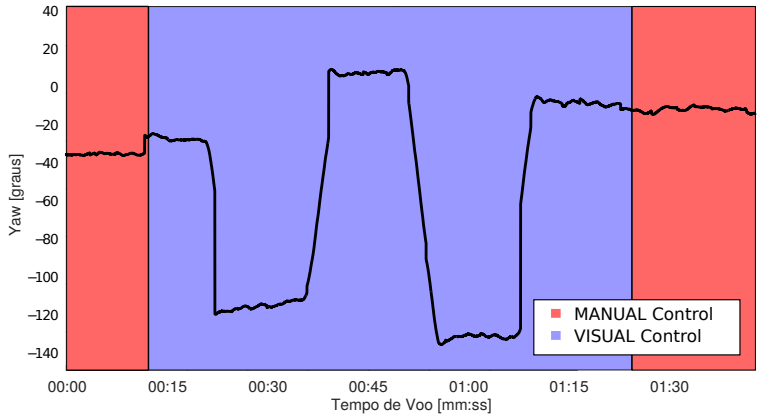
lilás.

Figura 36 – Comportamento do ângulo de *yaw* obtido a partir das informações extraídas do VANT durante a trajetória de Círculo e ZigZag.

(a) Trajetória Circular



(b) Trajetória em ZigZag



Fonte: Produção do próprio autor.

Podemos notar, a partir da Figura 36a, que os valores do ângulo de *yaw*, em relação ao modo de controle baseado em realimentação visual, apresentam um incremento linear, exceto em 0:17min e 1:00min. Em tais instantes, há um salto de $+180°$ a $-180°$, devido à forma como o Pixhawk lida com os ângulos de rotação, conforme descrito no Capítulo 5. Esse comportamento linear é um reflexo do movimento circular realizado pelo robô, ou seja, o VANT não só segue a posição do marcador, mas também sua orientação.

Finalmente, na Figura 36b, é possível observar um fenômeno similar para a experiência em forma de ZigZag. A particularidade dessa trajetória baseia-se no fato de que o VANT executa movimentos retilíneos e de rotação separadamente. Podemos ver isso em instantes 0:36min e 0:51min, quando o VANT gira positivamente (para a direita) e negativo (para a esquerda), respectivamente. Entre esses instantes, o ângulo de *yaw* é quase inalterado, ou

seja, os robôs seguem uma linha reta. As pequenas variações nestas seções retilíneas (por exemplo, de 0:40min a 0:51min) são justificadas por fatores externos.

6.7 Tarefa (as três missões)

Para comprovar que o sistema proposto oferece ao veículo a possibilidade de cumprir tarefas totalmente de forma autônoma, um teste que explora todas as missões apresentadas foi desenvolvido. Além disso, é pretendido que o sistema seja capaz de atuar em ambientes externos, estando, portanto, exposto à interferências do meio, com o vento e a variação de luminosidade.

O teste consiste em decolagem automática, busca e seguimento por meio de marcadores visuais e pouso automático. A tarefa terá o sistema de visão como principal fonte de realimentação para as tomadas de decisão e atitude.[4]

A primeira etapa é dada pela decolagem do veículo, que, até então, não obteve a detecção de nenhum marcador. O VANT levanta voo seguindo a rota definida pelo usuário, mantendo os ganhos e configuração de movimentos previamente estabelecidos.

O equipamento foi carregado com uma trajetória retilínea de subida, com velocidade máxima de deslocamento vertical de 2 m/s. A trajetória percorrida pelo veículo nessa missão pode ser observada na Figura 37, em que é possível observar que o VANT ascende retilineamente.

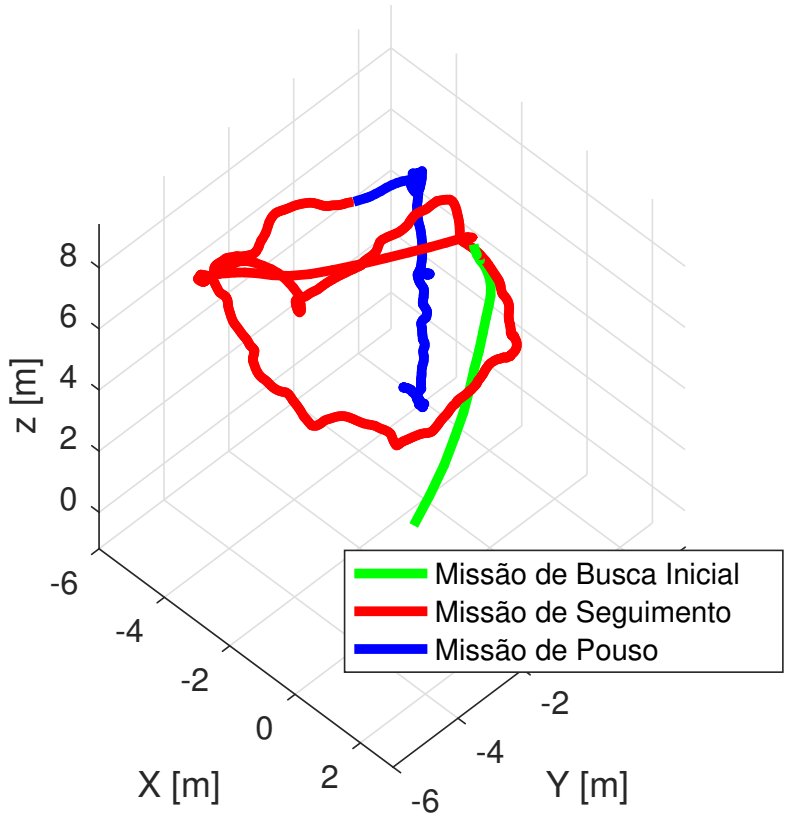
Durante essa parte do voo, o sistema de visão já está acionado e pronto para captar os padrões que eventualmente apareçam. Um fato importante nessa etapa é que a medida que o veículo for incrementando sua altitude, o campo de visão do sistema de captura de imagens aumenta, fazendo com que haja maior probabilidade de um marcador visual seja captado nas imagens. É importante ressaltar que há uma relação entre a capacidade de detecção de padrões em função da altura relativa do equipamento, como já discutido na Seção 6.6. Na Figura 38, é possível compreender melhor essa situação.

Ao mesmo tempo que o VANT levanta voo, um robô terrestre (Pioneer), configurado para realizar uma trajetória circular de 4 metros de raio, é ativado. Foi posto um marcador visual na parte de cima desse robô, de forma que ele atue como um alvo móvel, devendo ser seguido pelo VANT já em voo. Na Figura 39, é possível observar a imagem coletada pelo veículo em um instante antes da primeira detecção e a imagem referente ao exato momento em que o VANT detecta a marca e, então, altera sua missão.

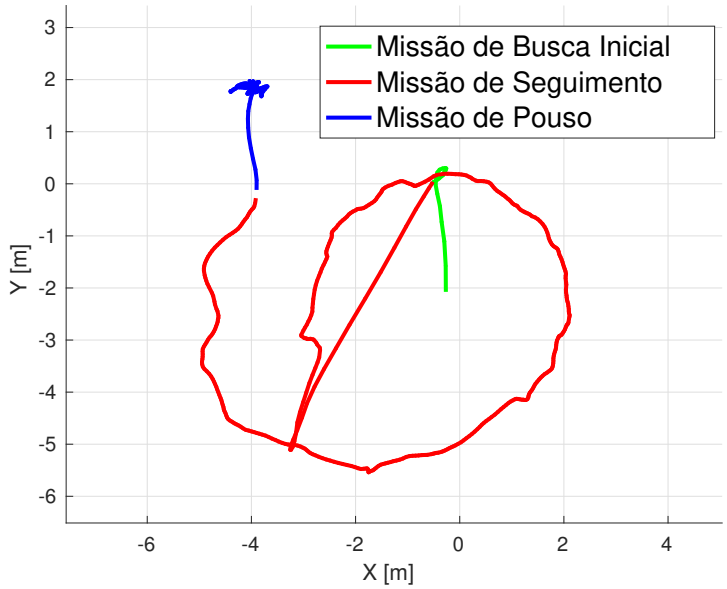
[4] Esse teste foi gravado e pode ser conferido em: <https://www.youtube.com/watch?v=0Zq3iUhUchg>.

Figura 37 – Trajetória percorrida.

(a) Representação tridimensional

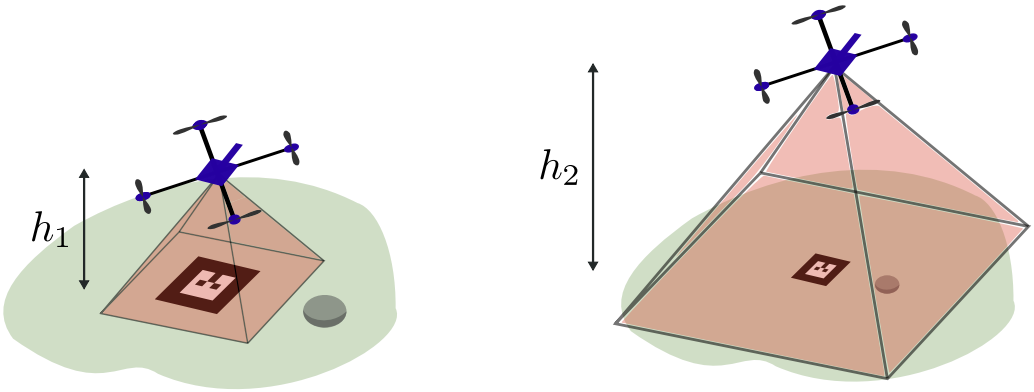


(b) Representação bidirecional (X - Y)



Fonte: Produção do próprio autor.

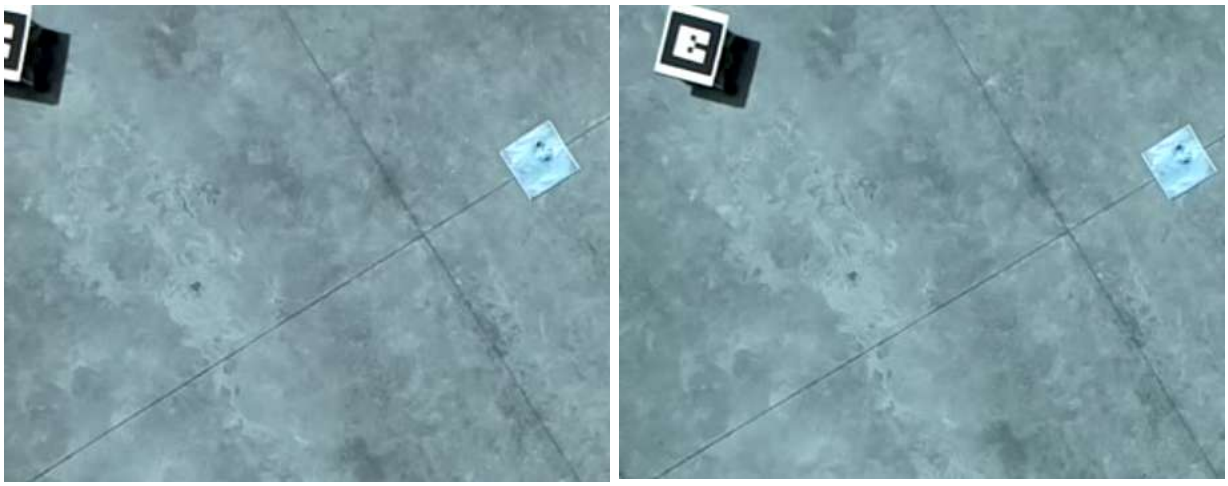
Figura 38 – Relação entre altura do veiculo e FOV do sistema de visão.



Fonte: Produção do próprio autor.

Figura 39 – Instante anterior à primeira detecção e instante da primeira detecção.

(a) Instante anterior à detecção (b) Instante da detecção



Fonte: Produção do próprio autor.

Essa parte do teste tem objetivo de confirmar a capacidade do veículo de levantar voo por conta própria e, a partir da detecção de um padrão, abandonar a trajetória de subida inicial e passar a perseguir o alvo (quando o padrão encontrado for referente ao processo de seguimento).

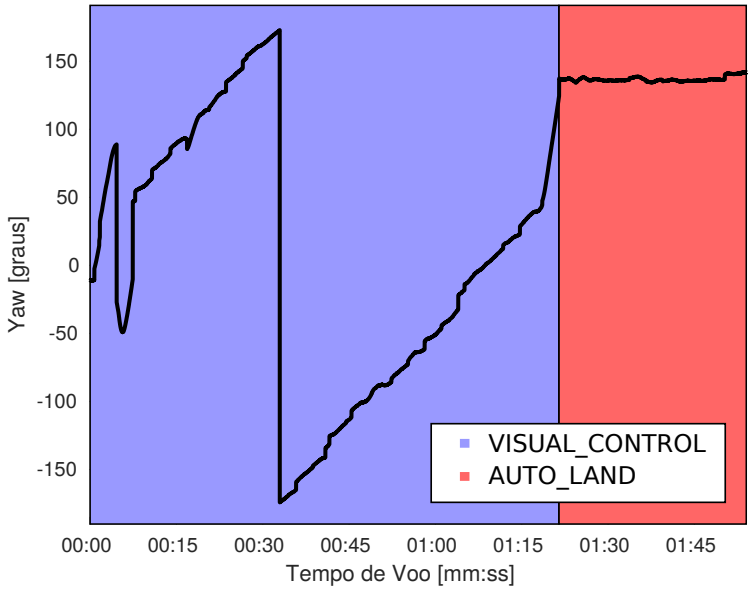
Nesse ponto, o teste passa pela atuação da segunda missão, que é pertinente ao seguimento de alvo móvel. Nela, o veículo tem o único objetivo de se manter na rota desenvolvida pelo padrão, que nesse caso, estará atrelada à movimentação do robô terrestre, esboçando uma cooperação entre robô terrestre e aéreo.

As características e peculiaridades dessa missão foram testadas e analisadas na Seção 6.6,

seguindo o mesmo padrão de desempenho. O robô Pioneer desenvolve, nesse experimento, uma rota circular, semelhante à já testada na seção supracitada.

Na figura 40, verifica-se que, além de seguir a posição do robô terrestre, o VANT varia sua orientação adequadamente, justificado pelo gráfico crescente e praticamente constante entre 00:15 min e 01:14 min.

Figura 40 – Comportamento do ângulo de YAW durante o experimento.



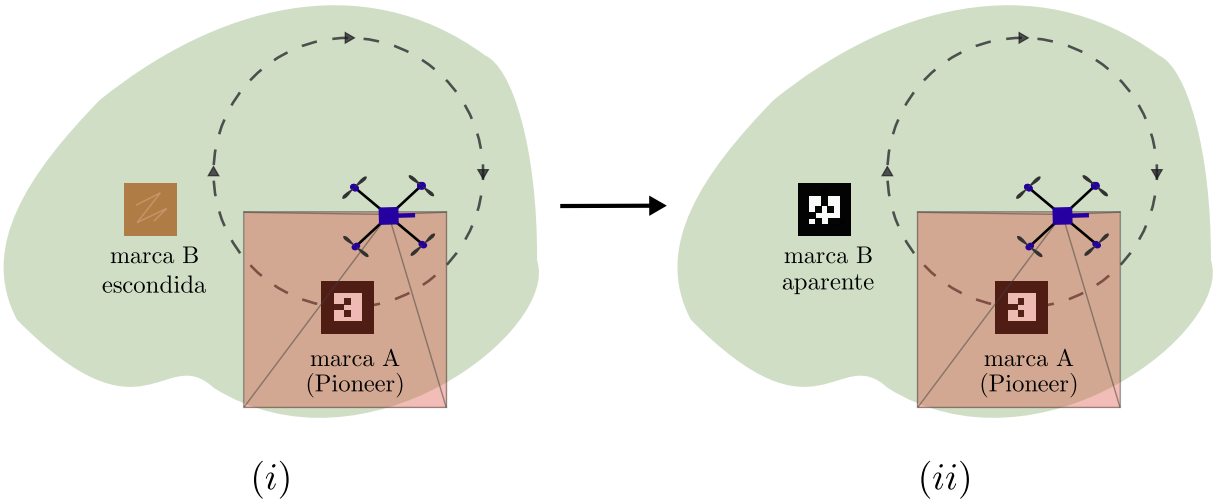
Fonte: Produção do próprio autor.

A etapa seguinte, e última, é a apresentação do marcador visual referente ao pouso, que nesse caso aconteceu na segunda volta do robô terrestre (ii) após o inicio do processo de seguimento. Essa dinâmica é ilustrada na Figura 41. O objetivo é comprovar a capacidade do equipamento em compreender diferentes marcadores na cena e cumprir a hierarquia de decisões do algorítmo.

O VANT, ao confirmar que o marcador do pouso está presente na cena, deixa de seguir o robô e passa a ignorar a presença de outros marcadores, indicando que ativou sua última missão, o pouso.

O teste se encaminha ao fim quando o VANT, na missão de pouso, vai em direção a marca referente (mantendo a altura pertinente). Ao atingir uma distância horizontal mínima do posicionamento do padrão, configurada para 50 cm, o veículo inicia o pouso, situação em que qualquer marcador passa a ser desconsiderado. Nesse instante, a única ação do VANT é o pouso, é também a última parte da tarefa.

Figura 41 – Dinâmica do experimento na missão de pouso. (i) Na primeira volta, o marcador de pouso B está oculto. (ii) Na segunda volta, o marcador B é revelado.



Fonte: Produção do próprio autor.

A trajetória percorrida pelo veículo nessa missão pode ser observada na Figura 37 na cor azul. O VANT, então, pousa sem alteração de posição horizontal, com velocidade de deslocamento máximo de 1 m/s, e sem variação de orientação.

7 CONCLUSÃO E TRABALHOS FUTUROS

VANTs são equipamentos versáteis, capazes de cobrir grandes áreas e obter dados a partir de um ângulo privilegiado. Entretanto, grande parte da aplicação atual depende de ação e controle manual. Este trabalho propõe maior automatização da movimentação e tomada de decisão do equipamento por meio de realimentação visual.

A aplicação pretendida depende de uma adequada aquisição de imagens e detecção de marcadores. Os primeiros testes (ainda com o veículo em solo) confirmaram a robustez do sistema de maneira que a detecção dos marcadores foi realizada suficientemente rápida. Além disso, o processo de perda e recaptura do marcador não se mostrou um problema, sendo, ainda, resguardado pela rotina de se manter a última posição pretendida até que se detecte novamente o marcador (Modo de Segmento de Padrão).

Uma vez garantida uma estável detecção de padrões, é possível realizar a interpretação destes marcadores. Nesta etapa, verificou-se a capacidade de identificar diferentes padrões e responder corretamente à esta entrada de informações. Seja para controlar a movimentação do equipamento ou para ordenar a tomada de decisões, o sistema apresentou fluidez de comandos, mantendo o veículo estável durante todos os experimentos. Uma nova possibilidade de leitura e detecção de padrões móveis é dada pelo pacote $ArUco$, que é similar ao ar_track_alvar, empregando algumas técnicas mais robustas, o que poderia trazer maior agilidade e segurança ao processo. A avaliação deste novo pacote, por enquanto, é tratado como trabalho futuro.

Uma das grandes contribuições deste trabalho envolve o módulo de seguimento de padrão móvel que permite ao VANT replicar satisfatoriamente a trajetória desenvolvida pelo marcador visual específico que estava acoplado a um robô terrestre. O desempenho do conjunto foi adequado ao problema proposto, visto que o sistema foi capaz de operar em situações limite para o robô terrestre.

Essa parcela do trabalho também foi incrementada com rotinas de segurança e tomadas de atitude padrão para tratar situações indesejadas ou inesperadas. Todos estes mecanismos permitem ao veículo a capacidade de operar em ambiente externo, sob influência de variação de iluminação, vento e certa imprecisão dos próprios sensores embarcados, como o GPS.

A validação do sistema completo foi feita a partir de três missões, que correspondem a

uma tarefa totalmente autônoma, em que o VANT decola em busca do marcador visual e passa a segui-lo quando o detecta. Também de forma automática, o veículo realiza o pouso quando recebe o estímulo pertinente, representado por outro marcador visual.

Algumas decisões durante a implementação do sistema o confere características de integração interessantes. A escolha por manter a movimentação e tomada de decisões do veículo com base em um referencial fixo local facilita a inserção de novos atuadores no sistema, aproximando o tema à esfera de cidades digitais e IoT uma vez que parâmetros e marcadores obtidos podem ser compartilhados entre um robô e outros.

Uma hipótese seria a cooperação de robôs terrestres e aéreos, em que um VANT interligado ao sistema teria a capacidade de fazer um voo de reconhecimento e gerar um mapa de deslocamento para os outros robôs, de maneira que uma vez solicitada uma tarefa, o agente com habilidades adequadas ao comprimento da tarefa seja escalado automaticamente e possa chegar até o local solicitado autonomamente.

Ainda nesse contexto, é possível que no sistema exista agentes com diferentes características, como um VANT ágil, com apenas um sistema de câmera, outro com um sistema de *payload releaser*, e até um robô terrestre com capacidade de carga maior. Esse último tem a tendência natural de acumular erros durante seu deslocamento apesar de todo os sistemas de compensação existentes. Uma solução possível seria a utilização de um VANT que possa identificar o marcador de destino do robô terrestre e com isso corrigir a trajetória do mesmo, mantendo precisão conforme o ambiente de atuação e a necessidade do solicitante da ação.

REFERÊNCIAS BIBLIOGRÁFICAS

AMORIM, L. A. et al. Construção de Mosaico Utilizando Imagens Aéreas Adquiridas de Forma Autônoma. In: *Anais do XII Simpósio Brasileiro de Automaćão Inteligente (SBAI'2015)*. [S.l.: s.n.], 2015.

APVRILLE, L.; TANZI, T.; DUGELAY, J.-L. Autonomous drones for assisting rescue services within the context of natural disasters. In: IEEE. *General Assembly and Scientific Symposium (URSI GASS), 2014 XXXIth URSI*. [S.l.], 2014. p. 1–4.

AR_TAG. 2017. <http://wiki.ros.org/ar_track_alvar>. Ar_track_alvar.

ARUCO. 2018. <http://www.uco.es/investiga/grupos/ava/node/26>. ArUco.

ATOEV, S. et al. Data analysis of the mavlink communication protocol. In: IEEE. *Information Science and Communications Technologies (ICISCT), 2017 International Conference on*. [S.l.], 2017. p. 1–3.

BARRETO, O. F. G.; SOUZA, M. R. de. As cidades digitais e o teletrabalho. In: *Anais do Congresso Brasileiro de Processo Coletivo e Cidadania*. [S.l.: s.n.], 2017.

BASTOS, K. *Céu de brigadeiro*. 2011. Fisikanarede.blogspot.com.br.

BENAVIDEZ, P. et al. Landing of an ardrone 2.0 quadcopter on a mobile base using fuzzy logic. In: IEEE. *World Automation Congress (WAC), 2014*. [S.l.], 2014. p. 803–812.

BUTZKE, J. et al. Planning for a ground-air robotic system with collaborative localization. In: IEEE. *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. [S.l.], 2016. p. 284–291.

CARVALHO, J. P. et al. Landing a uav in a dynamical target using fuzzy control and computer vision. In: *Anais do XXI Congresso Brasileiro de Automática (CBA'2016)*. [S.l.: s.n.], 2016.

CHI, H.-L.; KANG, S.-C.; WANG, X. Research trends and opportunities of augmented reality applications in architecture, engineering, and construction. *Automation in construction*, Elsevier, v. 33, p. 116–122, 2013.

CUONG. *Basic knowledge about quadcopter (part 2)*. 2017. Cuongtv.com.

FARIA, R. R. de; COSTA, M. E. A inserção dos veículos aéreos não tripuláveis (drones) como tecnologia de monitoramento no combate ao dano ambiental. *Revista Ordem Pública*, v. 8, n. 1, p. 81–103, 2015.

HARTLEY, R.; ZISSERMAN, A. *Multiple View Geometry in Computer Vision*. Cambridge: Cambridge University Press, 2003.

HOFFMANN, G. M. et al. Quadrotor helicopter flight dynamics and control: Theory and experiment. In: *Proc. of the AIAA Guidance, Navigation, and Control Conference*. [S.l.: s.n.], 2007. v. 2, p. 4.

JENSEN, T.; ZELLER, L.; APAN, A. The use of an unmanned aerial vehicle as a remote sensing platform in agriculture. *Australian Journal of Multi-disciplinary Engineering*, Taylor & Francis, v. 8, n. 2, p. 139–146, 2011.

MÁTHÉ, K.; BUŞONIU, L. Vision and control for uavs: A survey of general methods and of inexpensive platforms for infrastructure inspection. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 15, n. 7, p. 14887–14916, 2015.

MEIER, L. et al. Mavlink: Micro air vehicle communication protocol. *Online]. Tillgänglig: http://qgroundcontrol. org/mavlink/start.[Hämtad 2014-05-22]*, 2013.

MOTLAGH, N. H.; BAGAA, M.; TALEB, T. Uav-based iot platform: A crowd surveillance use case. *IEEE Communications Magazine*, IEEE, v. 55, n. 2, p. 128–134, 2017.

NEO-6. 2011. <https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_(GPS.G6-HW-09005).pdf>. Ublox GPS Modules.

NITSCHKE, C. Marker-based tracking with unmanned aerial vehicles. In: IEEE. *Robotics and Biomimetics (ROBIO), 2014 IEEE International Conference on*. [S.l.], 2014. p. 1331–1338.

OH, H. et al. Indoor uav control using multi-camera visual feedback. *Journal of Intelligent & Robotic Systems*, Springer, v. 61, n. 1-4, p. 57–84, 2011.

PIXHAWK. 2017. <https://pixhawk.org/>. PX4 autopilot.

QUIGLEY, M. et al. Ros: an open-source robot operating system. In: KOBE, JAPAN. *ICRA workshop on open source software*. [S.l.], 2009. v. 3, n. 3.2, p. 5.

RESTAS, A. Drone applications for supporting disaster management. *World Journal of Engineering and Technology*, Scientific Research Publishing, v. 3, n. 03, p. 316, 2015.

ROS. 2017. ROS.org.

SA, F. B. et al. Planejamento de Trajetória para um Robô Móvel usando Imagens capturadas por um VANT. In: *Anais do XX Congresso Brasileiro de Automática (CBA'2014)*. [S.l.: s.n.], 2014.

SAMPAIO, R. P. Sistema de controle de atitude embarcado para vôo autônomo de aviões em escala. Escola Politécnica/Instituto de Matemática, 2006.

ZAKARIA, O. et al. Deterministic method of visual servoing: robust object tracking by drone. In: IEEE. *Computer Graphics, Imaging and Visualization (CGiV), 2016 13th International Conference on*. [S.l.], 2016. p. 414–422.