

CURSO DE CIÊNCIA DA COMPUTAÇÃO – TCC	
() PRÉ-PROJETO (x) PROJETO	ANO/SEMESTRE: 2020/1

ARQUITETURA PARA NAVEGAÇÃO AUTÔNOMA UTILIZANDO DRONES

Matheus Mahnke

Prof. Dalton Solano dos Reis – Orientador

1. INTRODUÇÃO

Todos os anos processos antes feitos por humanos são automatizados, em qualquer área e qualquer campo da ciência, a tecnologia vem avançando para reduzir esforços humanos. Segundo Corrêa (2020, p. 16), “essa tecnologia trouxe mudanças à sociedade, tal como novos modelos de trabalho e novas profissões”.

Diante do crescente uso da tecnologia, um equipamento chama atenção, o drone, que transcende a capacidade de locomoção humana, pois com ele é possível acessar locais inacessíveis de forma terrestre. Por conta disso, a cada ano os drones vêm ganhando mais popularidade. Dentre as opções, os quadrotoros são os mais comuns. De acordo com Lugo e Zell (2014), quadrotoros são uma escolha muito popular por conta de sua robustez, mecânica simples, baixo peso e tamanho pequeno.

Os drones geralmente são equipados com diversos sensores, como altímetro, acelerômetro e giroscópio, porém, apesar de bastante preciso, durante uma navegação baseada nesses sensores pode acontecer acúmulo de erros, aumentando o desvio no decorrer da navegação. Sensores mais modernos ajudam a solucionar esse problema, porém acabam sendo inviabilizados pelo seu alto valor comercial. Segundo Mur-artal e Tardós (2017), dentre as diferentes modalidades de sensores, as câmeras são baratas e podem fornecer informações valiosas do ambiente permitindo o reconhecimento robusto e preciso do local.

De acordo com Martins, Ramos e Mora-Camino (2018), a vantagem de se trabalhar somente com navegação baseada em Visão Computacional é que a solução é simples e tem um baixo custo. Sendo assim, a navegação baseada em câmeras é uma boa solução tanto com a câmera apontada para baixo, permitindo detectar a velocidade horizontal ou até mesmo marcadores, quanto com a câmera frontal, sendo possível utilizar técnicas de processamento de imagem para se auto localizar no espaço ou detectar e desviar de obstáculos.

Entretanto, ao se utilizar técnicas de processamento de imagem em tempo real, o drone se torna dependente da conexão com um equipamento capaz de fazer esse processamento. Segundo Lugo e Zell (2014), a comunicação por meio de uma rede sem fio limita a distância de trabalho do sistema e introduz um atraso entre as informações de dados dos sensores e comandos de controle. Em contrapartida, novos modelos de microcontroladores possuem essa capacidade de processamento e podem ser levados a bordo do drone, de forma portátil, independente de conexão.

O drone pode executar um plano de voo, tanto controlado manualmente, com uma rota fixa baseada em sensores quanto baseado em reconhecimento de imagens. Contudo, nesses casos, ainda é necessário a ação humana na criação das rotas. De acordo com Corrêa (2020, p. 20), para voar autonomamente para uma determinada coordenada geográfica, é necessário um dispositivo receptor de sistemas que fornecem posicionamento geográfico espacial. Nesse contexto, o Global Positioning System (GPS) é uma importante ferramenta para a navegação autônoma. Com ele é possível determinar o destino e definir planos de voo em ambientes não conhecidos previamente, aumentando sua aplicabilidade comercial.

Diante disso, este trabalho propõe uma arquitetura de navegação baseada em GPS, aliado aos sensores e imagens de câmera para o reconhecimento e desvio de obstáculos em voo, com processamento a bordo resultando em uma navegação totalmente autônoma.

1.1. OBJETIVOS

O objetivo deste trabalho é disponibilizar uma arquitetura de navegação autônoma de drones baseada em GPS e Visão Computacional.

Os objetivos específicos são:

- detectar e desviar de obstáculos visíveis pela câmera;
- executar um plano de voo baseado em GPS;
- reconhecer e pousar em marcadores;
- definir a arquitetura para a navegação independente de servidor.

2. TRABALHOS CORRELATOS

A seguir são apresentados três trabalhos correlatos. Na seção 2.1 será abordado o trabalho Mur-artal e Tardós (2017), que descreve um método de estabilização baseado em câmera. Na seção 2.2 será descrito o trabalho de Corrêa (2020), que consiste em um sistema de navegação autônoma de drone baseando-se em GPS. Por fim, na seção 2.3 é apresentado o trabalho de Martins, Ramos e Mora-Camino (2018), que utiliza técnicas de processamento de imagem para reconhecimento e desvio de obstáculos em voo.

2.1. ORB-SLAM2: AN OPEN-SOURCE SLAM SYSTEM FOR MONOCULAR, STEREO AND RGB-D CAMERAS

O trabalho de Mur-artal e Tardós (2017) propõe um sistema Simultaneous Localization and Mapping (SLAM) de código aberto para câmeras monoculares, estéreo e Red, Green, Blue and Depth (RGB-D) incluindo funcionalidades de fechamento de *loop* e relocalização. Seu código foi publicado no intuito de disponibilizar uma solução SLAM pronto para uso. Técnicas de SLAM criam um mapa de um ambiente desconhecido e localizam o sensor no mapa em tempo real. A Figura 1 mostra um mapa gerado pelo ORB-SLAM2 com uma câmera estéreo, com vários fechamentos de *loop*.

Figura 1 – Trajetória e reconstrução esparsa de um sistema estéreo em um ambiente urbano com muitos fechamentos de *loop*



Fonte: Mur-artal e Tardós (2017, p. 1).

O reconhecimento de lugar é a chave do SLAM para relocalizar a câmera após perder o acompanhamento, seja por movimentos bruscos, oclusão da câmera ou reinicialização do sistema. Então o sistema mantém um grafo de visibilidade que liga outros *keyframes* observando os pontos comuns e uma árvore de caminho mínimo conectando todos os pontos. Essas estruturas permitem que o rastreamento e mapeamento operem localmente.

Os sistemas Monocular e Estéreo contam com um Bundle Adjustment (BA). O BA é utilizado para otimizar a orientação da câmera no rastreamento, minimizando os erros de projeção entre os pontos encontrados e a localização real (*motion-only* BA) para otimizar pontos de mapeamento e janela de *keyframes* local (local BA) e para otimizar todos os *keyframes* e pontos após o fechamento de *loop* (*full* BA).

Para testes, o ORB-SLAM2 foi comparado com outros sistemas SLAM mais atuais, utilizando três diferentes conjuntos de dados. O conjunto de dados KITTI, que contém sequências de imagens estéreo gravadas de um carro em ambientes de rodovias, com vários *loops*. O conjunto de dados EuRoC contendo sequências de

imagens estéreo gravadas de um Micro Aerial Vehicle (MAV), voando entre diferentes cômodos em um ambiente industrial. Já o conjunto de dados TUM RGB-D contendo sequências de Sensores RGB-D em diferentes categorias, usados para testar o sistema SLAM sob diferentes texturas, iluminações e estruturas.

As comparações mostraram que o ORB-SLAM2 atingiu acurácia maior em comparação aos outros sistemas SLAM mais atuais. No teste com o KITTI se tornou a melhor solução SLAM, atingindo zero desvio de localização em áreas já mapeadas. Os resultados dos testes com RGB-D demonstraram que quanto mais precisamente direcionada for a câmera, o BA tem melhor performance em comparação com o Iterative Closest Points (ICP).

2.2. DRONE AUTÔNOMO: VIGILÂNCIA AÉREA DE ESPAÇOS EXTERNOS

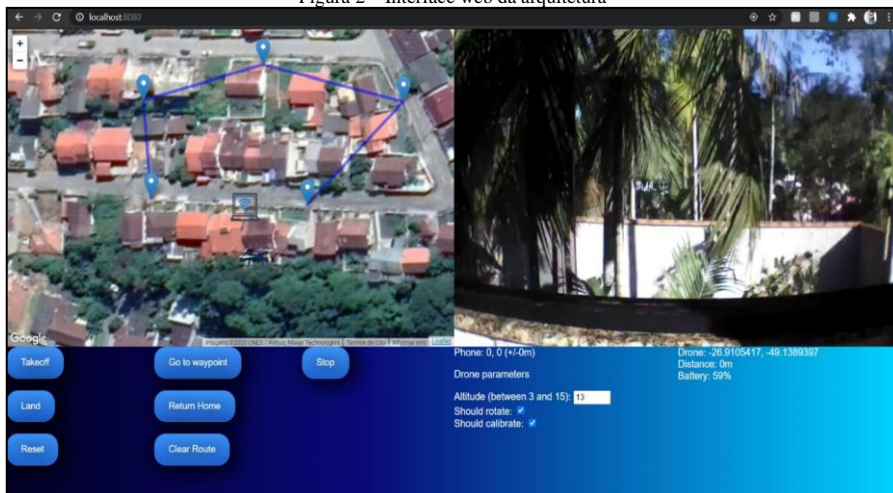
O trabalho de Corrêa (2020) consiste em uma arquitetura de voo autônomo aplicada no modelo Parrot AR.Drone 2.0, com navegação baseada em GPS visando aplicabilidade em vigilância de áreas externas. O drone carrega a bordo um giroscópio de 3 eixos, acelerômetro de 3 eixos, magnetômetro, sensor de pressão, altímetro ultrassônico, câmera vertical para medição da velocidade horizontal e uma câmera frontal transmitindo em tempo real para o servidor.

A arquitetura visa três diferentes dispositivos, sendo esses o drone, o servidor e o cliente (interface web). Para isso foi utilizado o Node.js, utilizando bibliotecas Node Package Manager (NPM) para a integração dos dispositivos em tempo real, sendo necessária a comunicação para o script de voo. Por navegar com base no GPS a arquitetura não visou uma navegabilidade precisa, utilizando apenas os sensores e recursos de odometria padrão do AR.Drone.

O uso do sistema se dá através de uma interface web, no qual o usuário pode cadastrar e selecionar uma rota, decolar o drone, executar o trajeto, retornar a base, pousar e limpar a rota criada. A arquitetura permite configurar um plano de voo para que o drone se direcione a uma determinado local. O sistema também faz o retorno automático caso identifique pouca carga de bateria.

A Figura 2 demonstra a tela de operacionalidade da arquitetura, é nela em que o usuário obtém comandos de voo, visão da câmera e informações dos sensores. Por fim, o resultado se deu na execução de três cenários de teste, executando as funções programadas, finalizando com uma margem de erro de até 6 metros na precisão.

Figura 2 – Interface web da arquitetura



Fonte: Corrêa (2020, p. 52).

2.3. A COMPUTER VISION BASED ALGORITHM FOR OBSTACLE AVOIDANCE (OUTDOOR FLIGHT)

O trabalho de Martins, Ramos e Mora-Camino (2018) têm como objetivo implementar técnicas de processamento de imagem para reconhecer e desviar de obstáculos durante o voo de um Unmanned Aerial Vehicle

(UAV), usando apenas a câmera frontal. Foi combinado *hardware* e *software* para capturar e tratar as imagens em tempo real e embarcar o algoritmo a bordo do drone.

Para a plataforma UAV, foi utilizado um quadrotor usando um controlador Pixhawk que é uma placa controladora de voo de código aberto, responsável pelo controle em baixo nível do quadrotor. O UAV é equipado com giroscópio, acelerômetro, magnetômetro, barômetro, um microcontrolador Raspberry Pi para o processamento de imagem, bem como um *software* embarcado que implementa as funções básicas de controle. Para o *software* foi usado a biblioteca gráfica OpenCV de código aberto e o *firmware* ROS Indigo embarcado no Raspberry Pi a bordo do drone.

As técnicas de processamento de imagem consistem em: capturar a imagem, aplicar escala de cinza na imagem, desfocar, detectar bordas, encontrar contornos, desenhar contornos, identificar obstáculos e áreas livres e por fim repetir o processo. A Figura 3 mostra a imagem capturada a partir da câmera frontal de um drone e a Figura 4 mostra os filtros já aplicados com as áreas mapeadas.

Figura 3 – Foto capturada, um drone voando



Fonte: Martins, Ramos e Mora-Camino (2018, p. 1).

Figura 4 - Áreas livres



Fonte: Martins, Ramos e Mora-Camino (2018, p. 2).

Nos testes aos quais foi submetido o drone desviou de todos os obstáculos que encontrou em seu caminho, porém o projeto limitou-se a desviar de obstáculos e não a seguir um plano de voo, e, dessa forma, cada desvio alterou o destino do drone. O algoritmo se mostrou eficaz para evitar colisões em voos externos, mas, por se basear em uma câmera, o algoritmo se torna sensível a mudanças de iluminação.

3. PROPOSTA DO PROTÓTIPO

Nesse capítulo apresenta-se a relevância do trabalho na área da saúde, na questão ambiental e na prática. Será também exibido um quadro comparativo entre os trabalhos correlatos e o proposto, finalizando com a metodologia e o cronograma para o desenvolvimento do projeto.

3.1. JUSTIFICATIVA

A aplicação de drones autônomos na área da saúde pode ser uma ótima abordagem em situações emergenciais. Segundo Scott e Scott (2017), a entrega oportuna de medicamentos, sangue e vacinas necessários com urgência pode ser facilitada com o uso de drones.

Drones são uma opção ecológica para o uso de motocicletas para entregas. De acordo com Park, Kim e Suh (2018), apesar de o Potencial de Aquecimento Global (PAG) ser maior para a produção de energia elétrica em

comparação a gasolina, o PAG gerado pela queima da gasolina foi 24 vezes maior do que a fase de produção, levando a uma diferença significativa no PAG total. Portanto, a emissão de gases de efeito estufa pode ser reduzido se o método de entrega por motocicletas for convertido para drones. Nesse cenário o uso de técnicas para navegação autônoma se destaca ao auxiliar ou até mesmo substituir a ação humana no processo.

Conforme detalhado no Quadro 1, todos os trabalhos implementam conceitos de navegação autônoma, porém com objetivos diferentes. O trabalho correlato de Martins, Ramos e Mora-Camino (2018) aponta uma solução independente de um sistema terrestre, não necessitando de qualquer infraestrutura para a realização do voo. Também implementa um sistema de reconhecimento e desvio de obstáculos com base na câmera. Em contrapartida, o trabalho de Corrêa (2020) possui um plano de voo dinâmico, baseado em GPS, apresentando-se como uma importante abordagem para aplicações comerciais. Já o trabalho de Mur-artal e Tardós (2017) é único dentre os correlatos, a possuir um sistema SLAM baseado em câmera, capaz de se auto localizar e reconstruir o mapa do caminho percorrido.

Quadro 1- Comparativo dos trabalhos correlatos

Trabalhos Correlatos	Mur-artal e Tardós (2017)	Corrêa (2020)	Martins, Ramos e Mora-Camino (2018)
Características			
usa GPS	não	sim	não
navegação baseada em câmera	sim	não	sim
processamento a bordo	não	não	sim
desvio de obstáculos	não	não	sim
plano de voo dinâmico	não	sim	não
possui sistema SLAM	sim	não	não

Fonte: elaborado pelo autor.

Por meio das informações citada, o trabalho proposto destaca-se pela combinação das técnicas citadas no Quadro 1. O protótipo conta com uma arquitetura de navegação autônoma a bordo, um sistema de auto localização com base em imagens de câmera, sem a necessidade de sensores caros, reduzindo o custo. Constará também com a capacidade de desviar de obstáculos com base em processamento de imagem, decolar, pousar e seguir um plano de voo baseado em GPS de forma autônoma.

Dessa forma, o trabalho poderá trazer contribuições práticas a respeito de navegação autônoma de drones e processamento de imagem em embarcados. Como também representa a possibilidade do aprofundamento sobre o tema de aeronavegabilidade baseado em processamento de imagens. O trabalho expõe um protótipo, que pode ser usado como base para projetos futuros utilizando navegação autônoma com drones.

3.2. REQUISITOS PRINCIPAIS DO PROTÓTIPO A SER TRABALHADO

Os Requisitos Funcionais (RFs) e os Requisitos Não Funcionais (RNFs) do protótipo são:

- o drone deverá seguir um plano de voo baseado em GPS (RF);
- o drone deverá possuir uma câmera frontal (RF);
- o drone deverá possuir uma câmera apontada para baixo (RF);
- o drone deverá possuir um sistema de estabilização com base na câmera e sensores (RF);
- o processamento em voo deverá ser totalmente a bordo, sendo dependente apenas da conexão com GPS (RF);
- o drone deverá desviar de objetos em voo (RF);
- a arquitetura deverá permitir o pouso em um marcador (RF);
- o drone deverá utilizar a biblioteca OpenCV para processar o reconhecimento das imagens (RNF);
- o drone deverá possuir um Raspberry Pi III para processamento de imagem a bordo (RNF).

3.3. METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

- levantamento bibliográfico do hardware: pesquisar formas de levar a bordo do drone um processador capaz de realizar processamento de imagens;
- levantamento bibliográfico da solução: pesquisar formas de realizar o processamento de imagens para estabilizar o drone, desviar de objetos e identificar marcadores;
- elicitação dos requisitos: refinar os requisitos para atingir os objetivos de acordo com os resultados

Comentado [MAW1]: Citadas

Comentado [MAW2]: A u-blox tem um GLONASS que pega coordenadas e junto com dados de correção, vindo da Internet, consegue baixar a margem de erro de mts para centímetros!

- dos levantamentos bibliográficos;
- d) especificação: realizar modelagem do diagrama de classes e do modelo entidade relacionamento a serem utilizados no projeto seguindo os padrões Unified Modeling Language (UML) com a ferramenta Draw.io;
- e) desenvolvimento do protótipo: criar ou fazer alterações em drone existente para carregar um processador, sendo capaz de ler dados dos sensores e enviar instruções de voo;
- f) definição das técnicas de processamento das imagens: definir quais técnicas de processamento de imagem utilizar para atingir os objetivos propostos, baseando-se nas técnicas levantadas nos trabalhos correlatos;
- g) testes: efetuar testes de voo verificando a conformidade com os objetivos propostos.

As etapas serão realizadas nos períodos relacionados no Quadro 2.

Quadro 2 - Cronograma

etapas / quinzenas	2021									
	ago.		set.		out.		nov.		dez.	
	1	2	1	2	1	2	1	2	1	2
levantamento bibliográfico do <i>hardware</i>										
levantamento bibliográfico da solução										
elicitação dos requisitos										
especificação										
desenvolvimento do protótipo										
definição das técnicas de processamento das imagens										
testes										

Fonte: elaborado pelo autor.

4. REVISÃO BIBLIOGRÁFICA

Este capítulo abordará os principais assuntos técnicos a respeito da navegação autônoma de drone. A seção 4.1 tem como objetivo abordar hardwares embarcados no drone, sendo esses o Arduino e o Raspberry Pi. A seção 4.2 apresenta a teoria por trás da detecção de obstáculos, tendo em vista a visão estérea.

4.1. HARDWARE

Atualmente percebe-se uma expansão na área que diz respeito a hardwares de código aberto. De acordo com Bonvoisin *et al.* (2017), o fenômeno é relativamente jovem e com projetos emergentes na última década, embora já tenha vários exemplos proeminentes. Esse movimento permite que equipamentos como o Arduino ganhe popularidade. O Arduino é uma plataforma de prototipagem de código aberto baseada em hardware e softwares fáceis de usar, com ele é possível ler sensores, ligar atuadores e até trafegar dados pela internet.

Embora o Arduino seja uma boa opção na maioria dos casos, outro equipamento também vem ganhando popularidade, é o caso do Raspberry Pi que é uma pequena placa de computador com pinos digitais totalmente programáveis. De acordo com Maksimović *et al.* (2014), análises comparativas com algumas das plataformas de prototipagem Internet of Things (IoT) existentes mostraram que, apesar de algumas desvantagens, o Raspberry Pi continua sendo um computador de baixo custo com uso muito bem-sucedido em diversas aplicações de pesquisa na visão de IoT.

Dê uma olhada no ZED-F9, da u-blox

Comentado [MAW3]: ←

4.2. DETECÇÃO DE OBSTÁCULOS 3D COM PROCESSAMENTO DE IMAGEM

Visão estéreo, segundo Khan *et al.* (2017, p. 381, tradução nossa):

É a maneira de encontrar informações 3D, ou seja, profundidade de várias imagens 2D. Envolve a colocação de duas câmeras separadas por alguma distância, chamada linha de base b , conforme mostrado na Figura 6. O eixo óptico e o eixo x da câmera são denotados por z e x respectivamente. Sejam PL e PR a coordenada x do ponto P (X, Y, Z) do mundo real, conforme projetado na imagem adquirida pelas câmeras esquerda e direita, respectivamente. A distância Z do mundo real do ponto P da câmera pode ser encontrada por meio da trigonometria, empregando a relação de disparidade. A disparidade d é a distância entre as coordenadas PL e PR e é mostrada

na Figura 5¹.

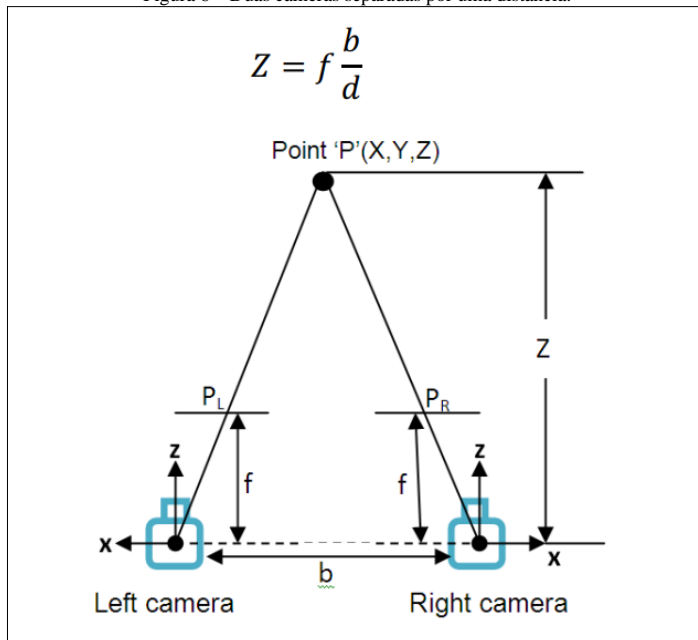
Figura 5 – Cálculo da disparidade entre as coordenadas PL e PR.

$$d = P_L - P_R = \frac{f(X)}{Z} - \frac{f(X-b)}{Z}$$

$$d = f \frac{b}{Z}$$

Fonte: Khan, *et al.* (2017, p. 381).

Figura 6 – Duas câmeras separadas por uma distância.



Fonte: Khan, *et al.* (2017, p. 381).

De acordo com Fan *et al.* (2018), um algoritmo de estimativa de disparidade geralmente consiste em quatro etapas: cálculo de custo, agregação de custo, otimização de disparidade e refinamento de disparidade.

O mapa de disparidade refere-se à diferença aparente de *pixels* entre as imagens tiradas das *webcams* esquerda e direita. Os objetos que estão presentes mais perto da *webcam* parecem mais claros do que os distantes. As distâncias exatas desses objetos podem ser encontradas usando (conforme figura 2). Desta forma, a possibilidade de

¹ Stereovision is the way of finding 3D information i.e. depth from multiple 2D images. It involves placing two cameras separated by at some distance, called baseline *b* as shown in Figure 2. The optical axis and *x*-axis of the camera are denoted by *z* and *x* respectively. Let *P_L* and *P_R* be the *x*-coordinate of the real-world point *P(X,Y,Z)*, as projected in the image acquired by left and right cameras respectively. The real-world distance *Z* of the point *P* from the camera can be found through trigonometry by employing the disparity relationship. The disparity *d* is the distance between coordinates *P_L* and *P_R* and is given by (1).

colisão é detectada a partir de um obstáculo próximo (KHAN *et al.* 2017, p. 382, tradução nossa)².

REFERÊNCIAS

- BONVOISIN, Jérémy *et al.* **What is the “source” of open source hardware?**. Journal of Open Hardware, p. 18. 2017.
- CORRÊA, Diego F. **Drone autônomo: Vigilância aérea de espaços externos**. 2020. 67 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- FAN, Rui *et al.* **Real-time stereo vision for road surface 3-d reconstruction**. In: 2018 IEEE International Conference on Imaging Systems and Techniques (IST). IEEE, 2018. p. 1-6.
- KHAN, Maham *et al.* **Stereovision-based real-time obstacle detection scheme for unmanned ground vehicle with steering wheel drive mechanism**. In: 2017 International Conference on Communication, Computing and Digital Systems (C-CODE). IEEE, 2017. p. 380-385.
- LUGO, Jacobo J.; ZELL, Andreas. **Framework for autonomous on-board navigation with the AR. Drone**. Journal of Intelligent & Robotic Systems, v. 73, n. 1, p. 401-412, 2014.
- MAKSIMOVIĆ, Mirjana *et al.* **Raspberry Pi as Internet of things hardware: performances and constraints**. design issues, v. 3, n. 8, 2014.
- MARTINS, Wander M.; RAMOS, Alexandre C. B.; MORA-CAMINO, Felix. **A Computer Vision Based Algorithm for Obstacle Avoidance**. In: Information Technology-New Generations. Springer, Cham, 2018. p. 569-575.
- MUR-ARTAL, Raul; TARDÓS, Juan D. **Orb-slam2**: An open-source slam system for monocular, stereo, and rgb-d cameras. IEEE Transactions on Robotics, v. 33, n. 5, p. 1255-1262, 2017.
- PARK, Jiyeon; KIM, Solhee; SUH, Kyo. **A comparative analysis of the environmental benefits of drone-based delivery services in urban and rural areas**. Sustainability, v. 10, n. 3, p. 888, 2018.
- SCOTT, Judy E.; SCOTT, Carlton H. **Drone delivery models for healthcare**. In: Proceedings of the 50th Hawaii international conference on system sciences. 2017.

² Disparity map refers to the apparent pixel difference between the images taken from left and right webcams. The objects that are present nearer to the webcam appear lighter than the distant ones. The exact distances from these objects can be found using (2). In this way the possibility of collision is detected from the nearby obstacle.

ASSINATURAS

(Atenção: todas as folhas devem estar rubricadas)

Assinatura do(a) Aluno(a): _____

Assinatura do(a) Orientador(a): _____

Assinatura do(a) Coorientador(a) (se houver): _____

Observações do orientador em relação a itens não atendidos do pré-projeto (se houver):

FORMULÁRIO DE AVALIAÇÃO – PROFESSOR TCC I

Acadêmico(a): _____

Avaliador(a): _____

ASPECTOS AVALIADOS ¹		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO O tema de pesquisa está devidamente contextualizado/delimitado?			
	O problema está claramente formulado?			
	2. OBJETIVOS O objetivo principal está claramente definido e é passível de ser alcançado?			
	Os objetivos específicos são coerentes com o objetivo principal?			
	3. JUSTIFICATIVA São apresentados argumentos científicos, técnicos ou metodológicos que justificam a proposta?			
ASPECTOS METODOLÓGICOS	São apresentadas as contribuições teóricas, práticas ou sociais que justificam a proposta?			
	4. METODOLOGIA Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?			
	Os métodos, recursos e o cronograma estão devidamente apresentados?			
	5. REVISÃO BIBLIOGRÁFICA (atenção para a diferença de conteúdo entre projeto e pré-projeto) Os assuntos apresentados são suficientes e têm relação com o tema do TCC?			
	6. LINGUAGEM USADA (redação) O texto completo é coerente e redigido corretamente em língua portuguesa, usando linguagem formal/científica?			
	A exposição do assunto é ordenada (as ideias estão bem encadeadas e a linguagem utilizada é clara)?			
	7. ORGANIZAÇÃO E APRESENTAÇÃO GRÁFICA DO TEXTO A organização e apresentação dos capítulos, seções, subseções e parágrafos estão de acordo com o modelo estabelecido?			
	8. ILUSTRAÇÕES (figuras, quadros, tabelas) As ilustrações são legíveis e obedecem às normas da ABNT?			
	9. REFERÊNCIAS E CITAÇÕES As referências obedecem às normas da ABNT?			
	As citações obedecem às normas da ABNT?			
	Todos os documentos citados foram referenciados e vice-versa, isto é, as citações e referências são consistentes?			

PARECER – PROFESSOR DE TCC I OU COORDENADOR DE TCC (PREENCHER APENAS NO PROJETO):

O projeto de TCC será reprovado se:

- qualquer um dos itens tiver resposta NÃO ATENDE;
- pelo menos 4 (**quatro**) itens dos **ASPECTOS TÉCNICOS** tiverem resposta ATENDE PARCIALMENTE; ou
- pelo menos 4 (**quatro**) itens dos **ASPECTOS METODOLÓGICOS** tiverem resposta ATENDE PARCIALMENTE.

PARECER: () APROVADO () REPROVADO

Assinatura: _____ Data: _____

¹ Quando o avaliador marcar algum item como atende parcialmente ou não atende, deve obrigatoriamente indicar os motivos no texto, para que o aluno saiba o porquê da avaliação.

FORMULÁRIO DE AVALIAÇÃO – PROFESSOR AVALIADOR

Acadêmico(a): Matheus Mahnke

Avaliador(a): Miguel Wisintainer

ASPECTOS AVALIADOS ¹		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO O tema de pesquisa está devidamente contextualizado/delimitado?	x		
	O problema está claramente formulado?	x		
	1. OBJETIVOS O objetivo principal está claramente definido e é passível de ser alcançado?	x		
	Os objetivos específicos são coerentes com o objetivo principal?	x		
	2. TRABALHOS CORRELATOS São apresentados trabalhos correlatos, bem como descritas as principais funcionalidades e os pontos fortes e fracos?	x		
	3. JUSTIFICATIVA Foi apresentado e discutido um quadro relacionando os trabalhos correlatos e suas principais funcionalidades com a proposta apresentada?	x		
	São apresentados argumentos científicos, técnicos ou metodológicos que justificam a proposta?	x		
	São apresentadas as contribuições teóricas, práticas ou sociais que justificam a proposta?	x		
	4. REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO Os requisitos funcionais e não funcionais foram claramente descritos?	x		
	5. METODOLOGIA Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?	x		
	Os métodos, recursos e o cronograma estão devidamente apresentados e são compatíveis com a metodologia proposta?	x		
	6. REVISÃO BIBLIOGRÁFICA (atenção para a diferença de conteúdo entre projeto e pré-projeto) Os assuntos apresentados são suficientes e têm relação com o tema do TCC?	x		
	As referências contemplam adequadamente os assuntos abordados (são indicadas obras atualizadas e as mais importantes da área)?	x		
ASPECTOS METODOLÓGICOS	7. LINGUAGEM USADA (redação) O texto completo é coerente e redigido corretamente em língua portuguesa, usando linguagem formal/científica?	x		
	A exposição do assunto é ordenada (as ideias estão bem encadeadas e a linguagem utilizada é clara)?	x		

PARECER – PROFESSOR AVALIADOR: (PREENCHER APENAS NO PROJETO)

O projeto de TCC ser deverá ser revisado, isto é, necessita de complementação, se:

- qualquer um dos itens tiver resposta NÃO ATENDE;
- pelo menos 5 (cinco) tiverem resposta ATENDE PARCIALMENTE.

PARECER: (x) APROVADO () REPROVADO

Assinatura: MIGUEL ALEXANDRE WISINTAINER 767.906.309-53 Data: 22/06/2021 _____

¹ Quando o avaliador marcar algum item como atende parcialmente ou não atende, deve obrigatoriamente indicar os motivos no texto, para que o aluno saiba o porquê da avaliação.