

## Plataforma computacional para auxiliar na busca e resgate de vítimas em situações de desastres naturais

### Computational platform to aid in the search and rescue of victims in situations of natural disasters

**Matheus Fellype Ferraz**

[matf.ferraz@gmail.com](mailto:matf.ferraz@gmail.com)

Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil

**Marco Aurelio Wehrmeister**

[wehrmeister@utfpr.edu.br](mailto:wehrmeister@utfpr.edu.br)

Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil

#### RESUMO

A utilização de veículos aéreos não tripulados (VANT) concedem hoje um novo patamar tecnológico para operações remotamente controladas. Este trabalho realizou o estudo, integração e testes de um protótipo de um VANT, com o uso de diferentes plataformas de controle (Navio2, APM2.0, Erle-Brain) incorporadas a protótipos multirrotores, cuja finalidade principal é a de realizar buscas em áreas de isolamento remoto. Ao longo do projeto foram realizadas etapas de construção de hardware, calibração e configurações de software, além do estudo e incorporação de uma câmera térmica sobre o protótipo. Foram realizadas simulações de missões em um cenário com conceitos similares, onde um VANT realiza reconhecimento e desvio de obstáculos enquanto inspeciona uma área. A parte física do protótipo encontra-se montada e operacional, já tendo sido realizados testes de voo com sucesso. Foi feita a implementação de um pacote de ROS para a comunicação dos dados obtidos pela câmera térmica, sendo desenvolvido um tutorial referente a montagem e instalação do protótipo e do pacote.

**PALAVRAS-CHAVE:** VANT. Câmera Térmica. ROS.

#### ABSTRACT

The use of unmanned aerial vehicle (UAV) today provides a new technological platform that can be used in remotely controlled operations. This work carried out the study, integration and testing of a prototype of an UAV, using different control platforms (Navio2, APM2.0, Erle-Brain) incorporated into multi-rotor prototypes, whose main purpose is to conduct searches in remote isolated areas. Throughout the of the project, there were stages of hardware assembling, calibration and software configurations, as well as incorporation of a thermal camera on the prototype. Mission simulations were performed in a scenario with similar concepts, where a VANT performs obstacles recognition and avoidance while inspecting an area. The physical part of the prototype is assembled and operational, and successful flight tests have been carried out. A ROS package was implemented to communicate the data obtained by the thermal camera. A tutorial was also developed regarding the assembly and installation of the prototype and the package.

**KEYWORDS:** UAV. Thermal Camera. ROS.

**Recebido:** 04 out. 2018.

**Aprovado:** 21 jan. 2019.

#### Direito autoral:

Este trabalho está licenciado sob os termos da Licença Creative Commons-Atribuição 4.0 Internacional.



## INTRODUÇÃO

Desastres naturais, como inundações, incêndios e terremotos, são acontecimentos muitas vezes difíceis de prever ou prevenir, e quando ocorrem trazem perigo a vida das inúmeras pessoas que estejam nas áreas afetadas. Vítimas desses desastres podem terminar feridas e/ou em situações de isolamento, em que equipes de busca devem agir rápido para conseguir resgatá-las com vida. Porém, nem sempre o processo de busca é bem-sucedido, mesmo quando feito por profissionais capacitados pode trazer riscos a eles quando em áreas de difícil acesso.

Veículos Aéreos Não-tripulados (VANT) são, como o nome diz, aeronaves que operam sem tripulação, podendo se mover livremente nos três eixos dimensionais. Esse tipo de veículo pode ser adquirido a um preço acessível, e devido a suas dimensões reduzidas, manobrabilidade e facilidade para acoplar sensores, eles podem ser utilizados em diversas aplicações.

Neste contexto, este projeto visa desenvolver um protótipo de VANT para realização de buscas em áreas de isolamento remoto, através da montagem de um frame multirrotor, ao qual será acoplado uma câmera térmica. Tal protótipo poderia facilmente sobrevoar áreas de difícil acesso, facilitando a busca de pessoas através do sensor térmico. Isso facilitaria a busca de sobreviventes, e também permitiria o envio de recursos básicos de sobrevivência (como água e comida) até a chegada do resgate.

## MÉTODOS

O desenvolvimento deste projeto consistiu de diversas etapas relacionadas ao estudo dos componentes eletrônicos e elementos de software necessários para a integração, montagem do hardware, configurações de software e testes e debug de falhas, bem como a parte de simulação de sistemas robóticos autônomos.

## FUNDAMENTAÇÃO TEÓRICA

A etapa de estudo consistiu em um levantamento de referências para adquirir conhecimentos sobre a fundamentação teórica necessária para o desenvolvimento do projeto. Alguns dos elementos estudados são o framework Ardupilot, o protocolo MavLink, o framework ROS[4] e os microcontroladores Erle-Brain[1], Navio2, e APM2.6. A configuração final do protótipo objetivada neste projeto é dada pelo diagrama da Figura 1.

## ESTRUTURA FÍSICA

Os equipamentos físicos essenciais utilizados durante o desenvolvimento deste projeto foram:

- a) 1xPlaca Controladora (Erle-Brain/Navio2/APM2.6);



- b) 1xFrame (Quadricóptero/Hexacóptero);
- c) 1xBateria;
- d) 1xCompasso/GPS;
- e) 1xRádio Transmissor Turnigy;
- f) 1xRádio Receptor D4R-II;
- g) 2xCartão MicroSD;
- h) 4x (6x) ESC;
- i) 4x (6x) Motor;
- j) 4x (6x) Hélice;
- k) 1xControlador de potência;
- l) 1xRaspberryPi 2B;
- m) 1xCâmera FLIR Lepton;
- n) Parafusos, cabos e conectores.

Durante a etapa de manutenção do hardware, o protótipo foi por diversas vezes montado e desmontado, de forma a realizar trocas da placa controladora, adição ou remoção de componentes, reparos depois de eventuais quedas, ou até a troca da estrutura do frame inteira. Foram testados dois frames do tipo hexacóptero e dois frames do tipo quadricóptero. Em todos eles foi realizado ao menos um voo manual, e em alguns casos também o voo autônomo.

### INTEGRAÇÃO DE COMPONENTES

Os sensores do drone, incluindo GPS, barômetro e acelerômetro, devem ser calibrados com o auxílio de uma base de controle, como o MissionPlanner, QGroundControl, Tower ou Mavproxy, sendo que todas estas foram testadas neste projeto. O rádio e os ESCs também precisaram passar por uma rotina de calibração.

A câmera térmica usada neste projeto é uma FLIR Lepton[2] com resolução 80x60. Ela foi conectada a uma Raspberry Pi, que se conecta à placa controladora via Wi-Fi. Foi desenvolvido um pacote ROS que é responsável pela captura e transmissão de imagens adquiridas pela câmera térmica.

Outros mecanismos testados durante o desenvolvimento deste projeto incluem uma garra para o transporte de objetos, um GPS RTK de alta precisão, transmissor de vídeo e receptor FPV, entre outros.

### SIMULAÇÃO

Testes com VANTs podem ser um processo muito lentos e dispendiosos, pois quando um teste falha, existe grande chance de resultar em danos físicos ao protótipo, que obrigam a gastos de dinheiro para reposição de peças, e de tempo para refazer etapas de montagem e calibração adequadas. Por isso, outra questão que foi abordada foi o uso de simulações para o desenvolvimento e

testes de plataformas robóticas. Foi então realizada uma simulação com o uso do software V-REP[3] onde um modelo quadricóptero precisaria percorrer um trajeto pré-determinado enquanto realiza a inspeção de uma rede de transmissão, desviando de obstáculos que porventura estiverem no caminho.

## RESULTADOS

Os drones montados conseguiram realizar voos com sucesso em diversas configurações de montagem. Nas figuras 1 e 2 é possível ver dois dos modelos montados no decorrer do projeto.

Figura 1: Frame Quadricóptero montado com placa Navio2



Figura 2: Frame Hexacóptero montado com placa Erle-Brain



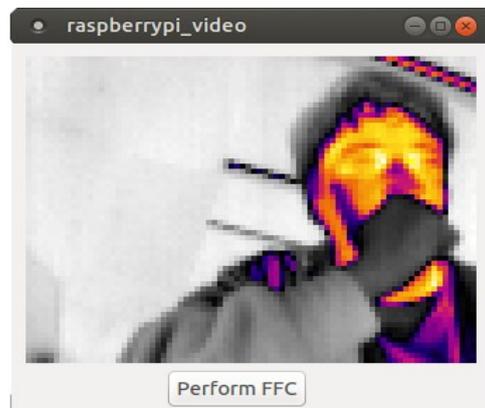
Em relação as placas controladoras, a APM 2.6 foi a única que conseguiu executar tanto o voo manual quanto o voo autônomo via MAVLink com sucesso. A Navio 2 executou com sucesso o voo manual, mas em todos os testes de voo autônomo resultaram em quedas no processo de decolagem. Por fim, o Erle-Brain realizou voos manuais com sucesso, mas devido aos recorrentes problemas encontrados devido a sua instabilidade, como o frequente corrompimento de seus cartões SD, acabou não sendo testado com voos autônomos. Os testes autônomos foram executados usando a base de controle MAVProxy, como um passo antes de migrar para o MAVROS. Embora tenha sido realizado o estudo e até o início da preparação de um script para missões via MAVROS, esta base de controle nunca chegou a ser testada em voo, pois a única placa que decolou corretamente via MAVLink é também a única das três que não possui integração com MAVROS.

A imagem da câmera térmica é adquirida e publicada corretamente, e foi desenvolvido um tutorial referente a montagem dela e instalação do pacote ROS para sua execução.

Na parte de simulação, foi desenvolvido um cenário de testes, similar a uma rede de transmissão de média ou baixa tensão, que um VANT deveria percorrer enquanto realiza uma inspeção na mesma, que pode ser visto na figura 4. Isso implica que o VANT tem que realizar um trajeto, enquanto se mantém em uma altura e distâncias corretas em relação às linhas de transmissão, e com a capacidade de evitar obstáculos inesperados. Para isso, foi adicionado ao drone simulado dois sensores de distância: o ultrassom e o Hokuyo. O sensor ultrassônico ficou responsável por medir a distância em relação às linhas de transmissão, enquanto o Hokuyo ficava voltado para a direção para a qual o drone estava se movendo, na busca por obstáculos dos quais o drone precisaria

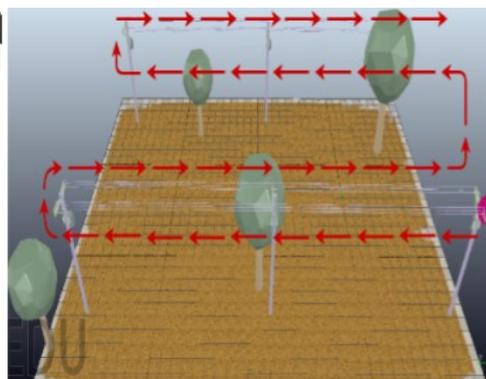
desviar. No caso de obstáculos, o drone busca subir pra desviar dos mesmos, e retorna a altura original usando o barômetro interno. O GPS também é usado para definir a trajetória. Foi desenvolvida então uma lógica fuzzy, que recebia como entrada os valores obtidos por esses quatro sensores, e como saída fornecia o movimento lineares nos eixos x, y e z, e movimento angular no eixo z (yaw). No fim, o drone simulado conseguiu, embora lentamente, realizar o trajeto desviando de obstáculos no caminho e mantendo uma distância adequada para as linhas de transmissão.

Fig. 3. Imagem câmera térmica.



Fonte: Autoria Própria (2018)

Fig. 4. Cenário de simulação.



Fonte: Autoria Própria (2018)

## CONCLUSÕES

Diversas configurações de protótipos puderam ser testadas, sendo que entre as placas controladoras todas conseguiram realizar voos, mas a Erle-Brain apresentou instabilidades no seu comportamento. As outras placas, a APM e a Navio2, se mostraram estáveis nos testes feitos com controle manual via rádio. Nos testes realizados via MAVLink, somente a APM levantou voo corretamente, pois a Navio2 não conseguiu fazer a decolagem de maneira estável, resultando em quedas do protótipo. As simulações forneceram boas noções de como o drone se comportaria quando em uma missão autônoma, visto que não foi possível chegar nessa etapa com os voos no protótipo físico.

Portanto, as atividades futuras para continuar este projeto incluem realizar mais testes de voo com autônomos com o protótipo, de forma a identificar a fonte de instabilidade, instalar um gimbal para maior mobilidade da câmera e elaboração de scripts para testes de missões autônomas, além de implementação de melhorias nas funcionalidades da estação base com relação a forma com que as informações são exibidas.

## REFERÊNCIAS

- [1] Erle Robotics. **Erle Robotics: Erle Brain**. Disponível em: <https://lsi.vc.ehu.eus/pablogn/docencia/manuales/Erle/erle-robotics-erle-brain-a-linux-brain-for-drones-en.pdf>. Acessado em: 08/2017.



[2] FLIR. **FLIR LEPTON® with Radiometry Datasheet**. Disponível em: <http://www.flir.com/uploadedFiles/OEM/Products/LWIR-Cameras/Lepton/Lepton%20Engineering%20Datasheet%20-%20with%20Radiometry.pdf> Acessado em: 08/2018.

[3] E. Rohmer, S. P. Singh, and M. Freese, **V-rep: A versatile and scalable robot simulation framework**, in Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on, pp. 1321–1326, IEEE, 2013.

[4] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, **Ros: an open-source robot operating system**, in ICRA workshop on open source software , vol. 3, p. 5, Kobe, Japan, 2009.

### AGRADECIMENTOS

Agradecemos ao apoio financeiro do CNPq, ao apoio de todos os membros do laboratório LESC e a orientação Prof. Dr. Marco A. Wehrmeister.