



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE CIÊNCIAS EXATAS E DA TERRA
DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO
MESTRADO EM SISTEMAS E COMPUTAÇÃO



Veículos Aéreos Não Tripulados e Sistema de Entrega: Estudo, Desenvolvimento e Testes

Manoel Pedro de Medeiros Neto

Natal-RN
Fevereiro de 2016

Manoel Pedro de Medeiros Neto

Veículos Aéreos Não Tripulados e Sistema de Entrega: Estudo, Desenvolvimento e Testes

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Sistemas e Computação do Departamento de Informática e Matemática Aplicada da Universidade Federal do Rio Grande do Norte como requisito final para a obtenção do grau de Mestre em Sistemas e Computação.

Linha de pesquisa:
Engenharia de Software

Orientador
Prof. Dr. Leonardo Cunha de Miranda

PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO (PPGSC)
DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA (DIMAP)
CENTRO DE CIÊNCIAS EXATAS E DA TERRA (CCET)
UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE (UFRN)

Natal-RN
Fevereiro de 2016

Catálogo da Publicação na Fonte
Universidade Federal do Rio Grande do Norte - UFRN
Sistema de Bibliotecas - SISBI

Medeiros Neto, Manoel Pedro de.

Veículos aéreos não tripulados e sistema de entrega: estudo, desenvolvimento e testes / Manoel Pedro de Medeiros Neto. - Natal, RN, 2016.

103f: il.

Orientador: Prof. Dr. Leonardo Cunha de Miranda.

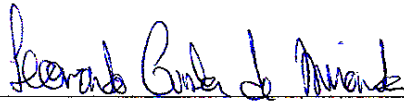
Dissertação (Mestrado) - Universidade Federal do Rio Grande do Norte. Centro de Ciências Exatas e da Terra. Programa de Pós-Graduação em Sistemas e Computação.

1. Engenharia de software. 2. Veículo não tripulado. 3. Veículo aéreo não tripulado. 4. Sistema de entrega. 5. Drone. 6. Quadricóptero. 7. Multirotor. I. Miranda, Leonardo Cunha de. II. Título.

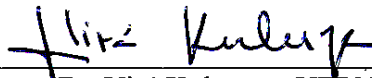
MANOEL PEDRO DE MEDEIROS NETO

Veículos Aéreos Não Tripulados e Sistema de Entrega: Estudo, Desenvolvimento e Testes

Esta Dissertação foi julgada adequada para a obtenção do título de mestre em Sistemas e Computação e aprovado em sua forma final pelo Programa de Pós-Graduação em Sistemas e Computação do Departamento de Informática e Matemática Aplicada da Universidade Federal do Rio Grande do Norte.

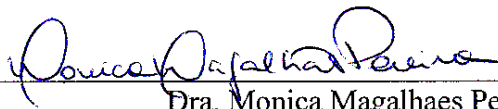


Dr. Leonardo Cunha de Miranda – UFRN
(Presidente)

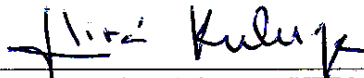


Dr. Uirá Kulesza – UFRN
(Coordenador do Programa)

Banca Examinadora



Dra. Monica Magalhaes Pereira – UFRN
(Examinadora)



Dr. Uirá Kulesza – UFRN
(Examinador)



Dr. Eduardo Brailio Wanderley Netto – IFRN
(Examinador)

Fevereiro, 2016

Dedico este trabalho à minha esposa Christiane e nossa filha Ana Luísa.

Agradecimentos

Agradeço à minha esposa Christiane pelo apoio incondicional em todos os momentos, sempre com palavras de apoio nos momentos de maior dificuldade, e à nossa filha Ana Luísa, com quem passei a dedicar menos tempo após iniciar este trabalho. Sem a presença de vocês em minha vida nada disso seria possível.

Agradeço aos meus Pais, Francisco Vicente de Medeiros e Ana Anita da Silva Medeiros, que mesmo com as dificuldades pelas quais passamos, trataram nossos estudos como prioridade, e aos meus irmãos Inácio e Ananília pelo crédito e incentivo demonstrados durante todos estes anos.

Agradeço ao meu orientador, o professor Leonardo Cunha de Miranda, por dedicar parte substancial de sua vida à educação e pesquisa, sempre acompanhando de perto o desenvolvimento de seus orientandos. Pela sua preocupação em aliar a pesquisa científica a algo prazeroso e por nos fazer acreditar que, com o planejamento adequado e esforço, mesmo as metas mais difíceis podem ser alcançadas.

Agradeço a todos que fazem parte do Physical Artifacts of Interaction Research Group – PAIRG – a distância não permitiu que eu interagisse com todos da forma como eu gostaria, mas tenho por vocês muito respeito e admiração, pois, ver pessoas tão jovens com tanta dedicação aos estudos me faz acreditar em um futuro melhor. Agradeço à professora Érica, pelas palavras de incentivo sempre nos momentos mais oportunos, meu muito obrigado.

Agradeço aos amigos Gilson e Sali e aos colegas de trabalho André Pinheiro, Daulton Rufino, Fábio Félix, Luciano Oseas, Marcos Vinícius, Maria Luísa e Túlio Miguel, pela valiosa contribuição durante todos os testes de voo. Agradecimento especial ao amigo André Luís, por abrir mão de seu tempo para ajudar no momento mais tenso da pesquisa, sem a sua colaboração parte dos voos não seria possível.

Agradeço a todos que fazem parte do Departamento de Informática e Matemática Aplicada – DIMAp – pela oportunidade de cursar minha pós-graduação em seu programa.

Agradeço aos membros da banca avaliadora, professor Eduardo Bráulio, professor Uirá Kulesza e professora Mônica Magalhães, por aceitarem o convite de participar desta banca de defesa e contribuir com este trabalho. Agradeço também ao professor Uirá Kulesza e à professora Mônica Magalhães pela valiosa contribuição na banca de qualificação, suas observações foram muito importantes para a conclusão deste trabalho.

"Inventar é imaginar o que ninguém pensou; é acreditar no que ninguém jurou; é arriscar o que ninguém ousou; é realizar o que ninguém tentou. Inventar é transcender."

(Alberto Santos Dumont)

Veículos Aéreos Não Tripulados e Sistema de Entrega: Estudo, Desenvolvimento e Testes

Autor: Manoel Pedro de Medeiros Neto

Orientador: Prof. Dr. Leonardo Cunha de Miranda

RESUMO

Veículos não tripulados estão cada vez mais presentes no cotidiano das empresas e das pessoas, pois esse tipo de veículo está de forma crescente desempenhando atividades que anteriormente eram apenas executadas por seres humanos. No entanto, para se compreender melhor o potencial de veículos não tripulados, é importante conhecer seus tipos, características, aplicações, limitações e desafios, pois somente com esse conhecimento pode-se entender as potencialidades do uso de veículos dessa natureza em aplicações variadas. Nesse contexto, na primeira parte desta pesquisa foram estudados os diferentes tipos de veículos não tripulados, i.e. terrestres, aquáticos, aéreos e híbridos. Durante a segunda fase da pesquisa, foi realizado um aprofundamento tendo como foco as interfaces de usuário para controle dos veículos aéreos não tripulados. Esses dois levantamentos iniciais do domínio, permitiram a identificação de desafios e oportunidades para o desenvolvimento de novas aplicações para esse contexto. Com base no conhecimento adquirido com esses estudos, então, foi desenvolvido um sistema de entrega automatizada de objetos para o campus de Universidades, denominado de PostDrone University, e desenvolvido um veículo aéreo não tripulado para realizar as entregas, denominado de PostDrone University UAV K-263. O sistema possui uma interface de usuário de fácil uso, que não requer conhecimentos de domínios específicos como aviação ou controle de aeronaves para sua operação. Por fim, diversos testes foram realizados com o intuito de validar e identificar as limitações da solução desenvolvida nesta pesquisa.

Palavras-chave: veículo não tripulado; veículo aéreo não tripulado; sistema de entrega, drone, quadricóptero, multirotor.

Unmanned Aerial Vehicles and Delivery System: Study, Development and Testing

Author: Manoel Pedro de Medeiros Neto

Advisor: Prof. Dr. Leonardo Cunha de Miranda

ABSTRACT

Unmanned vehicles are increasingly present in the daily of companies and people, because this kind of vehicle is performing ever more tasks that were previously only executed by human beings. However, to better understand the potential of unmanned vehicles, it is important to know their types, features, applications, limitations and challenges, thus with this knowledge it is possible to comprehend the possibilities of use of these vehicles in several applications. In this context, the first step of the present research consists in studying the different kinds of unmanned vehicles, i.e., ground, surface and underwater, aerial, and hybrid. During the second step of the research, a deepening study was accomplished, with focus on user interfaces of unmanned aerial vehicles. These two initial reviews of the domain allowed the identification of challenges and opportunities to the development of new applications for this context. Based on the acquired knowledge from these studies, then, an automated goods delivery system was developed for universities' campuses, called PostDrone University, and an unmanned vehicle to make the deliveries, called PostDrone University UAV K-263, was also developed. The system has an easy use UI, which does not require the user to have knowledge about specific domains, as aviation or aircraft control, for the operation of the system. Lastly, several test were accomplished aiming to validate the solution proposed in the present research and identify its limitations

Keywords: unmanned vehicles; unmanned aerial vehicle; delivery system, drone, quadcopter, multirotor.

Lista de Figuras

Figura 1. Veículos aquáticos não tripulados (a) AMSV monocasco (b) AMSV catamarã (c) AMUV com tanque de lastro (d) AMUV com motores elétricos. Fonte: [2][3][4][5].	22
Figura 2. Veículos terrestres não tripulados (a) UGV militar com tração em quatro pneus (b) UGV com tração em seis pneus (c) UGV militar com tração por esteiras (d) UGV de resgate com tração por esteiras (e) Robô com tração por patas. Fonte: [6][7][8][9][10].	23
Figura 3. Veículos aéreos não tripulados (a) Dirigível (b) Avião asa-alta (c) Helicóptero (d) Quadricóptero. Fonte: [11][12][13][14].	24
Figura 4. Veículos híbridos não tripulados (a) ACM-R5 (b) Hovercraft (c) HyTAQ Robot (d) Hexapode/Hexacooper. Fonte: [15][16].	24
Figura 5. Tipos de UAVs: (a) avião asa alta (b) Zagi (c) Canard (d) Helicóptero (e) Quadricóptero (f) Dirigível. Tipos de UAVs: (a) avião asa alta (b) Zagi (c) Canard (d) Helicóptero (e) Quadricóptero (f) Dirigível.	41
Figura 6. Exemplo de GUI de uma GCS. Fonte: [56].	41
Figura 7. Principais elementos de uma GUI de uma GCS (a) velocidade do ar, (b) horizonte simulado (c) altímetro (d) bússola e (e) variômetro.	42
Figura 8. Uso de interface BMI [71].	44
Figura 9. Gestos para controle de um UAV [74].	46
Figura 10. Interfaces utilizadas por [77] – a) Rádio transmissor; b) <i>Joystick</i> e c) IMU.	47
Figura 11. Interfaces táteis utilizadas por: a) [14] e b) [80][82].	48
Figura 12. PostDrone University UAV K-263.	60
Figura 13. Diagrama de blocos do K-263.	60
Figura 14. Componentes do PostDrone University UAV K-263 (a) frame (b) motor/hélice (c) módulo de força (d) micro-servo (e) ESC (f) GPS (g) kit de telemetria (h) bateria (i) distribuidor de força (j) HKPilot Mega.	61
Figura 15. Arquitetura do PostDrone University.	64
Figura 16. Diagrama de Estados das requisições de entrega do sistema.	65
Figura 17. Exemplo de arquivo de <i>waypoints</i> .	65
Figura 18. Diagrama de Atividades do gerenciamento de entregas do sistema.	66
Figura 19. Diagrama de Casos de Uso do sistema.	67
Figura 20. Exemplo da altitude mínima de segurança entre helipontos.	68
Figura 21. Tela do Aplicativo: a) cadastro de usuário; b) login e c) mural de notificações.	69
Figura 22. Diagrama de Atividades do cadastramento de nova requisição do sistema.	70

Figura 23. Testes de voo do UAV K-263 (a) modo de voo Loitter (b) take-off (c) pouso autônomo (d) <i>delivery box</i> acoplada com carga.	72
Figura 24. Passos para executar uma entrega.....	80
Figura 25. Altitude do <i>drone</i> durante a execução de uma missão autônoma.....	81
Figura 26. Detalhes dos testes de entrega automatizadas. a) Helipontos; b) Base para Origem – 28 metros; c) Origem para Destino – 33 metros; d) Destino para Base – 43 metros.....	81
Figura 27. Gráfico de qualidade do sinal nos módulos de telemetria.	83
Figura 28. Tela da ferramenta administrativa em funcionamento.	83
Figura 29. Primeira versão do <i>drone</i>	84
Figura 30. a) Pouso autônomo no heliponto; b) pouso autônomo fora do heliponto.	85
Figura 31. <i>Drone</i> em missão autônoma, executando o comando Loiter_time sobre um dos helipontos.....	85
Figura 32. Discrepâncias entre os valores do barômetro e do GPS.....	87
Figura 33. Gráfico de velocidade vertical durante acidente com carga de 200g.....	88
Figura 34. Gráfico de velocidade horizontal do <i>drone</i> com variadas cargas.....	89
Figura 35. Gráfico de consumo de corrente com a maior carga transportada.	89

Lista de Tabelas

Tabela 1. Sensores que compõem as IMUs nos artigos lidos.....	30
Tabela 2. Resumo dos trabalhos apresentados neste capítulo.....	31
Tabela 3. UIs empregadas no controle de UAVs.....	50
Tabela 4. Dispositivos empregados nas GCSs.	51
Tabela 5. Testes executados.	78
Tabela 6. Comandos das Missões.	80
Tabela 7. Consumo do <i>drone</i> com diferentes cargas.....	87
Tabela 8. Resumo das seções de voos executadas.	90

Lista de Abreviações e Siglas

AMSV	Autonomous Marine Surface Vehicle
AMUV	Autonomous Marine Underwater Vehicle
APM	ArduPilot Mega
BD	Banco de Dados
BMI	Brain-Machine Interface
CDI	Capacitor Discharge Ignition
DGPS	Differential Global Positioning System
DTMF	Dual-Tone Multi-Frequency
EEG	Eletroencefalografia
EKF	Extended Kalman Filter
ESC	Eletronic Speed Control
FAA	Federal Aviation Administration
GCM	Google Cloud Messaging
GCS	Ground Control Station
GPS	Global Position System
GUI	Graphical User Interface
HCI	Human-Computer Interaction
HDOP	Horizontal Dilution of Precision
HIV	Human Immunodeficiency Virus
IEEE	Institute of Electrical and Electronic Engineers
ILS	Instrument Landing System
IMU	Inertial Measurement Unit
JSON	JavaScript Object Notation
LED	Light Emitting Diode
LiPO	Lithium Polymer
MAVLink	Micro Air Vehicle Link
REST	REpresentational State Transfer
RTL	Return To Launch
SMS	Short Message Service
SUV	Sport Utility Vehicle
UAV	Unmanned Aerial Vehicle
UCRI	University of Cincinnati Research Institute

UGV	Unmanned Ground Vehicle
UI	User Interface
UV	Unmanned Vehicle
VTOL	Vertical Take-Off and Landing

Sumário

Lista de Figuras	x
Lista de Tabelas	xii
1 Introdução	16
1.1 Contexto, problemática e motivação	17
1.2 Objetivos	17
1.3 Relevância da pesquisa	18
1.4 Estrutura da dissertação	19
2 Veículos Não Tripulados: Estado da Arte e Desafios Computacionais	20
2.1 Introdução.....	20
2.2 Veículos Não Tripulados	21
2.2.1 Arquitetura e Tecnologias Empregadas com UVs	25
2.3 Estado da Arte.....	25
2.3.1 Síntese.....	29
2.4 Desafios.....	32
2.4.1 Aquáticos	32
2.4.2 Terrestres	33
2.4.3 Aéreos.....	33
2.4.4 Desafios gerais	34
2.5 Discussão	36
2.6 Conclusão.....	37
2.6.1 Comentários finais.....	37
3 Interfaces do Usuário em Sistemas de Controle de Veículos Aéreos Não Tripulados: Estado da Arte, Desafios e Oportunidades	38
3.1 Introdução.....	38
3.2 Veículos Aéreos Não Tripulados.....	39
3.3 Estado da Arte de UI para Controle de UAVs.....	42
3.3.1 Asas Fixas	42
3.3.2 Asas Rotativas	45
3.3.3 Mais leves que o ar	49
3.3.4 Diferentes UAVs	49
3.3.5 Síntese.....	50
3.4 Desafios e Oportunidades de UI para controle de UAV.....	51
3.4.1 Desafios comuns aos UAVs	51
3.4.2 Desafios específicos dos UAVs de Asas Fixas.....	53

3.4.3	Desafios específicos dos UAVs de Asas Rotativas.....	53
3.4.4	Desafios específicos dos UAVs mais leves que o ar	54
3.5	Conclusão.....	54
3.5.1	Comentários finais.....	55
4	PostDrone University: Veículo Aéreo não Tripulado e Sistema de Entrega no Campus.....	56
4.1	Introdução.....	56
4.2	Trabalhos relacionados.....	57
4.3	PostDrone University.....	59
4.3.1	UAV.....	60
4.3.2	Sistema.....	63
4.3.3	Servidor do Sistema.....	64
4.3.4	Servidor de Comunicação MAVLink	66
4.3.5	Ferramenta de Administração.....	67
4.3.6	Aplicativo	69
4.4	Testes de Voo do UAV K-263	71
4.5	Discussão	73
4.6	Conclusão.....	74
4.6.1	Comentários finais.....	74
5	PostDrone University: Resultados Práticos	75
5.1	Introdução.....	75
5.2	PostDrone University.....	76
5.3	Metodologia	77
5.3.1	Testes de <i>Software</i>	78
5.3.2	Testes de Voo.....	79
5.3.3	Testes de Integração.....	82
5.4	Resultados	82
5.5	Desafios e Discussão.....	90
5.6	Conclusão.....	92
5.6.1	Comentários finais.....	93
6	Considerações Finais.....	94
6.1	Trabalho realizado	94
6.2	Resultados	94
6.3	Perspectivas e Trabalhos Futuros.....	96
	Referências Bibliográficas.....	97

Capítulo 1

Introdução

A execução de determinadas atividades pode apresentar características que inviabilizam a presença humana no ambiente em que são executadas, seja pelos riscos inerentes à atividade ou pelo elevado custo em contratar pessoas para executá-las. Os Veículos não Tripulados – Unmanned Vehicles (UV) – surgiram para suprir essa demanda, apresentando vantagens como: baixo custo de produção e operação, maior versatilidade e maior período de operação sem interrupções. São equipamentos que podem ser operados remotamente por um operador/piloto ou desenvolver sua atividade de modo autônomo caso possuam um sistema de controle embarcado. Os UVs estão presentes hoje nos mais diversos ramos e podem ser classificados de acordo com o ambiente de operação, ou seja, terrestres, aquáticos, aéreos e híbridos.

Devido à sua versatilidade, os Veículos Aéreos não Tripulados – Unmanned Aerial Vehicles (UAVs) – estão presentes na maioria das pesquisas sobre UVs, essa versatilidade possibilita uma maior variedade de aplicações. São classificados pelo método de sustentação utilizado, podendo ser: de asas fixas, de asas rotativas e mais leves que o ar (balões e dirigíveis).

O foco desta pesquisa é o desenvolvimento de uma solução – *hardware* e *software* – capaz de gerenciar e efetuar entregas automatizadas de encomendas no perímetro de um Campus universitário utilizando-se de um UAV de asas rotativas. A ferramenta utilizada pelos usuários para o registro e acompanhamento de suas entregas será um aplicativo instalado em dispositivo móvel – *smartphone* ou *tablet* – que possua uma interface simples e intuitiva, que não requer conhecimentos em aviação ou controle de aeronaves por parte dos usuários, enquanto um sistema de acompanhamento e controle poderá ser utilizado por um usuário administrador.

1.1 Contexto, problemática e motivação

Os UVs são equipamentos móveis em sua maioria semelhantes a veículos em tamanho real em menor escala e que durante muitos anos foram objeto de pesquisa exclusivamente em âmbito militar, uma vez que os custos de produção e operacionais eram muito elevados. Dentre os UVs existentes os que apresentam a maior gama de aplicações são os UAVs, porém, a complexidade para garantir o controle e o grande número de restrições comuns mesmo às aeronaves tripuladas inviabilizaram por muitos anos as pesquisas com UAVs. No entanto, devido aos avanços e barateamento de componentes microeletrônicos e dos sistemas de controle, profissionais de diversas áreas do conhecimento passaram a realizar pesquisas nesse domínio. Tais avanços permitem o desenvolvimento de novos UAVs com graus de automação que dão suporte a atividades ainda não exploradas e conseqüentemente o surgimento de novos desafios de *hardware* e *software*.

O levantamento do estado da arte dos UVs, forneceu subsídios para o entendimento do domínio e inferência das principais atividades e desafios existentes, foi importante para a escolha dos UAVs como o tipo de UV a ser estudado com maior profundidade. A escolha pelo desenvolvimento de uma solução utilizando UAVs levou à necessidade de um segundo levantamento do estado da arte tendo como objeto de estudos os UAVs, tal estudo foi essencial no aprendizado sobre os UAVs e suas particularidades, conhecimento imprescindível principalmente para a construção do UAV proposto neste trabalho, bem como para a definição da solução a ser desenvolvida.

1.2 Objetivos

Durante as pesquisas iniciais, observou-se que UVs do tipo terrestre e aquáticos são utilizados em sua maioria para aplicações militares ou competições entre estudantes. Como os UAVs apresentaram maior versatilidade de aplicações, e o objetivo principal desta pesquisa é o desenvolvimento de uma solução que possa ser utilizada em âmbito civil e que traga benefícios à sociedade, os UAVs passaram a ser o objeto de estudos. Decidiu-se pelo desenvolvimento de uma solução completa – *hardware* e *software* – que possa gerenciar e efetuar entregas de encomendas utilizando-se de um UAV de asas rotativas.

Para que o objetivo principal possa ser alcançado os seguintes objetivos específicos são necessários:

- Entender o funcionamento dos UVs e identificar dentre os tipos estudados qual apresenta maior potencial para aplicações futuras;

- Compreender os diferentes tipos de UAVs existentes e identificar qual dos desafios encontrados será objeto de estudos;
- Conhecer os desafios de Interfaces do Usuário – *User Interfaces* (UIs) – de controle e *feedback* utilizados no controle de UAVs;
- Construir e testar um Veículo Aéreo Não Tripulado que possa executar missões autônomas;
- Propor e desenvolver um sistema e *hardware* para controle de UAV;
- Executar testes de voo para validar a montagem do UAV;
- Executar testes de voo autônomo para validar o sistema de controle embarcado e seus sensores;
- Executar testes de comunicação entre o UAV e a Estação de Controle – **Ground Control Station** (GCS);
- Desenvolver os *softwares* necessários ao funcionamento da solução;
- Descrever e efetuar testes de usabilidade do sistema simulando as entregas;
- Descrever e efetuar testes de entregas com o UAV.

1.3 Relevância da pesquisa

Apesar de existirem produtos à venda praticamente prontos para uso, bastando ao usuário ligar e pilotar seu equipamento, tais soluções apresentam limitações, como a arquitetura fechada e número reduzido de comandos disponíveis, que garantem apenas o uso recreativo do produto. Mesmo soluções profissionais que podem receber as mais variadas tarefas ainda requerem a participação de uma equipe treinada para a montagem, configuração e uso do equipamento e dos sistemas. Estes produtos prontos possuem sistemas de controle de arquitetura fechada, não permitindo o estudo e alterações em seu código-fonte, por isso decidiu-se que a pesquisa estaria voltada a sistemas de controle de código-aberto e que o UAV seria montado a partir do levantamento sobre os itens que o compõem, tentando aliar confiabilidade com preço acessível.

Mesmo os sistemas de controle mais simples podem apresentar informações pouco comuns aos usuários que não possuam conhecimentos em aviação, por isso o aplicativo a ser desenvolvido para que os usuários possam cadastrar e acompanhar suas entregas deverá possuir apenas os comandos para tal, ou seja, o usuário não precisa possuir conhecimentos de aviação, controle ou segurança de voo para utilizar o sistema. Outro *software* deverá fazer o controle mais completo da solução, inclusive apresentando informações do UAV durante as entregas.

1.4 Estrutura da dissertação

Neste ponto, é descrito brevemente o conteúdo de cada capítulo apresentado a seguir:

- **Capítulo 2 – Veículos Não Tripulados: Estado da Arte e Desafios Computacionais:** este capítulo apresenta o levantamento do estado da arte de todos os tipos de veículos não tripulados. Por meio dele foi possível descobrir as características mais importantes de cada tipo de veículo, suas aplicações práticas e os desafios computacionais gerais e específicos;
- **Capítulo 3 – Interfaces do Usuário em Sistemas de Controle de Veículos Aéreos Não Tripulados: Estado da Arte, Desafios e Oportunidades:** este capítulo apresenta o estado da arte das interfaces do usuário no controle de veículos aéreos não tripulados.
- **Capítulo 4 – PostDrone University: Veículo Aéreo Não Tripulado e Sistema de Entrega Automatizado:** este capítulo apresenta os detalhes do desenvolvimento da solução denominada PostDrone University, da escolha e montagem do *drone* (veículo aéreo não tripulado de asas rotativas) e do desenvolvimento de quatro *softwares* (Servidor do Sistema, Servidor de Comunicação MAVLink, Ferramenta Administrativa e Aplicativo Móvel);
- **Capítulo 5 – PostDrone University: Resultados Práticos:** este capítulo apresenta a metodologia dos testes executados na solução (necessários para verificar o comportamento do sistema bem como garantir que a comunicação entre os componentes do sistema está funcionando), dos testes de voo feitos com o *drone* (necessários para verificar as características de voo controlado e de voo autônomo) e testes de integração (quando o sistema recebe cadastros de entregas, gerencia essas entregas e envia missões ao *drone* para que este execute a entrega).
- **Capítulo 6 – Conclusões:** este capítulo resume os resultados da pesquisa e suas contribuições.

Capítulo 2

Veículos Não Tripulados: Estado da Arte e Desafios Computacionais

2.1 Introdução

Os veículos não tripulados são equipamentos móveis autônomos ou operados remotamente, desenvolvidos para que cumpram missões onde a presença humana deva ser evitada, seja para reduzir custos de operação, minimizar riscos à saúde ou para garantir a vida humana. Os altos investimentos necessários às pesquisas neste domínio fizeram com que durante anos as aplicações, com veículos não tripulados fossem objeto de estudo apenas no âmbito militar ou em grandes corporações, contudo, a evolução em diversos setores, especialmente a microeletrônica e a computação, culminou na considerável redução nos custos de componentes eletrônicos e ferramentas de desenvolvimento para novos e mais eficientes sistemas de controle. Atualmente a utilização de veículos não tripulados está presente nos mais diversos ramos, seja no âmbito militar com veículos de espionagem, defesa ou ataque, ou no âmbito civil com veículos para localização e resgate ou que executem tarefas cotidianas, tais como, pintar paredes, capturar imagens aéreas, dirigir carros e controlar empilhadeiras.

Para acompanhar os avanços nas tecnologias de *hardware* – e.g., sensores, atuadores e tecnologias de comunicação –, os sistemas de controle de veículos não tripulados tiveram que evoluir, principalmente devido ao aumento nos dados repassados pelos sensores, que deverão ser tratados e transformados em informações úteis para o funcionamento do veículo ou para o cumprimento das missões determinadas a ele.

Soluções computacionais de baixo custo nesse contexto são desejáveis, independentemente da aplicação do veículo não tripulado. Setores importantes para a economia como a agricultura, podem ser beneficiados com alternativas que envolvam a utilização de veículos não tripulados de

baixo custo. Neste contexto este capítulo apresenta o estado da arte dos veículos não tripulados com foco em soluções de baixo custo. Ainda, visando agregar valor ao estudo, identificamos e analisamos diversos desafios de *hardware* e *software* acerca do tema.

Este capítulo está organizado da seguinte maneira: a Seção 2.2 apresenta os veículos não tripulados; a Seção 2.3 apresenta o estado da arte do domínio; a Seção 2.4 apresenta os desafios de *hardware* e *software* acerca do tema; a Seção 2.5 apresenta uma discussão dos desafios sob a ótica da computação; e a Seção 2.6 apresenta as considerações finais.

2.2 Veículos Não Tripulados

Acredita-se que o primeiro veículo não tripulado construído foi um pequeno barco desenvolvido pelo cientista Nikola Tesla no final do Século XVIII. A solução desenvolvida por Tesla, patenteada em 1898 [1], demonstrou o potencial da transmissão de dados via ondas de rádio, antecipando como seria o futuro das comunicações remotas no controle de UVs.

A primeira classificação que pode ser realizada entre os UVs está relacionada ao ambiente onde esses veículos serão operados, podendo ser: **aquático, terrestre, aéreo** ou **híbridos**.

Os veículos aquáticos não tripulados são divididos em duas categorias, i.e., veículos marinhos não tripulados de superfície – ***Autonomous Marine Surface Vehicles (AMSVs)*** e veículos marinhos não tripulados submersíveis – ***Autonomous Marine Underwater Vehicles (AMUVs)***.

Os AMSVs são veículos aquáticos que navegam na superfície da água, sendo os barcos convencionais monocasco (Figura 1a) e catamarãs (barco com dois cascos) (Figura 1b) os mais utilizados para este fim. Esses veículos podem ser movidos por moto-propulsão – elétrica ou à combustão interna – ou impulsionados pelos ventos assim como os veleiros. Os AMSVs possuem poucas restrições de operação, sendo as condições climáticas a principal.

Por sua vez, os AMUVs são veículos aquáticos que podem navegar na superfície da água ou submersos. O método de propulsão encontrado como padrão neste tipo de UV é por meio de motores elétricos. O principal fator para que seja possível ao AMUV submergir e emergir é a sua flutuabilidade, podendo ser positiva para emersão, negativa para submersão ou neutra para manter a profundidade estabilizada. Existem três técnicas para controle de submersão e emersão dos AMUVs:

- i) tanques de lastro, i.e., são tanques que podem ser preenchidos com água tornando a flutuabilidade negativa ou preenchidos com ar pressurizado expulsando a água e tornando a flutuabilidade positiva;
- ii) motores elétricos, i.e., esta técnica só é possível se o AMUV possuir flutuabilidade neutra, assim, motores elétricos girando hélices podem fazer com que o AMUV possa submergir ou emergir apenas alterando o sentido de rotação dos motores;

iii) superfícies de controle (Figura 1c), i.e., são superfícies móveis que, assim como nos aviões, controlam a inclinação, como na técnica anterior o AMUV deverá possuir flutuabilidade neutra. Alguns AMUVs podem utilizar mais de uma técnica, tal como, utilizar motores elétricos em conjunto com o tanque de lastro (Figura 1d), a Figura 1 apresenta dois AMSVs (Figura 1a e Figura 1b) e dois AMUVs (Figura 1c e Figura 1d).

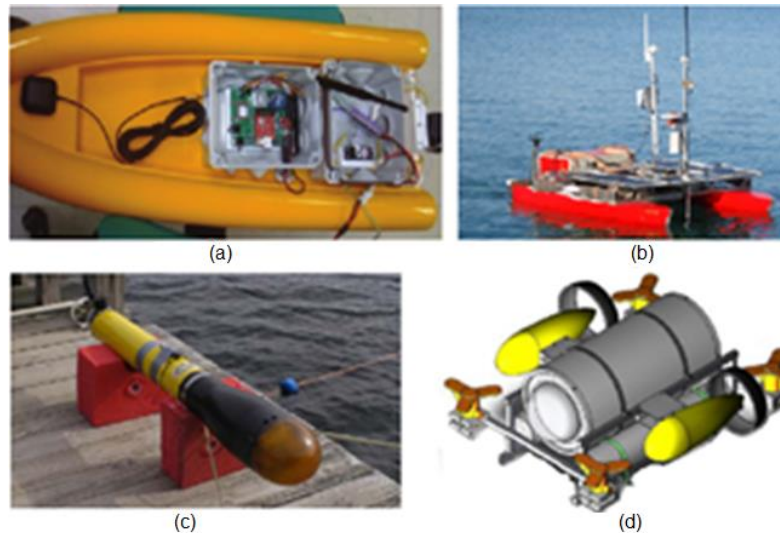


Figura 1. Veículos aquáticos não tripulados (a) AMSV monocasco (b) AMSV catamarã (c) AMUV com tanque de lastro (d) AMUV com motores elétricos. Fonte: [2][3][4][5].

Os Veículos Terrestres Não Tripulados – *Unmanned Ground Vehicles (UGVs)* possuem características que tornam o controle de navegação e posicionamento relativamente simples em comparação com outros UVs, uma vez que eles podem locomover-se apenas em dois sentidos, i.e., frente e trás, e esquerda e direita. Podem utilizar propulsão elétrica ou à combustão interna e normalmente não apresentam restrições quanto ao peso das baterias ou tamanho de seus motores.

A característica mais importante nos UGVs é o tipo de tração empregada, uma vez que essa tração deve ser adequada ao ambiente onde o UGV será utilizado, podendo ser:

- i) esteiras (Figura 2c e Figura 2d): é o tipo de tração mais indicado para ambientes onde o terreno seja irregular ou com lama, porém, a velocidade de locomoção é menor que em UGVs que utilizem outras técnicas;
- ii) pneus (Figura 2a e Figura 2b): UGVs com este tipo de tração apresentam maior velocidade de locomoção e podem ser utilizados em situações que requeiram menos barulho durante sua operação, a tração pode ser conseguida em um ou mais eixos ou independente em cada pneu;
- iii) patas (Figura 2e): alguns UGVs podem também apresentar semelhanças com animais, como quadrúpedes capazes de transportar cargas.

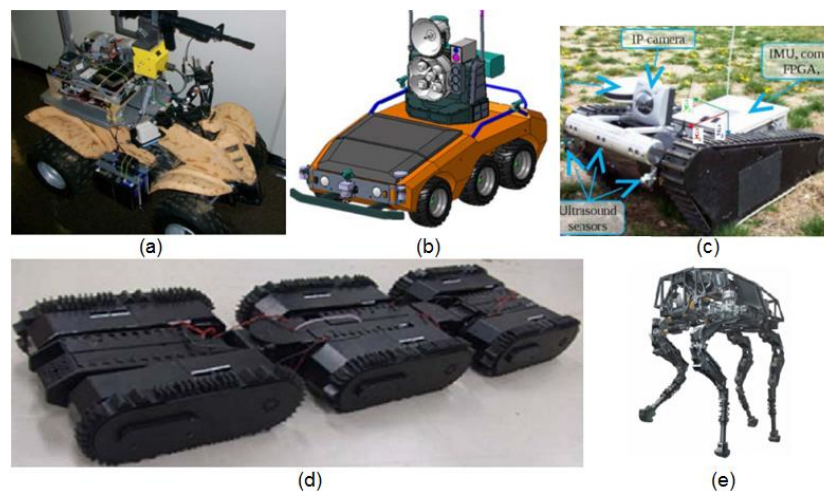


Figura 2. Veículos terrestres não tripulados (a) UGV militar com tração em quatro pneus (b) UGV com tração em seis pneus (c) UGV militar com tração por esteiras (d) UGV de resgate com tração por esteiras (e) Robô com tração por patas. Fonte: [6][7][8][9][10].

Os *drones* surgiram durante a segunda guerra mundial para fins militares e podem ser divididos em três categorias básicas: veículos de asas fixas (Figura 5b), veículos de asas rotativas (Figura 5c e Figura 5d) e mais leves que o ar (Figura 5a). Os UAVs de asa fixa são mais indicados para aplicações que requerem maior autonomia de voo, maior velocidade, maior altitude de operação e grandes deslocamentos, como em missões de vigilância de fronteiras. Algumas classificações estão relacionadas a:

- i) motorização, i.e., elétrico, combustão interna e turbina;
- ii) posição da asa, i.e., asa alta, asa baixa e canard.

UAVs de asas rotativas são mais versáteis e tem como características mais importantes as possibilidades de decolar e pousar verticalmente e de manter o voo pairado, podem ser divididos por:

- i) motorização, i.e., elétrico, combustão interna e turbina;
- ii) quantidade de conjuntos (motor/hélice), e.g., helicóptero, tricóptero e quadricóptero;
- iii) tipos de hélices, i.e., passo¹ fixo e passo variável.

¹ O passo nesse domínio é o ângulo de incidência das pás da hélice em relação à direção do movimento.

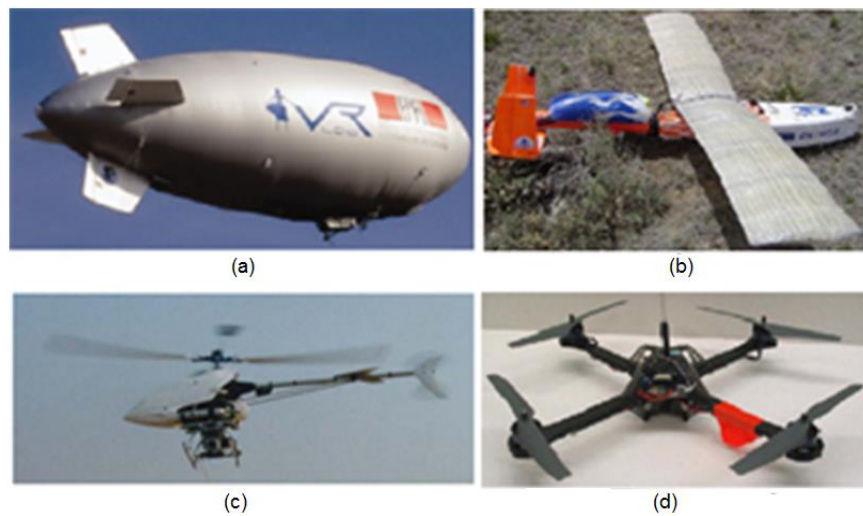


Figura 3. Veículos aéreos não tripulados (a) Dirigível (b) Avião asa-alta (c) Helicóptero (d) Quadricóptero. Fonte: [11][12][13][14].

Alguns UVs têm capacidade de locomover-se em diferentes ambientes – e.g., água, terra e ar – e, como não encontramos na literatura definições próprias para esse tipo de UV, denominaremos de **veículos híbridos não tripulados** [15] e [16]. Podem ser apresentados como veículos ou como pequenos robôs com algumas semelhanças com animais. A Figura 4 apresenta quatro diferentes veículos híbridos não tripulados.

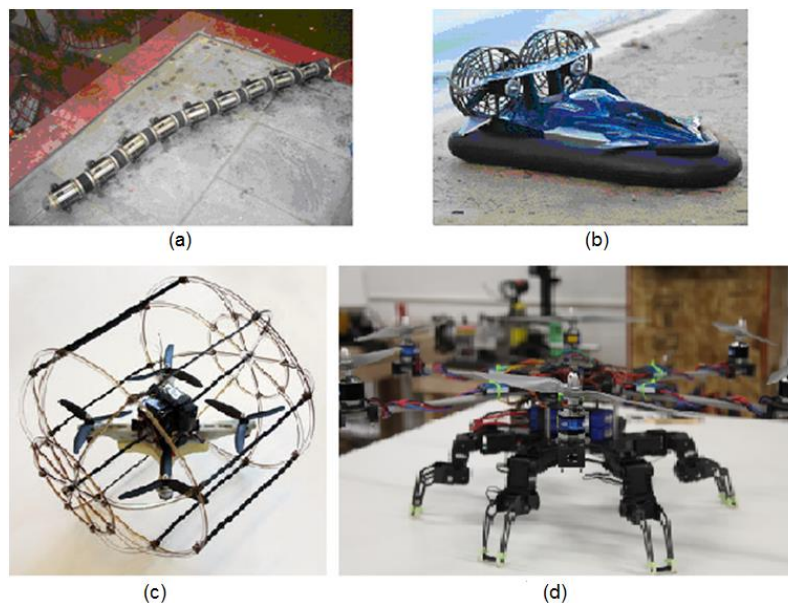


Figura 4. Veículos híbridos não tripulados (a) ACM-R5 (b) Hovercraft (c) HyTAQ Robot (d) Hexapode/Hexacopter. Fonte: [15]²[16]³.

² <http://www.skymall.com/images/products/DLA/20110912/204093034d.jpg>

³ http://madlabindustries.com/wp-content/gallery/hexapod-quadcopter/hexawalker_0336.jpg

2.2.1 Arquitetura e Tecnologias Empregadas com UVs

Diversas tecnologias de *hardware* e *software* em comum são aplicadas a praticamente todos os tipos de UVs – e.g., sensores, atuadores, interfaces de comunicação, microcontroladores, sistema operacional, ferramentas de desenvolvimento e protocolos de comunicação.

Especial atenção deve ser dada à GCS, ela é encarregada pela execução de aplicativos para o planejamento da missão, pelo recebimento de imagens e dados telemétricos dos sensores presentes nos UVs e apresentá-los em uma interface – geralmente gráfica – ao operador remoto. Também é possível controlar os UVs através de interfaces de controle instaladas na GCS, que pode ser composta por um conjunto de periféricos instalados em um ou mais computadores ou por equipamentos portáteis como *tablets* e *smartphones*.

Quanto às formas de controle dos UVs, eles podem ser:

- i) **autônomo**: é capaz de completar uma missão de acordo com o planejamento e de tomar decisões autonomamente caso surjam obstáculos ou riscos não previstos inicialmente;
- ii) **semi-autônomo**: pode completar uma missão a partir de um planejamento prévio, porém, caso surjam imprevistos ele não será capaz de tomar decisões autonomamente;
- iii) **controlado remotamente**: são controlados por um operador remoto. Os UVs controlados remotamente podem repassar os comandos recebidos diretamente aos seus atuadores, porém, alguns tipos de UVs necessitam que tais comandos sejam processados para em seguida enviar sinais de controle aos atuadores, podendo inclusive optar por não executar os comandos recebidos caso isso coloque o UV ou a missão em risco.

Uma plataforma de uso geral que tem sido bastante utilizada como controlador de UVs é a plataforma Arduino⁴, tanto que já existe um produto comercial preparado para o controle de UVs denominado Ardupilot Mega (APM)⁵, essa plataforma pode receber um conjunto de sistemas de código-livre denominado Ardupilot (ArduRover, ArduPlane e ArduCopter, para controle de carros, aviões e helicópteros e multicópteros).

2.3 Estado da Arte

Nesta seção apresentamos os trabalhos mais relevantes sobre UVs, escolhidos dentre artigos oriundos, principalmente, de periódicos disponíveis na IEEEExplore Digital Library e ACM Digital Library, organizados pelo tipo de UV, i. e., aquáticos, terrestres, aéreos e híbridos. Os termos pesquisados inicialmente estavam relacionados a UVs de baixo custo, entretanto, para um

⁴ <http://arduino.cc>

⁵ <http://diydrones.com/notes/ArduPilot>

melhor entendimento do domínio observou-se a necessidade de pesquisar UVs sem restringir a pesquisa apenas aos de baixo custo.

A maioria dos trabalhos desenvolvidos envolvendo **AMSVs** e **AMUVs** são de natureza militar, mais especificamente voltados para a localização e neutralização de minas explosivas. Esse tipo de atividade é importante para garantir a segurança da navegação em determinadas partes do mundo, onde a presença de minas explosivas ainda é uma realidade, como por exemplo o Mar Báltico, onde segundo Pereda et al. [2], existem mais de 100.000 minas ainda ativas em decorrência da Segunda Guerra Mundial. A solução apresentada em [2] (Figura 1a) consiste de uma embarcação monocasco que reboca um dispositivo emissor de campos elétricos e magnéticos capazes de detonar minas submersas, mesmo que estejam abaixo da lama ou no fundo do oceano. Com o mesmo intuito, um catamarã foi desenvolvido por Bruzzone et al. [3] (Figura 1b) para a identificação e localização de minas. Também para neutralização de minas, o AMUV semelhante a um torpedo proposto por Schulz et al. [4] (Figura 1c) que foi capaz de localizar e neutralizar minas.

Ainda no âmbito militar, Cadena [17] propõem um AMUV capaz de transportar e lançar pequenos UAVs armazenados dentro de cilindros, ou seja, uma plataforma móvel que permanece submersa até o momento adequado para o lançamento de pequenos UAVs de espionagem ou mesmo de ataque.

O AMUV desenvolvido por Wang et al. [5] (Figura 1d) foi utilizado como ferramenta para inspecionar as caixas de mar⁶ de navios, como essa atividade requer um veículo manobrável, a locomoção e inclinação são conseguidos por meio de motores enquanto a emersão e submersão é garantida por tanques de lastro.

Foram encontrados trabalhos envolvendo **UGVs** para os mais diversos propósitos, i.g., aplicações militares, participação em competições, missões de busca e resgate em ambientes hostis, como o UGV apresentado por Bhatia et al. [18], capaz de auxiliar na localização de vítimas em áreas de desastres ou o sistema autônomo responsável por expulsar aves do perímetro de aeroportos proposto por Kin et al. [7] (Figura 2b). Um UGV de uso militar proposto por Bouhraoua et al. [6] (Figura 2a), controlado remotamente, possui uma metralhadora instalada em uma plataforma giratória, permitindo que a mesma seja apontada para os alvos desejados mesmo com o UGV em movimento.

As competições também são importantes para o desenvolvimento de UVs, principalmente UGVs. Neste domínio Rawashdeh et al. [19] apresentaram um UGV adaptado no chassi de uma cadeira de rodas elétrica, capaz de percorrer um caminho demarcado por linhas brancas em um campo de futebol, enquanto Appelqvist et al. [20] prepararam um UGV de grande porte para

⁶ Caixas de Mar são entradas de água nos cascos dos navios que permitem que a água circule em determinadas áreas do mesmo, seja para refrigerar partes dos motores ou mesmo como garantia do fornecimento de água para o combate a incêndios.

competições, usando para isso um veículo utilitário esportivo (*Sport Utility Vehicle* – SUV), adaptado para executar missões autônomas.

Alguns UGVs fogem aos tradicionais modelos de veículos, como o UGV quadrúpede apresentado por Wooden et al. [10] (Figura 2e), que mede 3 metros de comprimento por 2,5 metros de altura, esse UGV é capaz de transportar pesadas cargas mesmo em ambientes irregulares ou escorregadios, ou o UGV criado por Miyanaka et al. [9] denominado KOHGA2 (Figura 2d), de pequena dimensão e semelhante a uma cobra. Esse UGV é capaz de entrar em locais onde outros UGVs não conseguem, seja pelas dimensões em relação ao ambiente ou pelas condições do terreno.

Devido à sua grande flexibilidade, os UAVs estão presentes na maioria das pesquisas encontradas, desde as que apresentam os detalhes técnicos e as principais características deste tipo de UV (Gebauer et al. [21], Gupte et al. [22], Noda et al. [23], Sa et al. [24], Mclean et al. [25]), importantes para o entendimento do domínio, até as que apresentam soluções para os mais diversos fins, como: aplicações militares que propõem a utilização de UAVs de ataque, Ruff et al. [26]; utilização de UAVs na execução de tarefas normalmente executadas por humanos como pintar paredes Albers et al. [27]; capturar imagens aéreas, Gademer et al. [28] e até mesmo a utilização de brinquedos como o pequeno helicóptero apresentado por Divakar [29], adaptado para seguir linhas coloridas em uma superfície branca.

O microcontrolador Arduino Duemilanove foi utilizado por Leong et al. [30] como o sistema de controle de voo e estabilidade de seu quadricóptero, os resultados mostraram que com baixo custo de equipamentos e de implementação, consegue-se desenvolver um controlador de estabilidade e voo capaz de manter o quadricóptero em voo pairado estável. A utilização de aeromodelos é comum em pesquisas neste domínio, como o aeromodelo de asas fixas do tipo treinador utilizado por Amahah [31] para percorrer pontos de interesse determinados pelo operador, a plataforma APM é responsável pelo controle de estabilidade e pelo voo autônomo. Ainda utilizando o Arduino, Zafar et al. [32] desenvolveram um sistema para o controle remoto de um UAV de asas fixas, o trabalho tem como principal foco a troca de informações telemétricas e de controle entre o UAV e sua GCS.

Devido ao custo reduzido, os recursos disponíveis e a facilidade de uso, o brinquedo comercializado pela empresa ARDrone, denominado Parrot, tem sido usado como base para alguns estudos. Originalmente esse brinquedo pode ser controlado por um aplicativo instalado em dispositivos móveis utilizando o Android ou o IOs. Este aplicativo além de permitir o controle do Parrot exibe as imagens capturadas pela câmera *onboard*. Duas pesquisas fazem uso das imagens transmitidas pelo UAV. Berezny et al. [33] implementaram um controlador de voo autônomo baseado em visão computacional, enquanto Ng et al. [34] propõem um sistema de reconhecimento de gestos para o controle remoto do seu Parrot, esse sistema armazena os principais gestos utilizados por falcoeiros e associa a uma ação a ser efetuada pelo UAV. O trabalho mais complexo utilizando este equipamento foi apresentado por Alvissalim et al. [35],

onde vários Parrot's com capacidade de voo autônomo executam tarefas de modo colaborativo, como reestabelecer as comunicações em ambientes onde a estrutura de comunicação tenha sido rompida.

Algumas tarefas necessitam que os UAVs executem missões colaborativas, onde os UAVs trabalham em equipe, como o time de pequenos quadricópteros capaz de executar missões de vigilância apresentado por Michael et al. [36] ou de restabelecer a comunicação em ambientes onde os canais de comunicação tenham sido destruídos como em Blum et al. [37] e Alvissalim et al. [35], outros autores Michael et al. [36], Rudol et al. [38] e Nathan et al. [39] também trataram da execução colaborativa de missões utilizando UVs. Devido ao seu grau de automação, as missões colaborativas dependem muito de um bom sistema de planejamento de missões, por isso a necessidade de pesquisa envolvendo algoritmos para o planejamento de missões, como as soluções apresentadas por de la Cruz et al. [40], Culligan et al. [41], Wu et al. [42] e Yang et al. [43], todos para o controle de missões colaborativas com uso de UAVs. O algoritmo desenvolvido em [42] para planejamento e execução de missões com grupos de UAVs autônomos, leva em conta as regras de voo da aviação civil e comercial da região em que os UAVs serão utilizados.

O controlador de voo e estabilidade MicroPilot's MP2028g foi encontrado em dois trabalhos, Rudol et al. [38] desenvolveram um sistema para que um UGV trabalhe em conjunto com um UAV, esse UGV é capaz de seguir e determinar a exata posição e sentido em que o UAV se encontra, graças a um conjunto de LED's coloridos instalados no UAV, enquanto Chandler et al. [12] desenvolveram um UAV de asas fixas para no futuro, executar uma missão em Marte.

Para garantir que um quadricóptero mantenha o voo pairado, Fowers et al. [44] desenvolveram um sistema de visão computacional capaz de compensar os movimentos involuntários do UAV, mesmo os mais lentos e imperceptíveis aos sensores inerciais. O sistema faz um mapeamento dos objetos que compõem as imagens capturadas pela câmera, e utiliza como referência de localização para que o UAV permaneça no ponto determinado. Um quadricóptero capaz de pintar pontos formando desenhos em paredes é proposto por Khosouhee et al. [45], para isso o UAV posiciona-se próximo a uma parede definida como a tela de pintura e nas posições onde a GCS indique, ele deve pintar pontos, o conjunto de pontos forma a imagem final.

Hing *et al.* em [46] e [47] apresentaram trabalhos com foco bem definido em interfaces, em [46] estudam as melhores alternativas para o controle de um multicóptero e em [47] medem a fadiga gerada ao operador/piloto durante o controle de um multicóptero.

Na aviação o pouso é um momento crítico onde muitos acidentes podem ocorrer, principalmente para aeronaves de asas fixas, tanto que alguns trabalhos estão focados em garantir a segurança do UAV neste momento, como apresentado por Saghafi et al. [48], Shaker et al. [49] e Das et al. [50]. O pouso de multicópteros em superfícies planas é relativamente simples de

executar, basta reduzir de forma lenta e gradual a velocidade dos motores do UAV, porém, em superfícies oscilantes esse método não é o mais indicado, por isso Das et al. [50] apresentaram um sistema de apoio a pouso de um quadricóptero em superfícies oscilantes, tal como o convés de um navio.

A autonomia é um requisito importante para qualquer tipo de UV, porém, para UAVs esse requisito é mais relevante, por isso a utilização de plataformas computacionais com baixo consumo de energia é vital para aumentar a autonomia do UAV. Para melhorar a performance neste quesito Murdoch et al. [51] apresentaram estudos visando aumentar a eficiência na captação de eletricidade por meio de painéis solares instalados em um UAV podendo assim aumentar a autonomia do mesmo.

Uma parcela pequena dos trabalhos encontrados apresenta soluções para UVs **híbridos**, alguns deles com características de animais, como o veículo em formato de cobra (Figura 4a) desenvolvido por Yu et al. [15], capaz de locomover-se em terra e na água. Entre os UVs híbridos o que apresenta maior desempenho quanto à locomoção em todos os ambientes onde consegue atuar é o *hovercraft* (veículo que tem como apoio um colchão de ar – Figura 4b), sua velocidade de locomoção é praticamente a mesma tanto em terra como na água, porém, o fato de não haver atrito entre o veículo e a superfície torna o controle do mesmo mais complexo, isso levou Cheng-long et al. [52] a desenvolverem um algoritmo capaz de melhorar o controle das manobras para este tipo de veículo. Um quadricóptero alojado em uma armação metálica em formato cilíndrico, que devido à sua aparência é denominada “gaiola de esquilo” (Figura 4c), foi apresentado por Kalantari et al. [16], o HyTAQ Robot (*Hybrid Terrestrial and Aerial Quadrotor*) é um veículo capaz de mover-se por terra ou mesmo levantar voo, atenuando os problemas mais desafiadores de dois tipos de UVs que são a baixa autonomia dos UAVs e a dificuldade para transpor obstáculos dos UGVs. Os resultados preliminares obtidos por [16] apontam que o UV híbrido foi capaz de percorrer uma distância de quatro a seis vezes maior que atuando apenas como UGV ou UAV. Outro trabalho voltado a uma solução para UVs híbridos que possam locomover-se em terra e também voar foi apresentado pela empresa MadLab Industries, onde a estrutura de um Hexacóptero foi agregada a um UGV de seis patas denominado Hexapode (Figura 4d).

2.3.1 Síntese

Os sensores que dão aos UVs os dados necessários à sua localização, posição, orientação e aceleração, são chamados sensores **proprioceptivos**, eles conseguem enviar dados que, ao serem tratados, geram informações que garantem o equilíbrio, localização e a correta movimentação dos UVs, estes sensores também são conhecidos como sensores de medição inercial. Com a miniaturização dos componentes eletrônicos, diversos destes sensores passaram a compor um único componente denominado de unidade de medição inercial – ***Inertial Measurement Unit***

(IMU) – essa unidade é responsável por determinar entre outras coisas a atitude⁷ do UV, sua aceleração ou mesmo sua posição em relação ao norte magnético, para isso, as IMUs podem conter um ou mais dos seguintes componentes: bússola digital, acelerômetro e giroscópio.

Além dos sensores mencionados, as IMUs costumam também utilizar dois outros componentes como os sensores de altitude (barômetro ou sonar) e o GPS. A maioria dos trabalhos lidos apresenta a utilização de dois ou mais sensores na IMU, a Tabela 1 apresenta os sensores utilizados nos trabalhos lidos.

Tabela 1. Sensores que compõem as IMUs nos artigos lidos.

Artigos	Acelerômetro	Bússola Digital	Giroscópio	GPS	Sensor Altitude
[5]	X				
[20][53][2][4][54]				X	
[3][7]		X		X	
[33][50][30][34][38][55]	X		X		
[56]			X	X	
[44]	X	X	X		
[19]	X	X		X	
[35][22][45]	X		X	X	
[17]	X	X	X	X	
[13]		X	X	X	X
[32]	X		X	X	X
[12][21][36][23][57]	X	X	X	X	X

Algumas características são comuns a todos ou quase todos os UVs, uma delas é a necessidade de comunicação, entre o UV e sua GCS ou entre os UVs em missões colaborativas. Observa-se que o padrão XBee está presente em muitas das pesquisas ([2][14][19][23][32][36][45]), os principais motivos para isso são: baixo consumo energético e imunidade contra interferências. A tecnologia IEEE 802.11 e suas variações também foram utilizadas para garantir a comunicação entre o UV e sua GCS ([3][4][6][20][33][34][35][37]). Outras técnicas foram apresentadas, como em [58] que utilizou os sinais DTMF (*Dual-Tone Multi-Frequency*) para enviar comandos a um UGV ou [12] que utilizaram dois pares de radioamadores para garantir a comunicação full-duplex entre a GCS e um UAV.

Além de utilizar IMUs, os UAVs necessitam de um sistema capaz de manter o controle e a estabilidade em voo, isso se deve ao maior grau de dificuldade no controle dos mesmos, principalmente os UAVs de asas rotativas. Dentre as tecnologias apresentadas para o controle de voo e estabilidade, as mais utilizadas são: (Astec Autopilot [14], Kestrel Autopilot [44] e MicroPilot's MP2025g [12][38]), a utilização da plataforma Arduino e suas variações foi a mais citada ([31][50][29][45][30][23]), esta plataforma consegue agregar características importantes como: **baixo custo, maior oferta de recursos, confiabilidade e projetos abertos (*hardware e software*)**.

⁷ Atitude é a posição em que o UV encontra-se em relação à linha do horizonte, podendo estar completamente alinhado à mesma ou inclinado em um ou mais dos seus eixos.

A Tabela 2 apresenta um resumo dos trabalhos apresentados neste capítulo, organizados de acordo com o ano de publicação, tipo de veículo, aplicação (Civil, Militar ou Não Declarado) e um pequeno resumo do que foi apresentado no capítulo.

Tabela 2. Resumo dos trabalhos apresentados neste capítulo.

Trabalho		Tipo de UV	Tipo de aplicação
Ano	Ref.		
1999	[26]	UAV	M - O autor modela um sistema de controle para UAVs de ataque.
2005	[12]	UAV	M - Uso em Marte.
	[13]	UAV	C - O autor investiga os três tipos de UAVs à procura da melhor solução que possa apoiar no combate à incêndios, sendo considerada a melhor escolha um UAV de asas rotativas.
	[4]	UMV	M - Localização e neutralização de minas.
2007	[44]	UAV	ND - Garantir o posicionamento por meio de imagens.
	[20]	UGV	M - Reconhecimento tático e vigilância.
	[9]	UGV	ND - Localização de vítimas de desastres.
2008	[41]	UAV	ND - Algoritmos para planejamento de missões.
	[38]	UAV	ND - Trabalho colaborativo entre UAV e UGV.
	[53]	UAV	C - Estudam a fadiga durante a operação de UAVs em missões de resgate em áreas de desastres.
	[5]	UMV	C - Limpeza de partes submersas de navios.
2009	[40]	UAV	ND - Algoritmos para planejamento de missões.
	[25]	UAV	ND - Apresenta diversos conceitos sobre multicópteros que servem como base para pesquisas do domínio.
	[31]	UAV	ND - Voo autônomo de um UAV de asa fixa.
	[17]	UMV	M - Transporte e lançamento de UAVs.
	[54]	UAV	C - Dirigível utilizado na obtenção de imagens aéreas que possam ser usadas na cartografia e topografia.
	[42]	UAV	ND - Algoritmos para planejamento de missões que leva em conta entre outras coisas, as regras de voo no ambiente em que será utilizado.
2010	[48]	UAV	ND - Algoritmos para pouso autônomo de UAVs de asas fixas.
	[28]	UAV	ND - Utilização de um quadricóptero para aquisição de imagens a baixa altitude.
	[47]	UAV	ND - O autor apresenta três interfaces de controle para avaliar qual seria a melhor no controle de um UAV de asas rotativas.
	[2]	UMV	M - Localização e neutralização de minas.
	[52]	Híbrido	ND - Melhorar o controle de direção em Hovercrafts.
	[6]	UGV	M - UGV de ataque.
	[10]	UGV	M - Robô quadrúpede capaz de transportar cargas.
2011	[43]	UAV	ND - Algoritmos para planejamento de missões.
	[49]	UAV	ND - Algoritmos para pouso autônomo de UAVs de asas rotativas.
	[27]	UAV	ND - UAV de asas rotativas capaz de interagir com o ambiente por meio de contato físico.
	[46]	UAV	ND - Estudo da fadiga durante a operação de UAVs de asas rotativas, utilizando fNIR.
	[29]	UAV	ND - Adaptar um pequeno helicóptero para voo autônomo.
	[34]	UAV	ND - Controle de UAV por gestos.
	[32]	UAV	ND - Apresenta uma solução de baixo custo para controle de UAVs.
	[3]	UMV	M - Localização e neutralização de minas.
2011	[15]	Híbrido	ND - UV híbrido em formato de cobra.
	[18]	UGV	C - Localização de vítimas de desastres.
	[7]	UGV	M - UGV capaz de expulsar aves de aeroportos.
	[36]	UAV	M - Trabalho colaborativo de UAVs para garantir a vigilância.
	[23]	UAV	ND - Apresenta diversos conceitos sobre multicópteros que servem como base para pesquisas do domínio.
	[57]	UAV	ND - O autor apresenta um controlador de voo para quadricóptero que utiliza um sistema operacional de tempo real com memória protegida.
	[58]	UGV	ND - Apresenta uma forma pouco explorada para o envio de comandos a um UGV, através dos tons DTMF (Dual-Tone Multi-Frequency) dos aparelhos telefônicos.
	[39]	UAV	ND - UAVs capazes de executar missões de modo colaborativo.
	[59]	UAV	ND - Utilização do sensor Microsoft Kinect para determinar a altitude e profundidade do ambiente.
	[56]	UAV	M - Os autores desenvolveram um sistema completo (GCS, controle de voo e estabilidade) para UAVs de asas fixas e asas rotativas.

2012	[35]	UAV	C - Uso de UAVs para restabelecer comunicação sem fios.
	[33]	UAV	ND - Controle de UAV por gestos
	[37]	UAV	C - Uso de UAVs para restabelecer comunicação sem fios.
	[50]	UAV	M - Auxiliar no pouso autônomo em superfícies oscilantes.
	[21]	UAV	ND - Apresenta diversos conceitos sobre multicópteros que servem como base para pesquisas do domínio.
	[22]	UAV	ND - Apresenta diversos conceitos sobre multicópteros que servem como base para pesquisas do domínio.
	[45]	UAV	C - Pintar paredes utilizando UAVs.
	[30]	UAV	C - Uso da plataforma Arduino no controle de UAVs.
	[19]	UGV	ND - UGV autônomo para competições.
2013	[24]	UAV	ND - Apresenta diversos conceitos sobre multicópteros que servem como base para pesquisas do domínio e o desenvolvimento de um sistema de controle de voo e estabilidade.
	[16]	Híbrido	ND - Veículo híbrido aéreo e terrestre.
	[60]	UAV	ND - Desenvolver um circuito MPPT (<i>Maximum Power Point Tracker</i>) para painéis solares instalados em um UAV.

2.4 Desafios

Os desafios existentes no desenvolvimento e operação de UVs podem variar de acordo com o tipo do UV ou sua aplicação, desta forma, serão apresentados os principais desafios de acordo com o tipo de UV.

2.4.1 Aquáticos

Os AMUVs são os UVs aquáticos que possuem maiores restrições e conseqüentemente maiores desafios, dentre os principais pode-se citar: o *hardware* deve **resistir à pressão externa** a que o AMUV é exposto durante sua operação; a determinação da **localização** do AMUV por meio de GPS só é conseguida quando o mesmo está acima da linha d'água, o que exige que o mesmo retorne à superfície regularmente para determinar sua localização para possíveis correções de trajetória; dificuldade em garantir a **comunicação** entre a GCS e o AMUV enquanto o mesmo está submerso. Quanto à precisão na localização de AMUVs que não navegam a grandes profundidades, pode-se utilizar uma antena retrátil para que o GPS possa ser enviado à superfície da água em tempos regulares, garantindo o correto posicionamento sem que o AMUV seja exposto.

Para cumprir missões de localização e neutralização de minas aquáticas, os sistemas de planejamento utilizados determinam o percurso que o AMUV ou AMUV deve percorrer, porém, nenhum dos trabalhos deixa claro que alternativas são utilizadas para compensar as variações que ocorrem nas correntes que podem alterar a trajetória do UV, principalmente para aqueles trabalhos onde os UVs rebocam equipamentos responsáveis pela localização ou neutralização de minas, nestes casos o UV pode seguir corretamente seu trajeto enquanto o dispositivo rebocado está distante do lugar esperado. Pode-se estimar a correta posição do equipamento rebocado efetuando cálculos usando como base a posição do UV e informações sobre o cabo de reboque

como: distância entre o UV e o dispositivo rebocado, direção apontada pelo cabo e seu grau de inclinação.

Alguns trabalhos apresentados que tem como objetivo a localização e possivelmente neutralização de minas, necessitam que um operador execute uma **análise da imagem capturada** e decida se o objeto encontrado é ou não uma mina, essa análise poderia ser executada por sensores capazes de detectar elementos comuns às minas e também por sistemas capazes de analisar as imagens capturadas, executando um filtro de borda que defina o formato do objeto para posterior busca em um banco de dados com formatos comuns das minas.

2.4.2 Terrestres

Aplicações que façam uso de UGVs completamente autônomos lidam com um problema comum, o replanejamento de rota, uma vez que apenas os sensores de obstáculo em conjunto com o GPS não são capazes de prover soluções para este problema. As alternativas estudadas envolvem a aquisição, tratamento e processamento das imagens obtidas por câmeras instaladas no UGV, estas tarefas devem ser executadas pelo equipamento de processamento *onboard* no UGV. Dependendo do tamanho do UGV, a instalação de tal equipamento não representa um problema, uma vez que as restrições quanto ao volume, peso ou consumo elétrico são menores para este tipo de UV.

O ambiente onde o UGV é operado interferirá diretamente no tipo de comunicação utilizado, podendo ocorrer desde o bloqueio na transmissão dos dados, devido aos vários obstáculos existentes, até interferências causadas por outras tecnologias. A segurança na transmissão dos dados entre o UGV e a GCS deve receber especial atenção do desenvolvedor, principalmente os dados relacionados ao controle do UGV. Mesmo os pesquisadores que participam de projetos desenvolvidos com interesses unicamente civis, devem ser criteriosos nos cuidados relacionados à segurança das informações transmitidas.

2.4.3 Aéreos

UAVs de asas fixas necessitam de grandes áreas planas para decolagens e pousos, e o tamanho para estas áreas está diretamente relacionado ao tamanho do UAV, além disso, a decolagem e pouso totalmente autônomos para este tipo de UAV são operações complexas até mesmo na aviação. Alguns trabalhos encontrados fazem uso de visão computacional para auxiliar no pouso autônomo, porém, apenas essa técnica torna o pouso arriscado. O pouso automatizado em modernas aeronaves, civis e militares, é garantido através do ILS (*Instrument Landing System*) – trata-se de um sistema que usa os sinais de rádio de diversas antenas, instaladas em pontos específicos do aeroporto – que mantém o avião alinhado à pista de pouso e com ângulo de

descida apropriado, ou seja, a forma mais segura encontrada na aviação é a utilização de mecanismos tanto na aeronave como também no aeroporto.

Para determinar a altitude dos UGVs geralmente os pesquisadores utilizam sensores barométricos, porém, a utilização apenas deste sensor para medição de altitude na operação de UAVs pode ocasionar acidentes, por isso, além de sensores barométricos, deve-se fazer uso de sensores que determinem a distância entre o UAV e o solo, os mais utilizados para este fim são os sonares, que enviam ondas sonoras e calculam o tempo de retorno “eco” das ondas, e ópticos (baseados na emissão de luz, podendo ser laser ou infravermelha). As principais características destes dois tipos de sensores estão relacionadas à precisão, alcance e consumo elétrico, sendo os sensores ópticos os mais precisos e com maior consumo elétrico e os sonares os de maior alcance e menor consumo elétrico. Desta forma, deve-se levar em conta as necessidades do UAV e da aplicação para que os sensores mais adequados sejam utilizados.

Apesar da importância em aplicar conhecimentos na criação de novas **interfaces** de controle, poucos trabalhos têm o foco direcionado nos métodos de **interação** entre o operador e o UAV. As interfaces de controle dos UAVs são adaptações de produtos comerciais, o que afeta diretamente no controle, reduz o desempenho do operador, principalmente quanto à qualidade dos comandos enviados, aumentando as chances de acidentes, uma vez que o contexto de uso é diferente do original.

O crescente número de acidentes envolvendo UAVs pode estar diretamente relacionado à falta de “**consciência situacional**”, ou seja, o operador não detém todas as informações relacionadas ao ambiente que cerca o UAV, e o corpo do operador não sente as forças que a estrutura do UAV suporta ao executar determinadas manobras. A falta destas sensações torna o controle mais cansativo, pois o operador despense maior atenção para compensar essa deficiência.

As **câmeras** que antes eram utilizadas apenas na captura de imagens para envio ou armazenamento, em alguns trabalhos são utilizadas para outros fins, como por exemplo, auxiliar no pouso autônomo de pequenos UAVs em ambientes fechados, onde o uso de GPS é inviável devido à pouca precisão. As soluções que fazem uso de câmeras no auxílio ao voo autônomo utilizam dispositivos *onboard* para o **processamento das imagens** (aquisição da imagem, filtragem e segmentação, detecção de bordas, reconhecimento de padrões entre outros), tais dispositivos devem ser compactos e apresentar baixo consumo elétrico.

2.4.4 Desafios gerais

Alguns desafios são inerentes a qualquer tipo de UV, embora as soluções para suplantar tais desafios possam ser diferentes para cada um deles, por exemplo: a utilização de **GPS** (**Global Positioning System**) é citada em todos os trabalhos onde o UV possui algum grau de automação, porém, este **sistema de localização** pode apresentar uma **margem de erro** que

afeta negativamente o resultado em atividades onde a precisão seja um requisito fundamental, daí a necessidade de utilização do sistema de posicionamento **DGPS** (*Differential Global Positioning System*) ou simplesmente GPS Diferencial, esta solução requer uma estação de comparação, além do sistema convencional de posicionamento GPS.

A **comunicação** é um item fundamental no domínio de UVs, a tecnologia escolhida para transmitir as informações entre UVs e GCS dependerá principalmente do fluxo de dados que será transmitido e da distância entre o UV e a GCS. Para transferir apenas dados de controle e informações telemétricas a curtas distâncias, soluções baseadas no padrão IEEE 802.15.4 como o XBee são suficientes, caso seja necessário o envio de imagens capturadas por câmeras *onboard*, faz-se necessário padrões como o IEEE 802.11 para curtas distâncias ou o padrão IEEE 802.16 para longas distâncias.

Os diversos componentes instalados em um UV podem gerar **interferência nas comunicações** ou mesmo nos sensores, desta forma, a localização das antenas e sensores deve ser definida de forma que tais interferências não ocorram, por exemplo, os UVs que utilizam motores à gasolina dependem de um dispositivo conhecido como CDI (*Capacitor Discharge Ignition*) que envia uma corrente alta para que a vela de ignição gere a faísca necessária ao funcionamento do motor, causando uma forte interferência junto ao motor e ao CDI, por isso, os componentes responsáveis pela comunicação e também muitos dos sensores devem ser instalados a uma distância segura do CDI.

O componente eletrônico embarcado mais importante em um UV é seu elemento de processamento de dados, que pode ser um **computador** convencional de uso doméstico ou equipamentos dedicados desenvolvidos unicamente para o fim de controlar UVs, porém, devido às suas restrições, os UAVs necessitam de equipamentos de processamento com baixo consumo elétrico, pouco peso e volume.

Os desafios relacionados a *software* podem ser divididos entre os *softwares* presentes na GCS e no UV. Quanto ao *software* instalado no UV, algumas plataformas de arquitetura fechada tratam os dados e fornecem apenas as informações de controle aos atuadores, nestes casos o desenvolvedor não é capaz de ler ou alterar o código-fonte do sistema. Apesar de apresentarem maiores dificuldades de montagem e configuração, algumas soluções de código aberto, permitem ao desenvolvedor estudar e alterar o sistema, adequando-o às suas necessidades. A complexidade destes programas está diretamente ligada às características do UV, bem como da missão que será executada. O sistema deve ser eficiente em vários aspectos, como: boa gerência no uso de recursos (memória e consumo elétrico); velocidade de processamento compatível com os requisitos do UV; em muitos casos o sistema deve ser tolerante às falhas; e o sistema de planejamento **presente no UV** deve ser capaz de tratar qualquer obstáculo que surja durante a operação para que o UV alcance as metas estabelecidas no planejamento da missão.

O desenvolvimento de sistemas de planejamento de rotas para missões autônomas, bem como de sistemas **embarcados** para tomada de decisão, são desafios que ainda apresentam

lacunas a serem tratadas em trabalhos futuros, principalmente aqueles que fazem uso de visão computacional e de inteligência artificial.

2.5 Discussão

Muitas vezes para conseguir aliar baixo custo ao domínio de UVs é necessário utilizar produtos comerciais não destinados para este fim, apesar de resolver problemas relacionados aos custos de produção do UV, essa alternativa pode reduzir a qualidade do mesmo, principalmente quando as interfaces de controle e *feedback* são desenvolvidas com estes produtos. Mesmo as pesquisas que não estão voltadas diretamente ao estudo de interfaces devem analisar cuidadosamente se esta prática afeta no resultado do trabalho. Os desafios existentes no desenvolvimento e operação de UVs podem variar de acordo com o tipo do UV ou sua aplicação, desta forma, serão apresentados os principais desafios de acordo com o tipo de UV.

As tecnologias de comunicação devem ser definidas de acordo com o tipo de UV, uma vez que os ambientes de operação variam muito de acordo com o UV escolhido, por exemplo: o ambiente de operação de AMUVs inviabiliza a utilização de redes sem fio tradicionais por meio de ondas de rádio, principalmente devido às limitações relacionadas ao alcance e estabilidade do sinal. Essa comunicação pode ser viabilizada por meio de modems acústicos, porém, mesmo essa alternativa possui limitações de alcance e velocidade.

Apesar de importante, questões relacionadas à segurança durante a operação de UVs foram citados em apenas três trabalhos. Diversos aspectos devem ser levados em consideração e, para cada tipo de UV, uma abordagem deve ser implantada. Os possíveis danos que um UV pode causar em caso de impacto devem ser minimizados sempre que possível, UGVs podem tratar esse risco simplesmente utilizando sistemas de parada de emergência, porém, alguns tipos de UVs não podem simplesmente parar para evitar impactos. A segurança na troca de informações de controle também deve ser tratada, principalmente para aqueles UVs utilizados em missões militares.

Outro problema relacionado à segurança envolve o uso do GPS, o UV pode sofrer ataques conhecidos como “GPS *Spoofing Attack*”, que são falsos sinais de GPS enviados com o intuito de alterar a rota planejada para o UV. Essa técnica permitiu que em dezembro de 2011, um UAV americano “Lockheed Martin RQ-170 Sentinel” tivesse seu voo autônomo alterado, de forma a pousar seguramente em um aeroporto iraniano, neste episódio um UAV utilizado em missões de espionagem e ataque, que custa seis milhões de dólares, foi capturado para ser estudado por cientistas iranianos visando a obtenção de tecnologia americana. Alguns trabalhos encontrados fora do escopo desta pesquisa já apresentam soluções para mitigar esse tipo de ataque, por exemplo em [61][62][63][64].

2.6 Conclusão

Este capítulo apresentou o estado da arte dos veículos não tripulados com soluções empregadas para os mais diversos fins. Com base nesse levantamento, identificamos e analisamos desafios de *hardware* e de *software* desse domínio.

De todas as categorias de veículos não tripulados, uma destacou-se das demais pela quantidade de estudos encontrados, ou seja, os aéreos, que são os veículos não tripulados mais versáteis encontrados durante a pesquisa, entretanto, é o tipo de UV de controle mais complexo, ao ponto de necessitar de um sistema exclusivo para o controle e estabilização do voo.

O grande número de trabalhos envolvendo UAVs e a escassa oferta de trabalhos que apresentem interfaces de controle mais completas, inclusive com recursos hápticos ou táteis para a apresentação de dados obtidos dos sensores, servem de motivação para trabalhos futuros, desde a pesquisa mais aprofundada de interfaces hápticas e suas aplicações em UAVs ou o desenvolvimento de uma interface que auxilie no controle de um UAV do tipo quadricóptero.

2.6.1 Comentários finais

Por meio deste levantamento foi possível identificar os tipos existentes de UVs, bem como suas aplicações práticas e principais limitações, também foi possível identificar e entender o funcionamento dos principais componentes utilizados por cada tipo de UV, como: dispositivos de controle, dispositivos de comunicação, atuadores e sensores utilizados. Devido ao maior número de pesquisas encontradas relacionadas aos UAVs e à diversidade de aplicações dos mesmos, decidimos então realizar um segundo levantamento, voltado apenas aos UAVs com foco em suas interfaces de controle e *feedback*.

Capítulo 3

Interfaces do Usuário em Sistemas de Controle de Veículos Aéreos Não Tripulados: Estado da Arte, Desafios e Oportunidades

3.1 Introdução

Os UAVs, são aeronaves de variados tipos e tamanhos, controladas remotamente, semi-autônomas ou totalmente autônomas e que devido ao seu elevado custo, durante muitos anos, foram objeto de estudo exclusivamente em âmbito militar. No entanto, devido aos avanços e barateamento de componentes microeletrônicos e avanços nos sistemas de controle, profissionais civis de diversas áreas do conhecimento passaram a realizar pesquisas nesse domínio. Atualmente, um usuário final pode usar em casa o seu próprio UAV, tais como os UAVs vendidos para os praticantes de modelismo (e.g. AR.Drone, Blade, IdeaFly e Walkera⁸), os UAVs montados por entusiastas por eletrônica (e.g. ArduPlane/ArduCopter, MultiWii, FlyPilot⁹) e os UAVs de uso profissional nas mais diversas áreas (e.g. DJI, ZeroUAV e GAIUI¹⁰). Várias das soluções citadas anteriormente permitem que mesmo usuários com pouco ou nenhum

⁸ Os sites desses produtos são: <http://ardrone2.parrot.com>, <http://www.bladehelis.com>, <http://www.idea-fly.com> e <http://www.walkera.com>.

⁹ Os sites desses produtos são: <http://www.ardupilot.com>, <http://www.multiwii.com> e <http://www.flyduino.net>.

¹⁰ Os sites desses produtos são: <http://www.dji.com>, <http://www.zerouav.com> e <http://www.gaiui.com.tw>.

conhecimento sobre aviação consigam controlar seu UAV, embora algumas aplicações requeiram a participação efetiva de um piloto ou mesmo de uma equipe de controle.

A GCS, via suas UIs, é um dos componentes essenciais no controle de UAVs. Através dessas UIs o piloto/operador pode controlar e visualizar importantes informações sobre o UAV e o ambiente que o cerca. Os estudos de Interação Humano-Computador – *Human-Computer Interaction* (HCI) – podem trazer benefícios para as UIs dos sistemas de controle de UAVs, desde um melhor aproveitamento dos recursos até a redução de acidentes. Segundo Cavett et al. [65], são raros os estudos envolvendo a área de HCI para melhorar o controle de UAVs, seja apresentando UIs de controle que facilitem os comandos do piloto/operador, seja para criar novas UIs que possam trazer uma quantidade maior de informações úteis para as tomadas de decisão.

Hing et al. [47] afirmam que o crescente número de acidentes envolvendo UAVs pode estar relacionado à falta de **consciência situacional**¹¹, ou seja, o piloto/operador não detém todas as informações relacionadas ao ambiente que cerca o UAV e não sente as forças que a estrutura do UAV suporta ao executar determinadas manobras. Em outro trabalho Hing et al. [46] afirmam que mesmo pilotos de aeronaves tripuladas relataram dificuldades para controlar UAVs, principalmente devido à ausência de consciência situacional, tal ausência torna a pilotagem mais cansativa dada a atenção despendida pelo piloto/operador.

Nesse contexto, apresentamos neste capítulo o estado da arte dos UAVs, sob a ótica das suas interfaces de controle. Assim, o objetivo de investigação deste trabalho é melhor compreender as UIs utilizadas para controlar UAVs, bem como, identificar desafios e oportunidades de pesquisa em HCI nesse domínio.

Este capítulo está organizado da seguinte maneira: a Seção 3.2 apresenta uma visão geral sobre os UAVs; a Seção 3.3 apresenta o estado da arte das interfaces de controle de UAVs; a Seção 3.4 descreve os desafios e oportunidades relacionados às UIs de controle de UAVs; a Seção 3.5 discute os resultados deste capítulo e apresenta as considerações finais.

3.2 Veículos Aéreos Não Tripulados

Os primeiros trabalhos envolvendo UAVs surgiram no início da década de 1970 com veículos aéreos de asa fixa, ou seja, aviões. Nessa época, ainda não existiam trabalhos acadêmicos voltados para a automação dos voos e os UAVs eram controlados remotamente apenas por pilotos.

¹¹ Ozartan et al. [100] entendem que a consciência situacional nesse domínio é de dois tipos, i.e. interna e externa. A interna está relacionada à percepção humana dentro do ambiente físico onde os eventos ocorrem. Por sua vez, a externa refere-se aos eventos que são convertidos em dados digitais, transmitidos e tratados para em seguida apresentar uma percepção virtual dos dados recebidos.

Atualmente, diversos trabalhos apresentam soluções que permitem aos UAVs a capacidade de voo parcial ou totalmente autônomo. Assim como as aeronaves tripuladas, os UAVs podem ser divididos em duas categorias, i.e., os mais leves que o ar (e.g. balões e dirigíveis) e os mais pesados que o ar (e.g. aviões, helicópteros e multicópteros). No entanto, considerando os objetivos deste trabalho, os UAVs serão apresentados em três categorias:

- i) de asas fixas;
- ii) de asas rotativas;
- iii) mais leves que o ar.

Os **UAVs de asas fixas** podem apresentar-se em diversos tamanhos. Os maiores UAVs de asas fixas são usados em missões militares de vigilância, reconhecimento, espionagem e de ataque, podendo permanecer em voo por dezenas de horas e percorrer milhares de quilômetros em uma única missão. São mais indicados para aplicações que requerem maior autonomia de voo, maior velocidade, maior altitude de operação e grandes deslocamentos. Em relação à posição ou formato de suas asas podem ser: de asa alta (Figura 5a), asa baixa, asa delta, também conhecida como *Zagi* (Figura 5b) e *Canard* (Figura 5c).

Os **UAVs de asas rotativas** são mais versáteis que os UAVs de asas fixas, visto que podem decolar e pousar verticalmente¹², além de poder permanecer em voo pairado. Podem ser classificadas de acordo com a quantidade de conjuntos motor/hélice, sendo as configurações mais comuns: helicóptero (Figura 5d), tricóptero, quadricóptero (Figura 5e), pentacóptero, hexacóptero e octacóptero. Em relação aos tipos de hélices utilizadas podem ser: passo fixo (conseguem aumentar ou diminuir o empuxo alterando a velocidade do motor) e passo variável (conseguem aumentar ou diminuir o empuxo alterando o passo da hélice).

Os **UAVs mais leves que o ar** – balões e dirigíveis – (Figura 5f) são usados principalmente em missões de monitoramento ambiental e publicidade. Seu voo lento e sua pouca manobrabilidade tornam esse tipo de UAV muito sensível às condições climáticas. Assim como os UAVs de asas rotativas, estes UAVs também podem ficar parados em uma posição desejada, decolar e pousar verticalmente.

¹² Recurso denominado de VTOL (*Vertical Take-Off and Landing*).

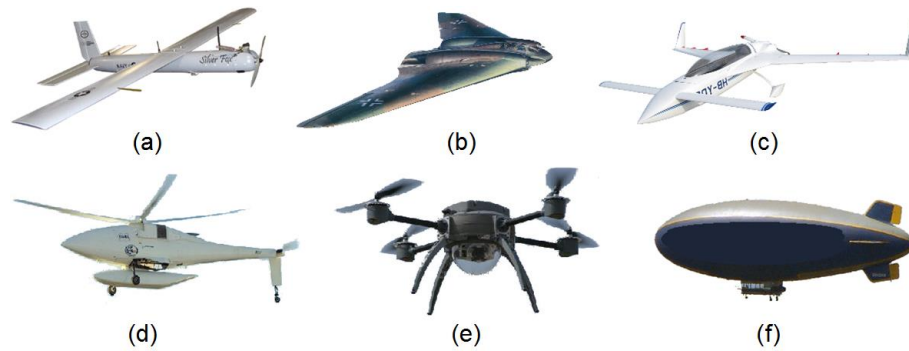


Figura 5. Tipos de UAVs: (a) avião asa alta (b) Zagi (c) Canard (d) Helicóptero (e) Quadricóptero (f) Dirigível. Tipos de UAVs: (a) avião asa alta (b) Zagi (c) Canard (d) Helicóptero (e) Quadricóptero (f) Dirigível.

Todos os tipos de UAVs possuem um requisito em comum que é a necessidade de uma GCS. A GCS é encarregada pelo controle do UAV e por receber os dados telemétricos dos sensores presentes nos UAVs e apresentar essas informações em uma interface, geralmente gráfica – **Graphical User Interface** (GUI) –, ao piloto/operador que está controlando o UAV (vide exemplo da Figura 6). A GCS pode ser composta por um conjunto de periféricos instalados em um ou mais computadores ou mesmo por equipamentos portáteis, tais como, *tablets* ou *smartphones*.



Figura 6. Exemplo de GUI de uma GCS. Fonte: [56].

Os UAVs semiautônomos são controlados remotamente por um piloto/operador por meio de dois tipos de interfaces da GCS, i.e. (i) a interface de comando, e (ii) a interface de *feedback*. A interface de comando permite ao piloto/operador enviar comandos ao UAV, enquanto que a interface de *feedback* deve exibir os dados telemétricos que apoiam o piloto/operador no controle

do UAV, seja exibindo as informações técnicas relevantes do seu voo ou, até mesmo, apresentando advertências quanto às manobras que ultrapassem os limites de segurança de voo. Os dados telemétricos mais importantes para qualquer tipo de UAV são: altitude, autonomia, velocidade horizontal e velocidade vertical. Algumas informações são importantes apenas para UAVs de asas fixas, tais como: velocidade horizontal, atitude e forças atuantes do/sobre o UAV. Como os dirigíveis são mais sensíveis à velocidade e direção do vento, obter dados relacionados às condições de vento são fundamentais para garantir a segurança dos mesmos.

Geralmente, as informações telemétricas são apresentadas da mesma forma que nos principais instrumentos presentes em aeronaves civis ou militares, ou seja, velocidade do ar (Figura 7a), horizonte simulado (Figura 7b) que apresenta a atitude, o altímetro (Figura 7c) que apresenta a altitude, a bússola (Figura 7d) que indica a direção, e o variômetro (Figura 7e) que apresenta a velocidade vertical. Cabe ressaltar que a apresentação das informações telemétricas nesse formato requer conhecimento prévio por parte do piloto/operador.



Figura 7. Principais elementos de uma GUI de uma GCS (a) velocidade do ar, (b) horizonte simulado (c) altímetro (d) bússola e (e) variômetro.

Julgamos relevante apresentar alguns dos conceitos fundamentais desse domínio nesse capítulo com o intuito de proporcionar um entendimento mais apurado dos trabalhos da literatura que compõem o estado da arte do tema.

3.3 Estado da Arte de UI para Controle de UAVs

Os trabalhos da literatura que compõem o estado da arte das interfaces de usuário para controle de UAVs foram organizados pelos tipos de UAV, de asas fixas, de asas rotativas e mais leves que o ar.

3.3.1 Asas Fixas

As informações dos sensores existentes nas aeronaves tripuladas são apresentadas em instrumentos instalados em seu painel de instrumentos, respeitando padrões de tamanho, posição, formato, escala e até mesmo quanto aos símbolos apresentados. Baseando-se neste fato,

Kang e Yuan [66] desenvolveram um sistema que apresenta as informações telemétricas através de uma GUI, com imagens simulando os principais instrumentos do painel de um avião modelo Cessna 172R, tais como, velocidade do ar, horizonte simulado, altímetro, força lateral, indicador de rumo e taxa de subida ou descida. Além disso, uma tela adicional denominada “Modo de Diagnóstico” pode ser apresentada ao usuário, informando numericamente os dados obtidos pelos sensores presentes no UAV. Seguindo o mesmo princípio, Zafar e Humza [32] desenvolveram uma GUI capaz de apresentar os dados dos sensores simulando os equipamentos encontrados em aviões, para controlar o UAV de asas fixas eles utilizam um *joystick* de videogame.

Como alternativa ao uso do horizonte simulado (Figura 7b) Quigley et al. [67] propuseram a utilização da imagem de um pequeno avião visto por trás, simulando as atitudes do UAV. Para controlar o UAV os autores [67] utilizaram um *joystick* para simuladores de voo e uma interface capaz de interpretar comandos de voz do piloto/operador e convertê-los em ações que deveriam ser executadas pelos atuadores do UAV.

Kayayurt et al. [68] afirmaram que a GCS é um dos principais componentes em um sistema de controle de UAVs, sendo tratada como *cockpit* virtual. Baseado nesse entendimento, os autores desenvolvem uma GUI seguindo padrões e normativas internacionais relacionados à segurança do voo. A preocupação com a segurança levou os autores a utilizarem sistemas operacionais de tempo real tanto na GCS como no UAV, bem como utilizam redundância de placas de controle de voo e estabilidade.

Preocupados com o crescente número de acidentes envolvendo UAVs e com a possibilidade de que esses acidentes estivessem relacionados aos comandos executados pelo piloto/operador, Crescenzo et al. [69] propuseram um conjunto de dispositivos que formam uma avançada interface para controle de UAVs, composta por:

- i) uma tela sensível ao toque que exibe a interface de controle do sistema;
- ii) uma grande tela capaz de gerar imagens tridimensionais simulando o ambiente onde o UAV está operando;
- iii) recursos sonoros capazes de enviar informações ao piloto/operador, inclusive com voz sintetizada.

A interface de controle é composta de quatro grupos de botões responsáveis pelos comandos básicos que o piloto/operador pode enviar ao UAV, estes comandos são formados por três parâmetros: o que deve ser feito (e.g., voar, fotografar e monitorar), onde deve ser feito (i.e. o alvo para o comando passado no parâmetro anterior), e um grau de prioridade que indica como a tarefa deve ser realizada (e.g., menor consumo de combustível, menor caminho ou maior velocidade). Ainda nessa interface é exibido um mapa da região onde o UAV está sendo operado.

Os elementos apresentados nas GUIs e as interfaces de controle podem variar de acordo com o tipo de UAV ou sua aplicação, em muitos casos a própria GUI pode também ser utilizada como interface de controle, Jasinsk et al. [70] desenvolvem uma GUI onde o piloto/operador

determina pontos de passagem em um mapa, indicando os locais onde o UAV deve executar ações, mesmo com o UAV em voo. Para Ruff et al. [26], o grande diferencial no desenvolvimento de uma GCS eficiente é selecionar os dados telemétricos de maneira correta e transformá-los em informações úteis, pois existe uma grande quantidade de dados que podem ser transmitidos para a GCS. Além de selecionar as informações que realmente são importantes, a forma como elas são apresentadas também é um fator que pode auxiliar ou gerar problemas ao piloto/operador. A GUI desenvolvida em [26] apresenta um mapa da região onde o UAV está em voo, botões de comandos e um quadro onde a imagem do UAV pode ser visualizada de diversos ângulos diferentes, esses autores propuseram a utilização um capacete com sensores capazes de detectar o movimento da cabeça do piloto/operador e convertê-los em ações de controle do UAV, enquanto uma pequena tela presa ao capacete exibe a GUI do sistema.

Akce et al. [71] investigam a eficiência do controle de UAVs por meio de uma Brain-Machine Interface (BMI) Figura 8. Essa interface é capaz de captar sinais elétricos – eletroencefalograma (EEG) – do cérebro do piloto/operador e determinar os comandos que devem ser enviados ao UAV. A BMI utilizada por [71] é do tipo não-invasiva, formada por eletrodos responsáveis pela captura de sinais elétricos em diversas áreas do cérebro. Medições com esse tipo de interface necessitam de tratamentos para eliminar “ruídos”, desta forma, apenas o controle de curvas à direita ou esquerda são feitos nos experimentos, enquanto os controles de velocidade e altitude do UAV são efetuados automaticamente. Os resultados apontam uma pequena diferença de desempenho entre a BMI e o uso do teclado convencional para controlar UAVs de asas fixas.

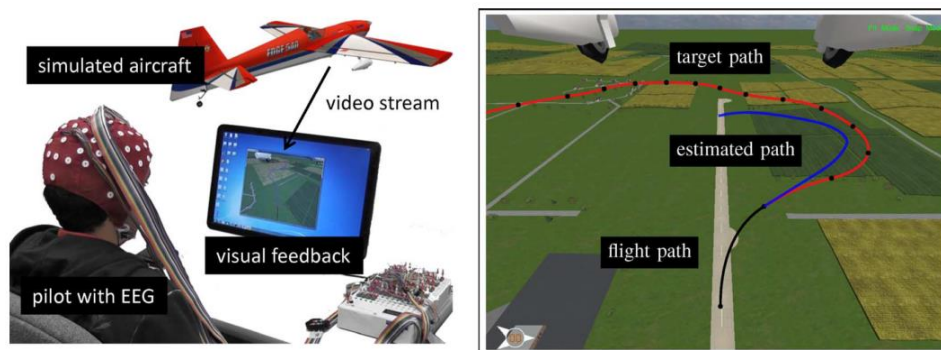


Figura 8. Uso de interface BMI [71]

Para garantir que o piloto/operador consiga manter-se no controle do UAV por períodos longos, Karlen et al. [72] propuseram ligar um sistema, denominado SymBodic, ao corpo do piloto/operador, esse sistema possui sensores capazes de detectar níveis de atenção e um colete vibratório, responsável por alertar baixos níveis de atenção ou quando a aeronave apresentasse atitudes fora do padrão de voo esperado. Duas sessões de testes foram executadas, cada uma com duração de 48 horas e períodos curtos de intervalo para que o piloto/operador possa dormir. Em intervalos regulares de aproximadamente 30 minutos, o simulador fazia com que o UAV mudasse drasticamente de atitude, como se ocorresse um forte vento lateral atingindo-o. Na primeira sessão o piloto não fazia uso do sistema proposto, enquanto na segunda sessão estes recursos

foram utilizados, observou-se que o piloto/operador reagiu mais prontamente quando o sistema SymBodic foi utilizado, principalmente nos momentos onde a habilidade cognitiva foi reduzida devido ao cansaço.

3.3.2 Asas Rotativas

Diversos equipamentos utilizados em computadores e consoles de videogames são adaptados e utilizados como interfaces de controle/*feedback* de UAVs, por exemplo, Al-Khatib et al. [73] desenvolveram um quadricóptero que pode operar autonomamente ou ser controlado através de comandos enviados pelo teclado da GCS, por um *joystick* Wii Nunchuck ou mesmo por um *smartphone* ou *tablet*. A empresa ARDrone comercializa um quadricóptero denominado Parrot. Esse UAV, categorizado como brinquedo, pode ser controlado por um aplicativo instalado em um dispositivo móvel e, devido aos recursos disponíveis e a facilidade de uso, tornou-se uma boa plataforma para estudos com UAVs. Berezny et al. [33] apresentaram uma solução para controlar um Parrot por meio de um sensor Kinect instalado em um computador, este sensor captava os gestos executados pelo piloto/operador, em seguida, o computador tratava as informações capturadas, decodificava o gesto, determinava a manobra desejada e então enviava comandos para o UAV. Por sua vez, Shan e Sharlin [34], também utilizaram um Parrot para experiências de controle por meio de gestos, a câmera do UAV era responsável por capturar as imagens enquanto seu sistema embarcado decodificava os gestos, transformando-os em comandos para o quadricóptero. [34] definiram que os gestos utilizados baseiam-se nos mesmos efetuados por falcoeiros no controle de seus falcões. Ainda utilizando o reconhecimento gestual, Pfeil et al. [74] criaram cinco diferentes formas de interagir com um Parrot, desde movimentos imitando um avião com os braços abertos a simples gestos com as mãos que coordenam os movimentos de rolagem¹³, arfagem¹⁴ e guinada¹⁵.

¹³ A rolagem é uma manobra onde o UAV gira no seu eixo longitudinal.

¹⁴ A arfagem é uma manobra onde o UAV gira no seu eixo transversal, auxiliando, desse modo, o ganho/perda de altitude.

¹⁵ A guinada é uma manobra onde o UAV gira no seu eixo vertical auxiliando, desse modo, a realização de curvas à esquerda ou à direita.



Figura 9. Gestos para controle de um UAV [74].

Azfar e Hazry [75] desenvolveram uma GUI que apresenta os principais componentes (i.e. horizonte simulado, altímetro e bússola) e também outras informações telemétricas, como nível de carga da bateria do UAV e aceleração dos motores. Todas essas informações gráficas também foram apresentadas numericamente em seus respectivos campos. Segundo esses autores [75] essa interface gráfica desenvolvida para multicópteros pode ser adaptada aos mais diversos tipos de UAVs. Wypych e Kuester [76] apontaram o surgimento de plataformas para multicópteros voltadas à prática de modelismo rádio controlado como principal motivador das pesquisas neste domínio, tanto que apresentaram um sistema para obtenção de imagens aéreas que tem como principal interface de controle um rádio transmissor usado por praticantes de modelismo e como interface de saída uma GUI, com informações telemétricas obtidas dos sensores presentes no UAV.

Preocupados com a utilização de interfaces de controle ou *feedback* não voltados para o domínio de UAVs, Sándor et al. [77] mediram a fadiga e a precisão no controle de um quadricóptero com a utilização de três interfaces de controle diferentes: rádio transmissor utilizado por modelistas (Figura 10a); *joystick* convencional utilizado em simuladores de voo (Figura 10b); e uma IMU capaz de detectar movimentos de inclinação e aceleração em três eixos (Figura 10c). Os resultados dos experimentos executados foram: o pior resultado obtido foi utilizando o rádio transmissor, chegou-se à conclusão que apenas usuários mais experientes deveriam controlar o UAV com esse tipo de interface; o *joystick* utilizado em simuladores de voo apresentou resultados melhores que o rádio; e a interface desenvolvida utilizando a IMU presa à uma das mãos do usuário foi o mais preciso. Para os autores, os resultados favoráveis ao controle através da IMU fortaleceram a ideia de que os comandos executados por movimentos das mãos apresentam-se como a melhor alternativa para o controle de quadricópteros.



Figura 10. Interfaces utilizadas por [77] – a) Rádio transmissor; b) *Joystick* e c) IMU.

Com o objetivo de medir a fadiga durante a operação de UAVs, Ming et al. [78] propuseram duas técnicas para o controle de um pequeno UAV de reconhecimento, utilizando uma tela sensível ao toque e utilizando um *joystick*. Para avaliar a fadiga durante a operação é utilizada a metodologia NASA-TLX¹⁶. Em relação ao tempo de treinamento o controle por meio de tela sensível ao toque mostrou-se ligeiramente menor que utilizando o *joystick*, o desempenho também é melhor com a tela sensível ao toque, porém, o *joystick* apresentou maior precisão na execução dos comandos.

A segurança na operação de veículos não tripulados preocupa Murphy et al. [53], que avaliaram os trabalhos de resgate após os furacões Vilma e Katrina. Três pessoas foram responsáveis pela operação de um UAV de asas rotativas, um piloto/operador que controlava o UAV por meio de um rádio transmissor utilizado por modelistas, um especialista em desastres que operava a câmera do UAV e um responsável por observar o ambiente à procura de possíveis riscos onde o UAV estava em operação. Diversos aspectos foram estudados, entre eles, a fadiga que esse tipo de operação pode causar ao piloto/operador. Após quatro voos de 10 minutos e mais dois voos de 15 minutos ficou evidente o cansaço do piloto/operador. As missões seguintes foram feitas com outro UAV de asas rotativas. Este UAV possuía duas câmeras que enviavam as imagens para as lentes de óculos especiais que eram usados pelo piloto/operador e pelo operador da câmera móvel. Baseado no trabalho [53], Peschel [79] utilizou o mesmo modelo de equipe para desenvolver um sistema capaz de manipular objetos utilizando UAVs. Para esse autor [79], os desenvolvedores deveriam apresentar interfaces táteis que pudessem passar informações sobre o objeto a ser manipulado e também sobre o ambiente.

A principal interface de *feedback* utilizada em todos os trabalhos é a GUI, porém, outras formas de interação podem aumentar a segurança e a precisão no controle de UAVs, Adam e Mark [14] utilizam um *joystick* háptico que enrijecia quando o UAV detectava obstáculos próximos e o piloto/operador tentava movimentá-lo em direção ao obstáculo, quanto mais perto o UAV se aproximava do obstáculo, maior era a rigidez do *joystick*. Da mesma forma Hyong et al. [80] apontavam que o *feedback* háptico pode ser usado para alertar o piloto/operador sobre os obstáculos no ambiente, [80] sugeriram que esta técnica pode ser utilizada também para auxiliar o piloto/operador a manter o UAV no trajeto esperado. Lam et al. [81] também partiram da ideia

¹⁶ Segundo Hing et al. [46], essa técnica é capaz de mensurar a exigência mental, física e temporal, bem como, a performance durante a execução de uma determinada tarefa.

de fornecer *feedback* tátil ao piloto/operador, porém, decidiram por não usar somente o enrijecimento do *joystick*, pois apenas essa técnica aumenta a fadiga durante a operação, a forma abordada por [81] foi fornecer o enrijecimento moderado limitando o curso do *joystick*, ou seja, se o piloto/operador tentava controlar o UAV em direção ao obstáculo o *joystick* enrijecia até um limite mínimo seguro de proximidade, quando o mesmo não mais permitiu comandos naquela direção. Os resultados obtidos por [81] mostraram que o *feedback* por enrijecimento do *joystick* apresenta um maior grau de segurança, em contrapartida, aumenta a carga de trabalho do piloto/operador, assim, a solução apontada pelos autores [81] manteve a segurança sem aumentar a carga de trabalho.

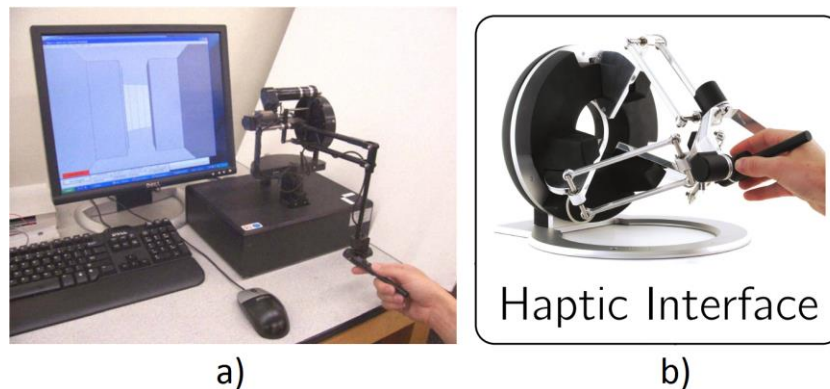


Figura 11. Interfaces táteis utilizadas por: a) [14] e b) [80][82].

Rüesch et al. [82] também fizeram estudos com um *joystick* capaz de prover *feedback* ao piloto/operador durante o controle de um quadricóptero, em um dos experimentos executados pelos autores a GCS e o piloto/operador encontravam-se na Universidade de Twente/Holanda enquanto o UAV encontrava-se na Universidade ETH em Zurich/Suíça. O quadricóptero utilizado por [82] possuía um sistema que controlava os comandos definidos como de baixo-nível (controles de voo e estabilidade), desta forma o piloto/operador ficava responsável pelos comandos definidos como de alto-nível (controlar as principais manobras do UAV). Smisek et al. [83] reforçaram o conceito de que os pilotos de aeronaves reais recebem *feedback* multissensorial, enquanto o piloto/operador de UAV geralmente recebe apenas estímulos visuais. Esses autores [83] afirmaram que fornecer *feedback* háptico aumenta a fadiga durante o controle do UAV, por isso, apresentaram o estudo das propriedades do sistema neuromuscular humano, os resultados foram usados como entradas na criação de um conjunto de parâmetros para o controle da rigidez do *joystick*, sem que isso acarretasse maior esforço por parte do piloto/operador em decorrência do aumento na rigidez do *joystick*.

Vargas et al. [84] instalaram uma barra com dois LEDs infravermelhos em um pequeno helicóptero, com isso *joysticks* WiiMote do videogame Nintendo Wii, instalados em pontos fixos de um ambiente, determinavam a posição exata do UAV, completando o sistema de controle, uma câmera de vídeo instalada no teto do ambiente conseguia definir a orientação do UAV e suas possíveis manobras, os dados obtidos pelos *joysticks* e pela câmera eram tratados na GCS que enviava comandos para ajustes no UAV, os resultados garantiram o correto posicionamento do

UAV demonstrando que o sistema era adequado para o rastreamento de voo em ambientes fechados.

Kasahara et al. [85] apresentaram uma solução como sendo o controle remoto universal para controlar dispositivos móveis, inclusive um UAV de asas rotativas, bastando para isso um *tablet* com uma câmera frontal. O usuário observava na tela do *tablet* a imagem capturada pela câmera, no momento em que a câmera detectava um marcador preso ao UAV, bastava ao usuário efetuar comandos através da tela *touch screen* do *tablet*. Para mover o UAV podia-se tocar na imagem do mesmo e arrastar na direção desejada ou, tocava-se na imagem do UAV e mantendo o toque, movimentava-se o *tablet* na direção desejada. Diferente de outras formas de interação, este sistema utilizava o UAV como referência para os comandos, independentemente da posição ou distância que esteja do usuário, quando o marcador do UAV era movido para uma posição na tela, o UAV fazia o mesmo movimento.

3.3.3 Mais leves que o ar

Visando reduzir a quantidade de pessoas responsáveis pelo controle de um UAV mais leve que o ar, do tipo dirigível, Righetti et al. [11] desenvolveram uma interface háptica que informava as condições do ambiente em que o UAV estava executando a missão. Como os dirigíveis são muito sensíveis às condições do vento e o piloto/operador não tem como identificar tais condições à distância, um cinto vibratório desenvolvido por esses autores informava ao piloto/operador a velocidade e direção do vento no ambiente onde o UAV estava operando, também desenvolveram óculos especiais que apresentavam as imagens capturadas pela câmera do UAV, enquanto sensores inerciais presentes nestes óculos permitiam que o piloto/operador controlasse a posição da câmera apenas movimentando a cabeça, assim, caso quisesse movimentar a câmera bastava movimentar a cabeça no sentido desejado, dessa forma, suas mãos ficavam livres para o controle do UAV através de um rádio transmissor.

3.3.4 Diferentes UAVs

Uma solução de GUI para uso tanto em UAVs de asas fixas como de asas rotativas foi proposta por Gong et al. [13], nela foram apresentadas imagens capturadas pela câmera presente no UAV, um mapa com o posicionamento e atitude do UAV. Rangel et al. [56] apresentaram um sistema portátil para controle de UAVs, a solução desenvolvida era composta de uma GCS capaz de interagir com UAVs de asa fixa e de asas rotativas, para isso, um equipamento, denominado “Caixa de Integração” instalado no UAV, garantia o correto funcionamento do sistema para os dois tipos de UAVs. Nessa Caixa de Integração estava o microcontrolador responsável pelo tratamento dos dados recebidos pelos sensores para, de acordo com o tipo de UAV em uso, enviar comandos aos atuadores. A Caixa de Integração permitia também que sensores pudessem

ser instalados ou removidos facilmente, tornando o sistema bastante flexível. Duas GUIs podiam ser utilizadas pelo piloto/operador, em ambas, a imagem capturada pela câmera era o principal elemento da tela e as informações telemétricas eram apresentadas como uma camada adicional à imagem principal, inclusive as informações quanto à atitude do UAV (Figura 6).

Soluções multi-modais, também foram aplicadas com o intuito de aumentar a consciência situacional, Maza et al. [86] propuseram um sistema que à princípio não controlava nenhum UAV, mas era capaz de simular a troca de informações entre a GCS e um UAV. A apresentação das informações ao piloto/operador era conseguida simultaneamente de três maneiras: visual, sonora e háptica. O sistema era composto de três monitores sensíveis ao toque, três dispositivos vibratórios sem fio distribuídos na mão esquerda, direita e no peito do piloto/operador, um mouse, fone de ouvido e alto-falantes que eram utilizados para a saída do sistema de áudio 3D.

3.3.5 Síntese

A Tabela 3 apresenta as interfaces utilizadas nos trabalhos que compõem o estado da arte de UIs para o controle de UAVs, enquanto a Tabela 4 apresenta uma visão geral dos dispositivos de *input/output* empregados nas interfaces de controle de UAVs descritos anteriormente.

As interfaces físicas de *input* – e.g., rádios transmissores, *joysticks*, teclados e mouses – são os dispositivos mais utilizados para o controle de UAVs. Outras formas de envio de comandos também são exploradas, como as interfaces gráficas e gestuais, sendo esta última mais utilizada em UAVs de asas rotativas. Três formas de interação são pouco exploradas, a utilização de interface háptica para o envio de comandos, interfaces sonoras e cérebro-computador. A Tabela 4 sintetiza os dispositivos empregados nas interfaces de controle/*feedback* das GCSs.

Tabela 3. UIs empregadas no controle de UAVs.

		Tipos de Interfaces					
		Gráfica	Física	Háptica	Gestual	Sonora	Cérebro-computador
Controle de UAVs	Interface de Comando (<i>input</i>)	[14][69][85][86][78][79]	[14][75][13][81][86][78][53][67][56][11][82][26][77][84][76][32]	[77]	[33][74][32]	[67]	[14]
	Interface de <i>Feedback</i> (<i>output</i>)	[71][73][33][75][69][13][80][70][66][85][68][86][86][78][53][67][56][11][82][26][34][76][32]	[47][72]	[14][80][81][86][11][82][83]	-	[69][86][67]	-

Tabela 4. Dispositivos empregados nas GCSs.

Tipo de UAV	Input	Output	Input/Output
Asas fixas	Joystick para simulador de voo Teclado convencional Sensores EEG Comandos de voz	Tela 3D Tela convencional Caixas de som Cabine de simulador	Tela sensível ao toque
Asas rotativas	Teclado convencional Joystick para simulador de voo Joystick Wii Nunchuck Joystick WiiMote Joystick háptico Sensor Kinect Sensores inerciais Câmera de vídeo Tela sensível ao toque	Joystick háptico Tela convencional Caixas de som	Smartphone Tablet
Dirigíveis	Joystick para simulador de voo		Óculos com tela e sensores inerciais

3.4 Desafios e Oportunidades de UI para controle de UAV

Para o melhor entendimento dos desafios do domínio, os mesmos serão agrupados pelo tipo de UAV. Vale destacar que os desafios apresentados a seguir foram identificados após analisarmos o estado da arte do tema.

3.4.1 Desafios comuns aos UAVs

Alguns trabalhos alertam para o **crecente número de acidentes** envolvendo UAVs e relacionam tais eventos às falhas do piloto/operador, tais falhas podem ser decorrentes da **fadiga, má qualidade das interfaces de controle/feedback** ou mesmo pela **falta de conhecimento técnico** sobre aviação. Para aumentar a consciência situacional de forma a reduzir os acidentes, as pesquisas apontam a utilização de *joysticks* capazes de fornecer *feedback* háptico ao piloto/operador. Estes equipamentos são relativamente caros e dependendo da técnica de *feedback* adotada podem aumentar o cansaço do piloto/operador. Como solução para minimizar os erros ocasionados pela fadiga, apenas um trabalho apresentou uma interface que detecta baixos níveis de atenção e fornece um *feedback* háptico para que o piloto/operador retome a atenção.

Outro desafio é apresentar uma **GCS adaptável** aos mais diversos tipos de UAVs e missões, com interfaces de controle/*feedback* adequadas ao controle de UAVs. As informações dos sensores e a forma como os diversos componentes que podem preencher a tela do sistema devem ser intercambiáveis, uma vez que cada tipo de UAV requer elementos diferentes na GUI,

por exemplo: o horizonte simulado é imprescindível para UAVs de asas fixas e asas rotativas, mas não é tão importante para dirigíveis.

O excesso de informações na GUI pode aumentar a fadiga do operador/piloto, dificultando o entendimento das informações durante a operação ou mesmo causando acidentes, principalmente no controle de UAVs de asas fixas e asas rotativas. Diversas alternativas podem ser aplicadas reduzindo a quantidade de informações das GUIs, tornando o controle de UAVs menos cansativo para o piloto/operador. Pesquisas com interfaces hápticas e sonoras podem ser melhor exploradas para advertir o piloto/operador quando necessário sem sobrecarregar sua GUI de informações. As interfaces de *feedback* utilizadas na aviação podem ser aplicadas às GCSs, como por exemplo, o aviso sonoro de ganho ou perda de altitude recebido pelos pilotos de planadores ou as mensagens de voz emitidas pelos sistemas de controle de aviões.

Quanto às GUIs que são apresentadas em **telas sensíveis ao toque**, que são as interfaces de controle/*feedback* mais comuns, são adequadas apenas para comandos que não dependam de **precisão ao toque**, uma vez que o toque na tela não possui a precisão necessária a certas atividades.

O pequeno número de trabalhos encontrados que fazem uso de **interface cérebro-máquina** deve-se às características de funcionamento que esse tipo de interface possui, os principais detalhes que podem comprometer o uso de tais interfaces neste domínio são: a calibragem que deve ser feita para cada piloto/operador; o tempo de execução de filtros capazes de tratar os dados antes que os mesmos possam ser transformados em comandos; o percentual de dados descartados após essa filtragem; situações de risco e conseqüentemente de estresse poderão aumentar ainda mais os ruídos aumentando o número de dados descartados. Por suas características de voo, os dirigíveis podem utilizar sensores EEG como interface de entrada sem que isso cause riscos à operação do mesmo.

Muitas informações obtidas pelos sensores do UAV são utilizadas nos cálculos para o controle do voo e estabilidade, essas informações podem também servir como entradas para **novas interfaces de feedback**, por exemplo: as informações obtidas pelos sensores inerciais podem ser capturadas e tratadas para apresentar *feedback* ao piloto/operador por meio de interfaces hápticas, sonoras ou físicas. Da mesma forma, sensores inerciais acoplados ao corpo do piloto/operador podem atuar como **novas interfaces de controle** para UAVs.

A **segurança** é também um requisito importante no desenvolvimento de qualquer sistema para controle de UAVs e as interfaces de controle/*feedback* são de fundamental importância para garantir tal requisito. Segundo Crescenzo et al. [69] 17% dos acidentes envolvendo UAVs são ocasionados pelo piloto/operador, seja por fadiga ou imperícia, porém, podem também estar relacionados às deficiências em suas interfaces de controle/*feedback*.

3.4.2 Desafios específicos dos UAVs de Asas Fixas

As **interfaces de *feedback*** são os principais alvos de pesquisas para UAVs de asas fixas, em especial as GUIs com imagens que simulam os instrumentos presentes em aeronaves tripuladas. Para auxiliar no controle do UAV as principais informações que devem ser repassadas ao piloto/operador são: atitude, direção, altitude, velocidade vertical e velocidade do ar.

Apesar de Quigley et al. [67] questionarem a utilização do horizonte simulado afirmando que apenas usuários treinados conseguirão fazer uma correta leitura do mesmo, o horizonte simulado é a melhor opção para representar as atitudes do UAV, apresentando não só a atitude como também os graus de inclinação do do UAV. O horizonte simulado pode ser apresentado como uma camada adicional às imagens capturadas pela câmera do UAV, neste caso, o piloto/operador deve ser treinado para o correto entendimento das informações.

As informações sobre velocidade do ar, velocidade vertical e altitude, são importantes principalmente quando o UAV apresentar características de voo onde os valores para esses indicadores estejam em uma faixa de risco, por exemplo, quando o UAV está próximo de sua velocidade mínima ou máxima de voo ou mesmo quando a velocidade de descida ultrapasse os limites seguros do UAV. Em aeronaves tripuladas esse tipo de alerta é apresentado por meio de **luzes de advertência** e de **vozes sintetizadas**, essa técnica também pode ser utilizada para UAVs, principalmente os **alertas sonoros**, uma vez que não sobrecarregam a GUI.

3.4.3 Desafios específicos dos UAVs de Asas Rotativas

O voo em ambientes fechados pode ser feito por UAVs de asas rotativas ou dirigíveis, entretanto, dependendo do espaço onde o UAV deve ser controlado, um dirigível não será uma boa escolha devido ao seu tamanho e pouca manobrabilidade, para esse tipo de operação os UAVs de asas rotativas são mais indicados. Embora a maioria dos projetos utilizem as imagens capturadas pela câmera *onboard* como principal forma de *feedback*, apenas essas imagens não garantirão a segurança durante a operação do UAV em ambientes fechados, onde podem existir **obstáculos em todas as direções**, inclusive **aquelas onde a câmera não está monitorando**. Já existem algumas soluções em que interfaces hápticas são utilizadas para advertir o piloto/operador sobre obstáculos, porém, avisos sonoros ou mesmo advertências na GUI podem também servir de alerta, **umentando a consciência situacional** via interfaces do usuário.

A **ergonomia** é outro fator importante a ser tratado, devendo **priorizar o menor esforço do piloto/operador**. Especial atenção deve ser dada às interfaces de controle dos UAVs de asas rotativas, que apesar de apresentarem a menor autonomia, e conseqüentemente voos mais curtos, exigem do piloto/operador um alto nível de atenção durante toda a operação, enquanto outros tipos de UAVs requerem tal nível apenas em dois momentos, tais como a decolagem e o pouso.

3.4.4 Desafios específicos dos UAVs mais leves que o ar

UAVs do tipo dirigíveis possuem características que permitem um maior grau de interação através da GUI, pois o seu voo é lento e suas possibilidades de manobras bem reduzidas (frente, trás, esquerda e direita), pode-se acrescentar diversas informações à sua GUI sem que isso sobrecarregue a operação do mesmo. Entretanto, como os dirigíveis são lentos e muito sensíveis às condições climáticas, deve-se apresentar em suas interfaces as situações de risco como: velocidade e direção do vento, advertir quanto aos obstáculos a uma distância bem maior que em outros UAVs, bem como alertar quanto à autonomia, garantindo que o UAV possa ser conduzido com segurança ao seu local de pouso.

3.5 Conclusão

Este capítulo apresenta o estado da arte de veículos aéreos não tripulados sob a ótica das suas interfaces de controle/*feedback*. Baseado nesse levantamento identificamos e analisamos desafios e oportunidades relacionados ao tema.

O uso de UAVs está crescendo em todos os domínios e os estudos voltados às interfaces de controle/*feedback* não acompanham tal crescimento. Até mesmo as pesquisas de cunho militar utilizam interfaces comuns a outras atividades, principalmente voltados ao controle de jogos de computador. Na maioria dos trabalhos da literatura, a interface de *feedback* mais citada foi a GUI apresentada em monitores. Embora alguns dos trabalhos apresentados façam uso de caros equipamentos que proveem uma maior consciência situacional, seus custos são proibitivos para a maioria das pesquisas.

Como identificado neste trabalho, ficou mais nítido que no contexto das interfaces de controle, a interface física de controle – e.g., teclado, *joystick* e mouse – são as mais utilizadas. Em relação à interface de *feedback*, a GUI é a mais empregada, no entanto, outras interfaces podem ser melhor exploradas por pesquisadores de HCI. As interfaces hápticas, gestuais, sonoras e cérebro-máquinas podem – ou devem – ser melhor exploradas como soluções alternativas para as interfaces de controle/*feedback* de UAVs.

Como trabalhos futuros, com base nos resultados deste capítulo, indicamos o aprofundamento no estudo de interfaces multi-modais para esse domínio, com foco em interfaces gestuais e hápticas para o controle de quadricópteros.

3.5.1 Comentários finais

A pesquisa sobre interfaces de controle e *feedback* em aplicações que utilizam UAVs mostrou que um dos maiores desafios para o controle destes veículos é a necessidade de o usuário possuir conhecimentos sobre aviação, bem como habilidades para controlar o voo do UAV, tais requisitos nem sempre podem ser atendidos por todos os usuários, por isso, decidimos desenvolver um sistema de entregas automatizadas por meio de UAVs, que não exigirá tais conhecimentos dos usuários, necessitando apenas de um simples cadastro por meio de um *smartphone*, o capítulo seguinte apresentará todas as etapas do desenvolvimento deste sistema.

Capítulo 4

PostDrone University: Veículo Aéreo não Tripulado e Sistema de Entrega no Campus

4.1 Introdução

Os veículos aéreos não tripulados – Unmanned Aerial Vehicles (UAVs) –, também conhecidos como *drones*, são aeronaves operadas remotamente, semi-autônomas ou autônomas, que podem ser empregadas para realizar, por exemplo, monitoramento ambiental, filmagens aéreas e, até mesmo, para a entrega de objetos. UAVs de asas rotativas apresentam características que os tornam boas escolhas para o serviço de entrega de objetos, tais como, a capacidade de decolar e pousar verticalmente, conhecida como VTOL (*Vertical Take-Off and Landing*), e a possibilidade do UAV permanecer em voo pairado.

Os avanços tecnológicos e o barateamento de componentes eletrônicos possibilitaram o surgimento de um novo tipo de UAV de asas rotativas, ou seja, os UAVs multirotores. Apesar da sua relativa simplicidade mecânica, seu controle é bastante complexo, mesmo para efetuar manobras simples ou manter o voo reto e nivelado. UAVs multirotores podem ser classificados de acordo com a quantidade de conjuntos motor/hélice, sendo os mais comuns os seguintes: tricóptero, quadricóptero, pentacóptero, hexacóptero e octacóptero.

Na maioria dos países o emprego de UAVs ainda depende da criação de legislação específica, o que impede a utilização comercial de UAVs, principalmente aqueles capazes de executar tarefas de forma autônoma. A Federal Aviation Administration (FAA), que é o órgão responsável por regulamentar a aviação civil nos Estados Unidos, define que os UAVs devem respeitar algumas normas de segurança, tais como, operar em altitude máxima de voo de 400 metros e manter o UAV sempre dentro do campo de visão de seu operador. Apesar disso,

Markwalter [87] sinaliza que somente em 2014 foram vendidos mais de 257 mil UAVs de pequeno porte nos EUA.

Nesse contexto, este capítulo apresenta um sistema automatizado de entregas utilizando um UAV do tipo multirrotor, onde os usuários finais do sistema, tais como, professores, pesquisadores, analistas, técnicos e alunos, podem utilizar o sistema para enviar objetos entre unidades do campus de uma universidade. Nosso grupo de pesquisa há mais de uma década vem desenvolvendo produtos com diferentes propósitos para o mercado de *Consumer Electronics*, tais como, os Adjustable Interactive Rings for Interactive Digital Television [88] que, inclusive, já foi patenteado [89]. Entendemos que o trabalho apresentado neste capítulo é mais uma solução com potencial para a indústria de eletrônicos, pois apresenta um sistema de entregas automatizadas de pequenos objetos através de um UAV; as tecnologias desenvolvidas, com poucas adaptações, também poderiam ser empregadas em outros contextos de aplicação.

Este capítulo está organizado da seguinte maneira: a Seção 4.2 apresenta trabalhos relacionados; a Seção 4.3 descreve a solução desenvolvida e seus dois principais componentes, ou seja, o UAV e o sistema de entrega; a Seção 4.4 apresenta testes do *drone*; a Seção 4.5 discute os resultados deste trabalho; e a Seção 4.6 apresenta as considerações finais.

4.2 Trabalhos relacionados

A empresa de comércio eletrônico Amazon já está desenvolvendo um sistema, denominado Amazon Prime Air [90], para entrega de produtos em áreas dentro de um raio de 16Km de seus centros de distribuição. Ao finalizar uma compra *online* o cliente pode optar pelo serviço Amazon Prime Air como forma de entrega dos produtos adquiridos e um UAV multirrotor efetua a entrega “na porta” do cliente. O sistema ainda está em fase de desenvolvimento e apenas em 2014 a FAA liberou o uso da solução para testes.

Empresas de transporte como FedEx, UPS e DHL também têm estudado o uso de multirrotores para dinamizar suas entregas. O Deutsche Post DHL Group já possui liberação na Europa para as entregas efetuadas pelo sistema denominado DHL Parcelcopter. Testes com esse sistema também já foram realizados e incluíram a entrega de medicamentos a 12Km de distância [91].

A empresa AMP Electric Vehicles e a University of Cincinnati Research Institute (UCRI) desenvolveram uma solução denominada de HorseFly [92]. Essa solução consiste de um UAV de asas rotativas utilizado para auxiliar os motoristas de caminhões durante as entregas. Cada caminhão de transporte possui um UAV e, quando necessário, o responsável pelo veículo coloca o objeto a ser entregue no bagageiro do UAV. Assim, o sistema embarcado do UAV interpreta o código de barras disponível no objeto a ser entregue com as informações do destinatário e o UAV inicia a entrega de modo semi-autônomo enquanto o motorista continua com as outras

entregas. A entrega não é completamente automatizada, visto que um piloto precisa remotamente controlar o UAV para realizar a aproximação e aterrissagem no endereço de entrega. Ao finalizar a entrega, o UAV retorna ao caminho para iniciar outra entrega ou carregar suas baterias. Para sanar essa limitação, estudos apontam a utilização de visão computacional para apoiar o pouso automatizado e, nessa linha, Yu-Fu et al. [93] ainda afirmaram que os métodos de detecção de objetos baseados no comportamento humano apresentam menor custo e menor peso que métodos tradicionais baseados em radares e lasers.

O UAV de asas fixas proposto por Emma [5] foi capaz de transportar cargas variadas, devido ao fato do compartimento de carga não ser parte fixa da estrutura do UAV permitindo, desse modo, que vários formatos e tamanhos de compartimentos pudessem ser utilizados. Essa solução reduziu o tempo do UAV em solo, uma vez que o processo de carregamento e descarregamento consistia apenas em desacoplar um compartimento de carga e acoplar um outro para a próxima entrega. Outros detalhes estruturais do UAV são modulares, permitindo alterar com certa facilidade sua configuração de voo e, desse modo, aumentando as aplicações do UAV.

O Project Wings da Google [94] apresentou uma solução composta de UAVs de diferentes tipos. Essa solução [94] é capaz de identificar situações onde exista uma emergência médica, determinar a localização e definir a quantidade e o tipo de UAV que melhor se adequa à situação para, em seguida, transportar medicamentos e equipamentos para médicos que estejam realizando o atendimento de emergência.

Para garantir um diagnóstico mais rápido de doenças comuns em várias regiões da África (i.e. HIV, tuberculose e malária), Boshielo et al. [95] desenvolveram um sistema responsável por transportar amostras coletadas (e.g., sangue e saliva) de pacientes residentes em comunidades afastadas para laboratórios clínicos de grandes centros urbanos. Esses autores [95] afirmaram que o tempo gasto no transporte regular terrestre atrasava o diagnóstico e, conseqüentemente, reduzia a eficácia no tratamento dos pacientes. O UAV de asas fixas desenvolvido por [95] foi capaz de transportar uma carga útil de até 500 gramas por distâncias de até 40 quilômetros.

Determinados ambientes não permitem o pouso de UAVs, seja por questões de segurança em locais de conflitos ou mesmo por não existir uma área apropriada para pouso. Nesse sentido, Yakimenko et al. [96] desenvolveram um sistema de entregas que utiliza compartimentos de carga que são lançados por UAVs em voo. A segurança e precisão de pouso são garantidas por paraquedas que controlam o pouso autonomamente. Segundo esses autores [96] o sistema pode chegar a uma precisão de 15 metros de raio do local determinado para a entrega.

Haque et al. [97] apresentaram um sistema capaz de efetuar entregas por meio de um quadricóptero, utilizando um *smartphone* como unidade de processamento embarcado. O *smartphone* recebe informações de entregas via SMS e utiliza o serviço de mapas do Google para localizar o destino da entrega e controlar o UAV durante o percurso.

Em síntese, os trabalhos relacionados, conforme apresentados anteriormente, descrevem soluções onde o UAV não possui uma área delimitada para a entrega da carga. Outras formas de

entrega não fazem uso de áreas para pouso e a carga deve descer presa a cabos, presa à paraquedas sendo lançada do UAV ainda em voo, ou mesmo presa ao UAV, mas que necessitam que um piloto controle remotamente o UAV para fazer a aproximação e aterrissagem de forma visual. Soluções que efetuam as entregas a grandes distâncias normalmente utilizam UAVs de asas fixas, porém, os processos de decolagem e pouso desses UAVs são complexos e dependem de uma grande área para executá-los. Os UAVs de asas rotativas possuem características que facilitam a decolagem e pouso mesmo em áreas pequenas, porém, sua autonomia impede que as entregas sejam efetuadas a distâncias muito longas. Desse modo, não foi encontrado no estado da arte e no estado da técnica uma solução que efetue entregas de forma automatizada em pontos de pouso pré-definidos e que seja de fácil utilização por parte dos usuários finais do sistema.

4.3 PostDrone University

PostDrone University¹⁷ é uma solução de entrega automatizada de pequenos objetos – e.g. documentos, apostilas, livros, correspondências, amostras para análises laboratoriais e componentes eletrônicos – entre unidades do campus de uma universidade por meio de um *drone*.

A solução é composta por um UAV (PostDrone University UAV K-263) e um sistema (PostDrone University System). O sistema, por sua vez, é composto por quatro programas, i.e. o servidor do sistema (PostDrone University System Server), a ferramenta administrativa do sistema (PostDrone University System Admin), o aplicativo cliente do sistema desenvolvido para dispositivos móveis (PostDrone University Android App) e o servidor de comunicação MAVLink (PostDrone University MAVLink Communication Server), que faz a comunicação entre o servidor do sistema e o UAV K-263.

Cada unidade do campus de uma universidade deve possuir um heliponto associado e, desse modo, as entregas são realizadas nesses helipontos, não sendo necessário conhecimento prévio da posição geográfica do heliponto por parte dos usuários finais do sistema, visto que os helipontos vinculados às unidades devem ser previamente cadastrados pelo administrador do sistema com seus dados de latitude, longitude, altitude e, ainda, com a altitude mínima de segurança para voos entre os helipontos, garantindo que o UAV K-263 sempre esteja voando mais alto que o maior obstáculo existente entre os helipontos. Além dos helipontos associados às unidades, também existe um heliponto-base, de onde o *drone* parte para executar as entregas e retorna após concluir as entregas.

¹⁷ <http://www.PostDronedroneuniversity.pairg.ufrn.br>.

4.3.1 UAV

PostDrone University UAV K-263¹⁸ (Figura 12) é um *drone* multirotor de asas rotativas do tipo quadricóptero, desenvolvido para realizar a entrega de pequenos objetos. Optamos por desenvolver um UAV multirotor para a solução PostDrone University, visto que esse tipo de UAV não precisa de uma área grande para realizar decolagens e pousos e, portanto, parece mais adequado para emprego no campus de universidades.



Figura 12. PostDrone University UAV K-263.

A Figura 13 apresenta o diagrama em blocos do *drone* e a Figura 14 os componentes que foram empregados na montagem do UAV K-263. Após montado, o UAV K-263 mede (CxLxA) 64 cm x 64 cm x 36 cm e pesa 2 Kg (sem carga), podendo transportar 500 gramas de carga útil. A autonomia de voo dependerá do peso da carga útil transportada, em testes com carga total (peso do *drone* mais a carga útil) de 2,3 Kg estima-se de 15 minutos. Levando-se em conta que o tempo necessário para se deslocar entre dois pontos a 38 metros de distância é de 20 segundos, estima-se que o *drone* poderá percorrer até 1700 metros antes de requisitar a recarga da bateria.

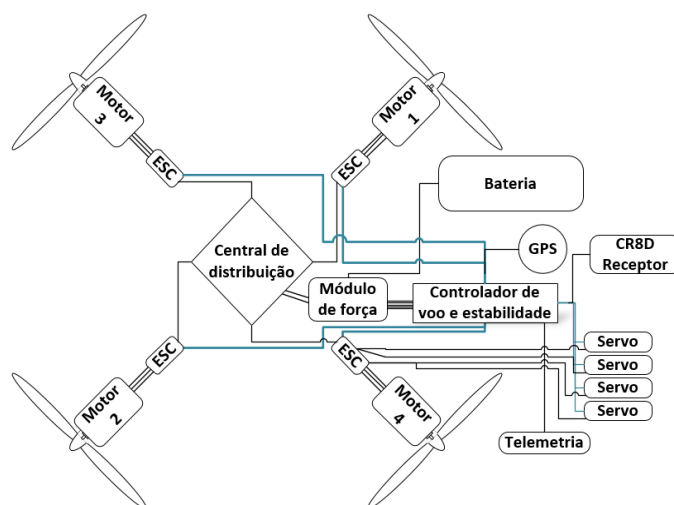


Figura 13. Diagrama de blocos do K-263.

¹⁸ O nome deste *drone* é uma homenagem ao primeiro avião utilizado pelo Correio Aéreo Brasileiro, ou seja, o “Curtiss Fledgling”. O prefixo dessa aeronave era K-263.

O principal componente eletrônico do *drone* é o controlador de voo e estabilidade. Esse controlador, de acordo com as informações recebidas dos seus sensores (i.e. giroscópio, acelerômetro, bússola e altímetro) ou de comandos enviados para o *drone*, aumenta ou diminui a velocidade dos motores para manter a estabilidade do voo ou para executar manobras.

O controlador de voo e estabilidade adotado no UAV K-263 é o HKPilot Mega v.2.5 (Figura 14j) que, por ser um clone do controlador ArduPilotMega (APM), é compatível com o conjunto de sistemas de código-livre denominado Arducopter/ArduPilot. Alguns sensores já estão presentes neste controlador, tais como, giroscópio de 3 eixos, acelerômetro de 6 eixos, bússola digital e barômetro. Para permitir o voo autônomo foi instalado o módulo GPS (Figura 14f).

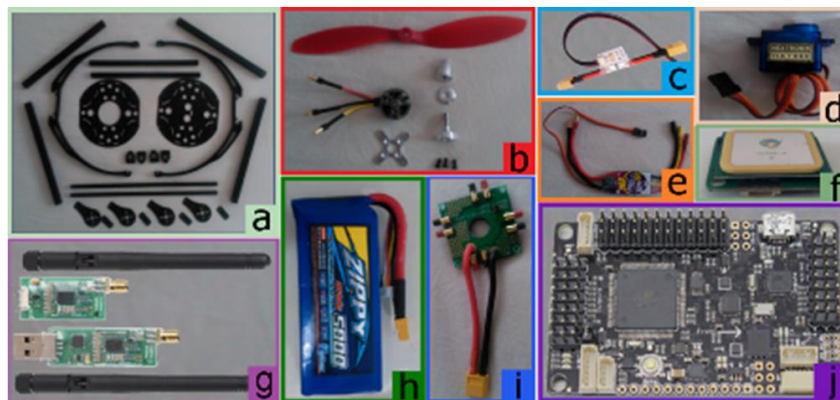


Figura 14. Componentes do PostDrone University UAV K-263 (a) frame (b) motor/hélice (c) módulo de força (d) micro-servo (e) ESC (f) GPS (g) kit de telemetria (h) bateria (i) distribuidor de força (j) HKPilot Mega.

Para completar a eletrônica de controle do *drone* foi instalado o kit de telemetria (Figura 14g), que permite a troca de mensagens entre o UAV K-263 e o PostDrone University MAVLink Communication Server, trocando dados por meio de um link full-duplex utilizando a banda de 915MHz. As características do equipamento de telemetria instalado permitem a troca de dados em um raio de 1,5Km.

Cada um dos quatro conjuntos propulsores do *drone* é formado por motor/hélice (Figura 14b). O motor utilizado é do tipo *brushless* (i.e. motor de corrente contínua sem escovas que, em relação aos motores com escovas, apresenta vantagens como melhor rendimento, menor ruído e maior vida útil) e sua capacidade de rotação é de 1.000 rpm/Volt. Como a bateria utilizada é de 11,1V (Figura 14h), a rotação máxima esperada é de 11.100 rpm, podendo para isso exigir uma corrente de até 15 A. A hélice utilizada tem as seguintes medidas: 9 polegadas de comprimento e 4,7 polegadas de passo (i.e. o ângulo de incidência das pás da hélice em relação à direção do movimento). Segundo dados do fabricante, essa configuração de motor/hélice é capaz de erguer até 632 gramas; como são utilizados quatro conjuntos, estima-se que a capacidade de carga do *drone* seja de aproximadamente 2,5 Kg.

O *drone* que, pela posição dos conjuntos motor/hélice, foi configurado no formato “QUAD X” (i.e. as estruturas de suporte dos conjuntos motor/hélice formam um “X” onde os componentes eletrônicos e a bateria ficam posicionados no centro). Nessa configuração, o UAV torna-se mais estável, uma vez que o *drone* reage aos comandos de maneira mais suave. Os motores *brushless* são síncronos e recebem de um ESC (*Electronic Speed Control*) (Figura 14e) a tensão necessária de trabalho bem como o controle de velocidade. Cada motor necessita de um ESC que deve ser escolhido de acordo com as características do motor, como a tensão de trabalho e a corrente máxima de operação. Para garantir o controle do quadricóptero os motores 1 e 2 devem girar no sentido anti-horário, enquanto os motores 3 e 4 giram no sentido horário. Para isso, deve-se inverter a polaridade dos pinos que ligam os ESCs aos motores 1 e 2 com o intuito de alterar o sentido da rotação. Os ESCs empregados no *drone* são capazes de suportar até o dobro da corrente máxima exigida pelos motores e, segundo dados do fabricante é um dos mais rápidos, operando em 1KHz. Para facilitar a identificação da parte frontal e traseira do UAV K-263 foram utilizadas hélices de cores diferentes (vide Figura 12). As hélices vermelhas são instaladas nos motores 1 e 3 (frente) enquanto que as hélices amarelas são instaladas nos motores 2 e 4.

O fornecimento de energia para a HKPilot Mega pode ser garantido por meio dos pinos de força de um dos ESCs, porém, essa alternativa pode trazer problemas futuros devido às interferências geradas pelo motor ligado ao ESC podendo, inclusive, causar reinícios da HKPilot Mega durante o voo. Para evitar esse problema e diminuir a probabilidade de incidentes graves, optou-se pela utilização de um módulo de força (Figura 14c) que é responsável por filtrar e eliminar ruídos, garantindo, assim, a correta alimentação tanto para os ESCs, como também, para a eletrônica embarcada do *drone*. Esse módulo de força é capaz de suportar tensões de até 18V e corrente de até 90A, garantindo maior confiabilidade e durabilidade à eletrônica embarcada do *drone*. Um conector alimenta a HKPilot Mega e outro conector alimenta a central de distribuição de força (Figura 14i), que distribui eletricidade aos quatro ESCs. Para isso foi necessário ligar apenas os pinos de controle dos ESCs à HKPilot Mega. Os ESCs instalados suportam tensões de até 18V e correntes de até 30A.

Quatro micro-servos (Figura 14d), instalados nos esquis (i.e. tipo de trem de pouso utilizado em helicópteros e aviões) do *drone*, são responsáveis por prender e desprender um bagageiro (na solução do PostDrone University, chamamos esse bagageiro de PostDrone University Delivery Box), ou seja, o compartimento onde os objetos a serem transportados são armazenados. O formato da *delivery box* permite o transporte de pequenos objetos, que durante as entregas fica presa aos esquis do *drone*, sendo liberada automaticamente no heliponto da unidade de destino.

Apesar da HKPilot Mega prover um canal de alimentação compatível com os servos, utilizar essa fonte poderia causar interferências capazes de reiniciar a placa em voo e,

consequentemente, causar um incidente grave. Por isso, os micro-servos recebem a tensão de um dos ESCs instalados e apenas os sinais de controle são enviados pela HKPilot Mega.

Toda a eletrônica é alimentada por uma bateria de Lithium Polymer (LiPO) (Figura 14h), que possui maior densidade de energia em comparação com baterias que utilizam outras químicas, podendo suportar altas taxas de descarga. Baterias de LiPO apresentam limitações e riscos que requerem certos cuidados, tais como, nunca descarregar abaixo de sua carga mínima, pois isso a danificaria completamente, e evitar impactos ou perfurações, uma vez que essas baterias são altamente explosivas.

A estrutura principal do *drone* é chamada de *frame* e é composta por um conjunto de peças estruturais confeccionadas com materiais leves e resistentes como alumínio e fibra de carbono (Figura 14a). A adoção desse *frame*, apesar de apresentar um custo mais elevado que *frames* de outros materiais (e.g. plástico, metal ou madeira), garantiu maior resistência e um menor peso final ao *drone*.

Apesar de todo o controle do *drone* ser feito de forma automatizada pelo sistema ArduCopter/ArduPilot, por questões de segurança e para garantir o controle do UAV caso ocorra alguma pane durante o voo, o *drone* possui um equipamento de rádio controle utilizado por praticantes de aerodelismo, composto por um receptor capaz de receber comandos do rádio transmissor, utilizando a frequência de 2,4 GHz em um raio de até 2Km.

4.3.2 Sistema

PostDrone University System é um sistema multiplataforma, de código-fonte aberto e que é composto de quatro programas, sendo dois servidores (i.e. o servidor do sistema e o servidor de comunicação MAVLink) e dois *softwares* clientes do sistema (i.e. a ferramenta de administração para *desktop*, e o aplicativo para dispositivos móveis). A linguagem de programação Java foi utilizada no desenvolvimento de todos os programas que compõem este sistema. O sistema, ainda, faz uso de dois serviços providos gratuitamente pela Google: (i) o serviço de mapas, i.e. o Google Maps; e (ii) o serviço de notificações, i.e. o Google Cloud Messaging (GCM).

As tecnologias utilizadas na comunicação de dados entre os diferentes componentes da solução PostDrone University são variadas. A transferência de dados entre o servidor de comunicação MAVLink e o UAV K-263 é bidirecional e garantida por meio do equipamento de telemetria instalado no UAV K-263 e no servidor de comunicação MAVLink, ambos utilizam o protocolo de comunicação para pequenos veículos não tripulados para troca de mensagens denominado de Micro Air Vehicle Link (MAVLink). O servidor do sistema envia as mensagens de missões e demais comandos ao servidor de comunicação MAVLink por meio de requisições ao *WebService*, enquanto o servidor de comunicação MAVLink envia os dados telemétricos do UAV K-263 (atitude, altitude, direção, localização e estado da bateria) ao servidor do sistema por meio de *sockets*. A comunicação partindo dos *softwares* clientes, i.e. a ferramenta de administração e

o aplicativo, para o servidor do sistema é feita por requisições ao *WebService*, enquanto a comunicação partindo do servidor do sistema para o aplicativo, utiliza o serviço GCM. Já a comunicação partindo do servidor do sistema para a ferramenta administrativa, utiliza *sockets* para enviar os dados telemétricos recebidos do servidor de comunicação MAVLink. A Figura 15 mostra o diagrama com a arquitetura do sistema.

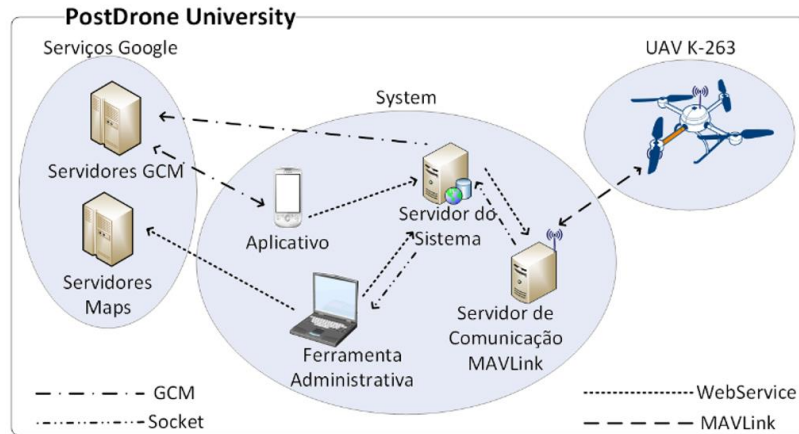


Figura 15. Arquitetura do PostDrone University.

Em relação a essa arquitetura vale comentar que o próprio servidor do sistema poderia enviar as mensagens ao UAV K-263 e, para isso, bastaria que o equipamento de telemetria fosse instalado nesse servidor. Porém, se o servidor do sistema precisar ser instalado longe da área de atuação do *drone*, o alcance da antena poderia não ser suficiente. Assim, optou-se por incluir na arquitetura um servidor que é responsável, exclusivamente, pela comunicação de dados entre o servidor do sistema e o *drone*. Essa arquitetura permite também que outras tecnologias de comunicação diferentes da MAVLink possam ser utilizadas na solução.

4.3.3 Servidor do Sistema

O servidor do sistema foi desenvolvido como um *WebService* do tipo REST (*REpresentational State Transfer*), sendo executado sob Tomcat 8 e os dados do sistema são armazenados em um Banco de Dados (BD) relacional MySQL.

Sempre que um dos clientes de *software* necessita inserir, alterar ou visualizar informações do BD, o *software* cliente envia uma requisição ao servidor do sistema que executa uma consulta ao BD e retorna um JSON (*JavaScript Object Notation*) a ser manipulado pelo *software* cliente requisitante.

Todas as requisições cadastradas pelos usuários são gerenciadas pelo servidor do sistema, que mantém um histórico das alterações ocorridas nas requisições como estados (vide diagrama de estados apresentada na Figura 16). Sempre que um novo estado é inserido para uma requisição o servidor do sistema necessita enviar informações aos usuários envolvidos com a requisição.

Para evitar que o dispositivo móvel verifique periodicamente se existem notificações, optou-se pela utilização do serviço GCM.

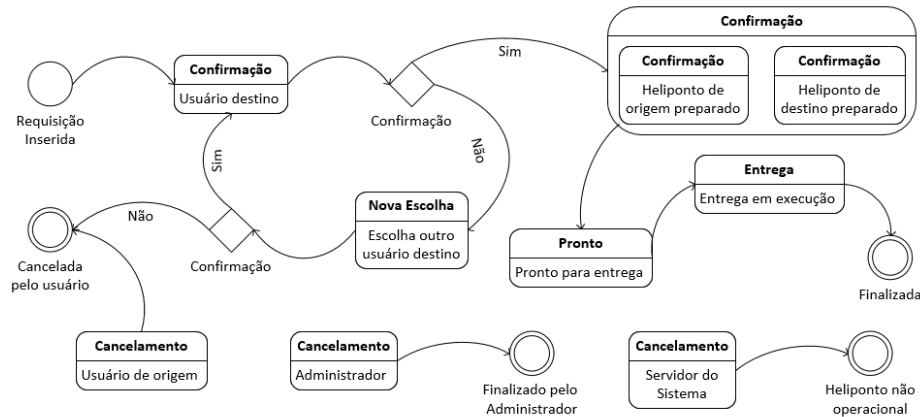


Figura 16. Diagrama de Estados das requisições de entrega do sistema.

Quando o administrador habilita entregas, o servidor do sistema passa a verificar a tabela de estados das requisições para descobrir as requisições que possuem o estado definido como “Pronto para entrega”, e a requisição mais antiga com essa característica é utilizada para montar um arquivo de pontos de parada – *waypoints* – onde o *drone* deve executar as tarefas necessárias à execução da entrega. Os *waypoints* podem conter comandos simples como levantar voo ou prender/desprender a *delivery box*, bem como, comandos mais complexos que dependem de informações de localização como: latitude, longitude e altitude. Em seguida, envia a mensagem ao servidor de comunicação MAVLink. Esse servidor, por sua vez, interpreta o conteúdo do arquivo de *waypoints* (vide exemplo na Figura 17) e, utilizando o protocolo MAVLink, transmite a mensagem ao UAV K-263 para que seja iniciada a entrega. Nesse momento os dispositivos dos usuários envolvidos recebem uma notificação de que a entrega será iniciada.

QGC WPL 110											
0	1	0	16	0	0	0	0	-6.25271	-36.5351	286	1
1	0	3	22	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	15 1
2	0	3	19	10.000000	0.000000	0.000000	0.000000	-6.25217	-36.5355	15	1
3	0	3	21	0.000000	0.000000	0.000000	0.000000	-6.25217	-36.5355	1.000000	1
4	0	3	22	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	7 1
5	0	3	19	10.000000	0.000000	0.000000	0.000000	-6.25236	-36.5349	7	1
6	0	3	21	0.000000	0.000000	0.000000	0.000000	-6.25236	-36.5349	1.000000	1
7	0	3	22	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	15 1
8	0	3	19	10.000000	0.000000	0.000000	0.000000	-6.25271	-36.5351	15	1
9	0	3	21	0.000000	0.000000	0.000000	0.000000	-6.25271	-36.5351	1.000000	1

Figura 17. Exemplo de arquivo de *waypoints*.

A Figura 18 apresenta o fluxo de atividades executadas pelo servidor do sistema para gerenciar as requisições e preparar as entregas.

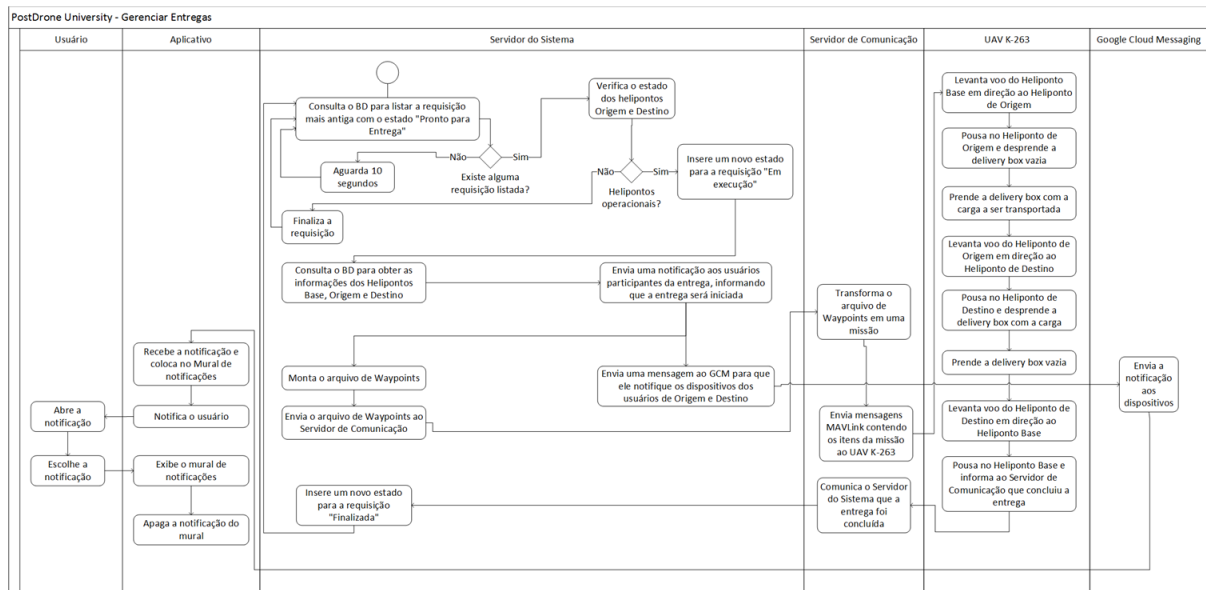


Figura 18. Diagrama de Atividades do gerenciamento de entregas do sistema.

4.3.4 Servidor de Comunicação MAVLink

Para garantir maior escalabilidade à solução PostDrone University, decidiu-se pela utilização de um servidor responsável apenas pela comunicação com o UAV K-263. Essa comunicação é proporcionada pelo equipamento de telemetria, que utiliza o protocolo MAVLink. A escolha de um servidor dedicado apenas à comunicação com o UAV K-263 permite também que o sistema possa trocar informações com outras tecnologias que utilizem protocolos de comunicação diferentes do MAVLink e sistemas de controle de voo e estabilidade diferentes do ArduCopter/ArduPilot, bastando ao servidor de comunicação interpretar o arquivo de *waypoints* recebido do servidor do sistema, escrever os comandos da missão de acordo com as tecnologias utilizadas e enviar ao *drone*.

Quando o servidor do sistema necessita enviar os dados de uma entrega, este servidor (sistema) faz uma requisição ao servidor de comunicação, passando os dados relacionados à entrega, i.e. arquivo de *waypoints*, para que o servidor de comunicação retransmita a mensagem MAVLink ao UAV K-263.

Durante todo o funcionamento do UAV K-263 a HKPilot Mega recebe dados telemétricos de vários sensores. Esses dados são utilizados para manter o voo estável e também enviados ao servidor de comunicação que retransmite ao servidor do sistema. O servidor do sistema armazena essas informações em arquivos de log e também as envia por meio de *sockets* que são consumidos pela ferramenta de administração, que apresenta essas informações em componentes de sua interface. Assim como nas caixas-pretas dos aviões comerciais, os logs armazenados permitem simular na ferramenta de administração as condições de bateria, altitude, posicionamento e atitudes do UAV K-263 durante os voos.

4.3.5 Ferramenta de Administração

A ferramenta de administração do PostDrone University é utilizada pelos administradores do sistema para, por exemplo, gerenciar unidades e seus helipontos, usuários, requisições, acompanhar entregas em andamento e visualizar o histórico de entregas já realizadas. Essa ferramenta também é responsável por exibir em tempo real o comportamento do UAV K-263 durante as entregas, como sua localização, altitude, estado da bateria e atitudes em voo.

As funcionalidades do PostDrone University System são apresentadas no Diagrama de Casos de Uso (Figura 19).

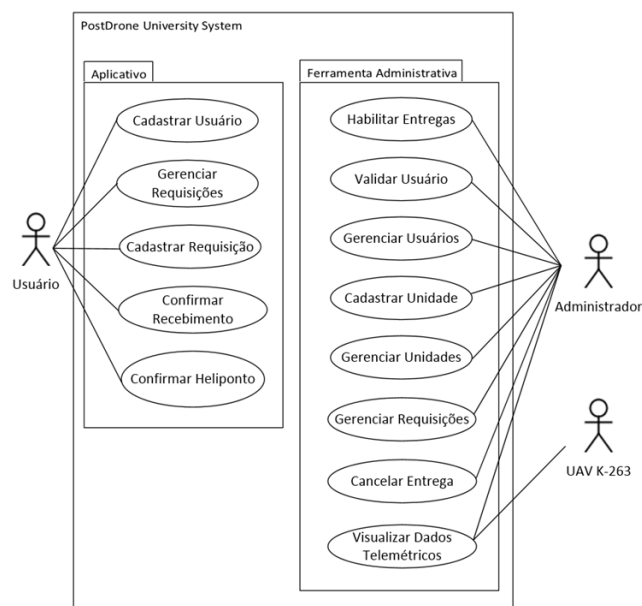


Figura 19. Diagrama de Casos de Uso do sistema.

Para exibir corretamente a localização do UAV K-263 a ferramenta de administração processa as coordenadas obtidas pelo GPS (latitude e longitude) do *drone* e solicita ao serviço de mapas do Google que envie uma imagem contendo um pequeno mapa, tendo como ponto central as coordenadas recebidas do *drone*. Como a imagem do mapa sempre terá como ponto central a posição do UAV K-263, uma imagem de um pequeno quadricóptero é posicionada no centro do mapa representando, em tempo real, a localização e a direção do UAV K-263. Para isso a ferramenta de administração utiliza as informações da bússola instalada no *drone*. O barômetro instalado no UAV K-263 informa a altitude em relação ao nível do mar, a HKPilot Mega define o primeiro valor de altitude recebido como a altitude inicial, com isso consegue subtrair a altitude de voo pela altitude inicial e retornar a “altitude relativa”, que é apresentada numericamente em metros. A atitude do *drone* é identificada como nos painéis de aeronaves, i.e. utilizando um horizonte artificial (i.e. instrumento que mostra ao piloto a posição da aeronave em relação ao

horizonte). Na interface da ferramenta ainda é apresentado graficamente a carga restante da bateria e informações mais detalhadas da bateria, tais como, tensão e corrente elétrica.

A primeira ação a ser realizada pelo administrador do sistema é cadastrar todas as unidades da Universidade e seus respectivos helipontos, iniciando pelo heliponto-base, de onde o *drone* parte para iniciar a entrega e volta ao completar. Para isso é necessário possuir as seguintes informações de cada um dos helipontos: latitude, longitude e altitude em relação ao nível do mar. Devido ao fato de o UAV K-263 não dispor de sensores de impacto e seguir em linha reta entre os helipontos, sempre que uma nova unidade é cadastrada o sistema solicita que o administrador informe a altitude mínima de segurança, ou seja, a altura dos maiores obstáculos entre o heliponto da nova unidade e todos os helipontos já cadastrados (Figura 20). Essa informação é importante para que o servidor do sistema prepare com segurança o arquivo de *waypoints* para todas as entregas. As unidades recém cadastradas, por padrão, não estão operacionais, devendo o administrador definir a unidade como operacional. Caso exista algum problema em um heliponto, o administrador pode definir a unidade como não-operacional. Como pode ser observado no diagrama de atividades da Figura 18, se uma requisição a ser iniciada tiver uma unidade marcada como não-operacional, seja de origem ou destino, o servidor do sistema cancelará tal requisição e notificará os usuários envolvidos.

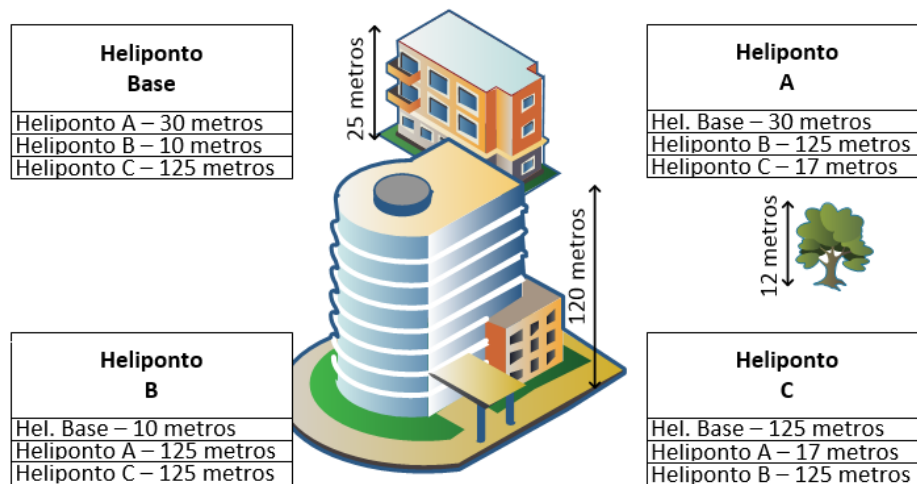


Figura 20. Exemplo da altitude mínima de segurança entre helipontos.

O administrador também é o responsável por gerenciar usuários e requisições. Pode-se, por exemplo, suspender ou reativar um usuário sempre que julgar necessário. As requisições podem ser visualizadas e gerenciadas, podendo o administrador visualizar a requisição em execução e também as requisições na fila, sendo que as últimas, inclusive, podem ser canceladas pelo administrador. Quando isso ocorrer todos os usuários envolvidos na requisição cancelada receberão uma notificação e a requisição receberá um novo estado “Finalizado pelo administrador” (vide Figura 16).

4.3.6 Aplicativo

O *software* para os usuários finais do sistema foi desenvolvido para dispositivos móveis com sistema operacional Android. O aplicativo permite ao usuário cadastrar, acompanhar e gerenciar as requisições de suas entregas. Para reduzir o consumo de bateria, seja por excesso de processamento ou acessos ao servidor do sistema, optou-se pela utilização de requisições ao *WebService* sempre que necessário inserir, alterar ou obter informações no BD do sistema.

O cadastro do usuário é feito no próprio aplicativo (Figura 21a), geralmente quando utilizado pela primeira vez. Porém, mesmo após cadastrado, o usuário ainda não poderá efetuar o *login* (Figura 21b) no sistema, uma vez que todos os usuários recém cadastrados estão inativos até que o administrador ative seus respectivos cadastros.

O cadastro de requisições é um processo que requer a participação de dois usuários, i.e. origem e destino. O usuário solicitante (origem) deve informar a unidade de origem (i.e. de onde partirá a encomenda) e a unidade de destino (i.e. para onde será enviada a encomenda), serão apresentadas apenas as unidades cujos helipontos estejam operacionais. A partir de então, o aplicativo retorna uma lista de usuários da unidade de destino que estão autenticados ao PostDrone University. Após a escolha do usuário de destino e o preenchimento de um breve comentário sobre a entrega é que a requisição pode ser cadastrada. Neste momento o servidor do sistema envia uma notificação ao dispositivo do usuário de destino com o intuito de saber do usuário se ele pode participar da requisição. Caso não possa aceitar a requisição, o servidor do sistema envia uma notificação ao dispositivo do usuário de origem informando-o quanto à negativa e o aplicativo apresentará uma lista com os usuários da unidade de destino que podem ser selecionados para receber a encomenda. Esse processo ocorrerá até que um usuário de destino aceite participar da entrega ou que o usuário solicitante cancele a requisição. A Figura 22 apresenta o diagrama de atividade desta funcionalidade.

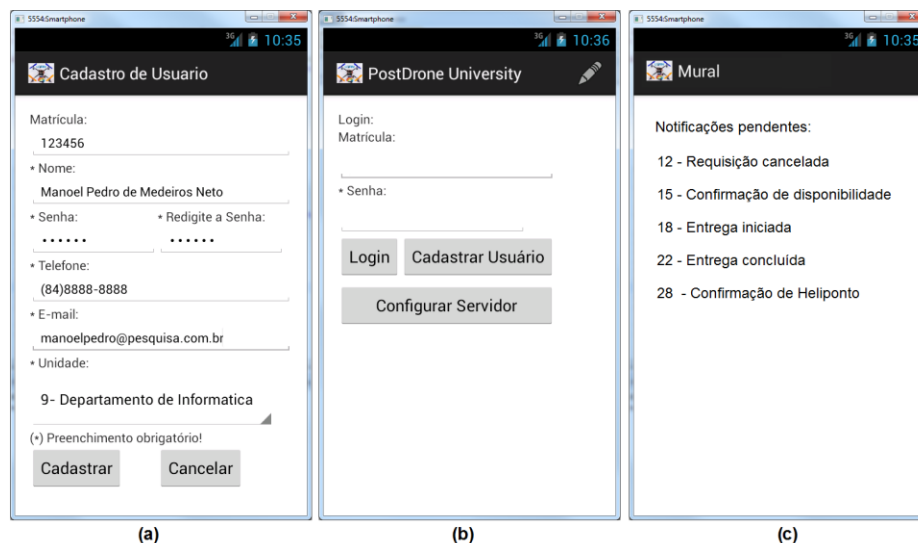


Figura 21. Tela do Aplicativo: a) cadastro de usuário; b) login e c) mural de notificações.

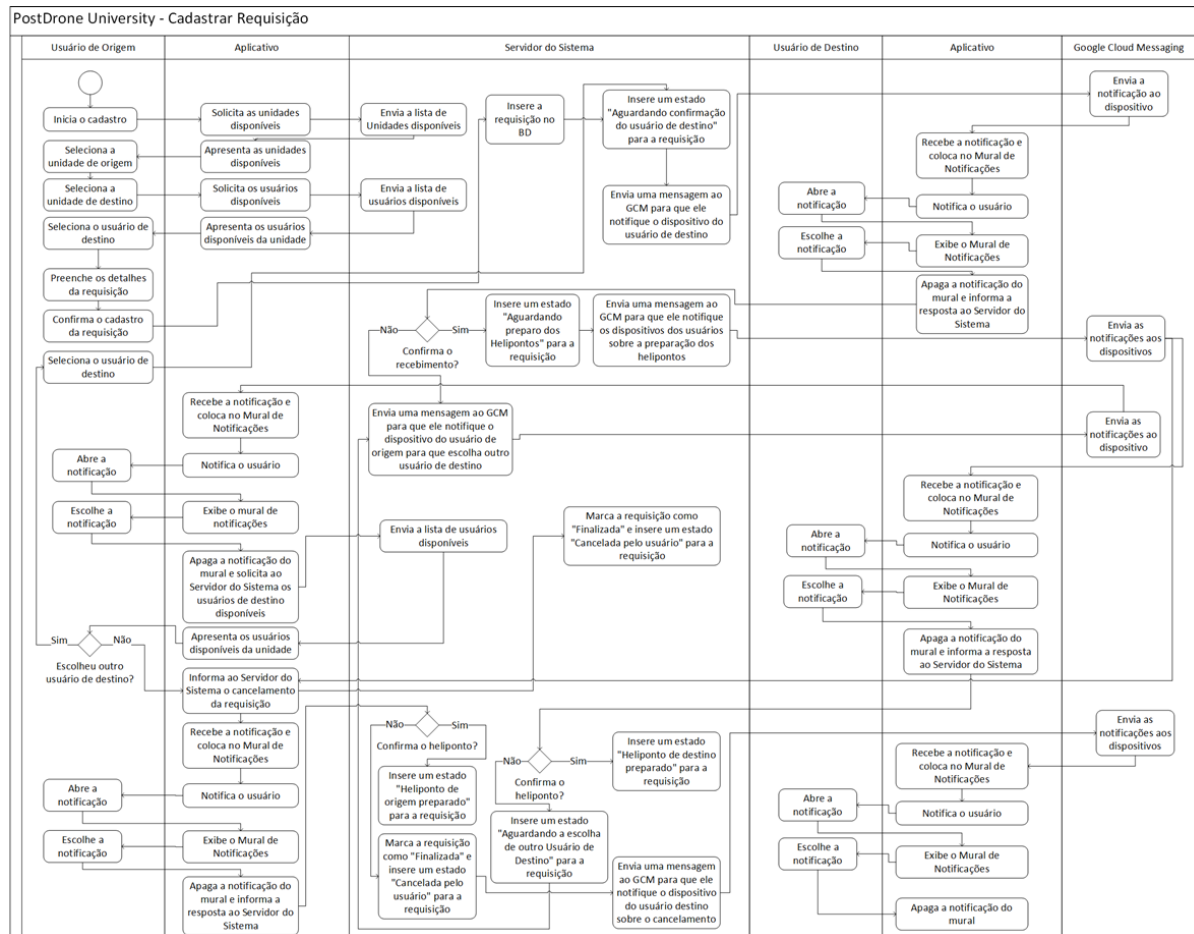


Figura 22. Diagrama de Atividades do cadastramento de nova requisição do sistema.

Toda a comunicação partindo do servidor do sistema para o aplicativo é feita por meio de notificações do serviço GCM. Como o usuário pode receber muitas notificações simultâneas e as notificações recebidas por um aplicativo sobrescrevem as anteriores, é necessário armazená-las assim que chegam ao dispositivo. Para isso foi criado um “Mural” de notificações (Figura 21c), que as armazena até que o usuário selecione uma delas e responda o que for solicitado. Ao término essa notificação será excluída do Mural. O mural é exibido ao selecionar uma notificação recém-chegada ou ao selecionar a opção adequada no aplicativo.

O usuário do sistema pode acompanhar todas as requisições ativas com as quais faça parte como origem ou como destino, entretanto, pode cancelar apenas as requisições onde seja o usuário de origem, desde que esta não possua o estado “Em execução”. Se o estado da requisição for diferente de “Em execução” e o usuário fizer parte dela como destino, ele pode cancelar o recebimento da requisição. Caso isso ocorra o usuário de origem deve escolher outro usuário válido para o recebimento da encomenda ou cancelar a requisição.

Quando o usuário efetua o *logout* do sistema, são canceladas todas as suas requisições e todas as aceitações de recebimento de requisições em que é parte como destino, forçando os usuários que inseriram tais requisições a escolherem outro usuário de destino.

4.4 Testes de Voo do UAV K-263

Durante todo o processo de montagem do *drone* alguns testes foram necessários para garantir a correta instalação dos componentes. Os três primeiros testes foram executados apenas com a HKPilot Mega conectada ao sistema de telemetria, aos quatro motores e seus respectivos ESCs e ao GPS, alimentados por uma bateria de 11,1V.

Inicialmente foi carregado o *firmware* adequado (ArduCopter Quad X v.3.2.1) no UAV K-263 e calibrados todos os seus sensores para, em seguida, testar o funcionamento da HKPilot Mega no controle dos quatro motores. Notou-se que os quatro ESCs necessitam de uma configuração inicial para que possam ser utilizados, essa configuração pode ser executada diretamente em cada ESC utilizando uma interface USB, desenvolvida pelo próprio fabricante do ESC, ligada ao computador, ou segundo o manual do ESC, por meio de comandos enviados pelo sistema de rádio controle. Neste último caso, cada ESC deve ser configurado individualmente, tendo o pino de controle ligado diretamente ao canal do acelerador do receptor do sistema de rádio controle.

Ainda como parte dos testes iniciais, a comunicação por meio do sistema de telemetria foi utilizada para que as informações dos sensores pudessem ser apresentadas no *software* Mission Planner¹⁹. Observou-se que o alcance em ambientes fechados e com a presença de obstáculos é bastante limitado. Por questões de segurança as hélices não devem estar instaladas durante esses testes iniciais.

Após o primeiro voo observou-se que a primeira versão do UAV K-263, além de apresentar desequilíbrio no centro de gravidade, poderia tornar-se instável em voo devido ao posicionamento dos fios e dos ESCs, que por estarem expostos, prejudicavam a aerodinâmica do *drone* e poderiam causar incidentes, caso entrasse em contato com as hélices. Com base nessas observações durante o primeiro voo do *drone* em campo, surgiu a necessidade de reorganizar todos os componentes, inclusive criando extensões para os fios de controle e alimentação dos motores que saem dos ESCs, passando-os pelo interior dos tubos de alumínio até os motores. Uma base de acrílico foi criada para proteger parte da eletrônica e servir de suporte ao GPS, sistema de telemetria e o receptor do sistema de rádio.

A segunda bateria de testes de voo teve o objetivo de verificar a estabilidade em voo controlado remotamente e dois modos de voo autônomos do UAV K-263. Para isso uma chave de três posições do rádio transmissor foi configurada para alternar entre os modos “Stabilize”, “Loitter” e “RTL” (*Return To Launch*). O modo Stabilize mantém o voo do UAV K-263 estável e respondendo aos comandos enviados por meio do sistema de rádio pelo piloto/operador. O modo Loitter quando ativado mantém o UAV K-263 em voo pairado (Figura 23a) no ponto

¹⁹ Estação de Controle em Terra para veículos não tripulados que utilizem a plataforma ArduPilotMega (APM) e similares. O sistema é desenvolvido para Windows e é de código-livre.

exato em que o comando é enviado, mesmo que rajadas de vento forcem o UAV K-263 a mudar de posição, o controlador de voo impede que isso ocorra, isso é possível graças às informações obtidas do barômetro e do GPS; o modo de voo RTL, quando acionado, altera a altitude do UAV K-263 para um valor de segurança definido pelo sistema (por padrão esse valor é de 15 metros podendo ser alterado), em seguida voa em direção ao ponto onde seus motores foram ligados para, então, pousar. Esse recurso é possível pois o sistema grava a posição geográfica do UAV K-263 no momento em que os motores são ligados. A Figura 23b mostra o ponto exato onde o UAV K-263 levantou voo enquanto a Figura 23c apresenta o UAV K-263 após o pouso automatizado pela função RTL. Observa-se nessa segunda bateria de testes que, mesmo sendo pequena a diferença entre o local de decolagem e o de pouso, essa diferença pode prejudicar o processo automatizado de recolhimento e liberação da *delivery box*.



Figura 23. Testes de voo do UAV K-263 (a) modo de voo Loitter (b) take-off (c) pouso autônomo (d) *delivery box* acoplada com carga.

Durante os primeiros voos de testes os dados telemétricos do UAV K-263 foram recebidos corretamente pelo *software* Mission Planner v.1.3.24 instalado em um computador, possibilitando principalmente o acompanhamento do consumo elétrico do UAV K-263. Os testes executados nos micro-servos instalados nos esquis do UAV K-263 (Figura 23d) mostraram que o sistema é capaz de prender e desprender sem problemas a *delivery box*.

Durante os testes de voo do UAV K-263 foi constatada uma pequena diferença de posicionamento, que prejudica a completa automatização das entregas, necessitando a intervenção de um usuário no momento de prender a *delivery box* ao UAV K-263. Apesar disso, todo o controle efetuado pelo servidor do sistema, bem como as funcionalidades do servidor de comunicação, da ferramenta de administração e do aplicativo móvel foram validados.

A execução de todos os testes foi importante para validar cada passo da montagem do UAV K-263, bem como corrigir o posicionamento de alguns componentes visando o melhor equilíbrio do *drone* em voo.

4.5 Discussão

Os testes de voo do *drone* mostram viabilidade na utilização do PostDrone University em conjunto com o sistema ArduCopter/ArduPilot instalado na placa controladora de voo e estabilidade HKPilot Mega. A entrega automatizada efetuada pelo *drone*, prevê que o mesmo levará uma *delivery box* vazia sempre que iniciar uma entrega. Assim, ao iniciar uma entrega o *drone* sairá do heliponto-base levando uma *delivery box* vazia, pousará no heliponto de origem e a desprenderá para, em seguida, prender a *delivery box* com a carga a ser transportada. Ao chegar no heliponto de destino o *drone* deve desprender a *delivery box* com a carga, prender uma *delivery box* vazia e voltar ao heliponto base para aguardar uma nova mensagem de entrega. Dessa forma, o pouso tem que ser executado em um ponto exato do heliponto, porém, a falta de sensores capazes de auxiliar na precisão do pouso prejudica a total automação da entrega e requer a ação humana para a troca da *delivery box*.

A *delivery box* utilizada nesta versão do UAV K-263 é capaz de armazenar pequenos objetos e, por ser um produto comercializado para outros fins, seu formato pouco aerodinâmico pode influenciar negativamente no voo do *drone*. Novos compartimentos de carga com formatos e tamanhos diferentes podem ser desenvolvidos para compor a solução. O desenvolvimento de técnicas que deem a precisão necessária ao *drone* para que possa pousar exatamente no local onde deva prender ou desprender a *delivery box* deve ser levado em conta para pesquisas futuras. Duas alternativas podem ser utilizadas para garantir essa precisão, i.e. construir helipontos capazes de posicionar o UAV K-263 na posição adequada de forma que esse *drone* possa prender ou desprender a *delivery box*, ou acrescentar sensores que possam garantir o pouso no ponto exato do heliponto.

Durante o desenvolvimento deste trabalho surgiu uma nova plataforma de controle de voo e estabilidade denominada Pixhawk que possui recursos de *hardware* superiores à HKPilot Mega e que pode ser utilizada sem adaptações à solução PostDrone University, uma vez que também utiliza o sistema ArduCopter/ArduPilot e a comunicação também se dá por meio do protocolo de comunicação MAVLink. Da mesma forma, outros UAVs de asas rotativas podem ser utilizados nesta solução sem a necessidade de adaptações, permitindo o transporte de objetos com maior volume ou peso, bastando apenas que o controlador de voo e estabilidade utilize o sistema ArduCopter/ArduPilot em conjunto com o protocolo de comunicação MAVLink.

Sensores de obstáculos podem ser instalados na HKPilot Mega, tornando mais seguro o voo em áreas com obstáculos. Entretanto, as funções que podem ser ativadas ao encontrar obstáculos ainda não permitem o voo autônomo em ambientes com obstáculos.

O equipamento de telemetria usado no servidor de comunicação é capaz de garantir a troca de informações com o UAV K-263 em um raio de até 1,5Km. No entanto, outros produtos

podem substituir o equipamento em questão aumentando o alcance nas comunicações, alguns deles garantem a comunicação em ambientes abertos em um raio de até 64Km.

Apesar de já existirem trabalhos visando o desenvolvimento e testes de sistemas de entrega automatizados por meio de UAVs, a liberação para tal atividade depende da regulamentação dos órgãos responsáveis pelo registro de aeronaves/pilotos e também pelo controle de tráfego aéreo em cada localidade. Portanto, a solução apresentada neste trabalho deve receber novos requisitos de acordo com a legislação em vigor na localidade em que for utilizada.

4.6 Conclusão

Apesar do grande número de aplicações emergentes e da presença cada vez maior no uso de *drones* no cotidiano das pessoas e empresas, poucas soluções estão, de fato, voltadas para sistemas de entrega por meio de *drones*, apesar do potencial de produtos e serviços para a indústria de Consumer Electronics.

Neste capítulo apresentamos uma solução capaz de efetuar entregas automatizadas de pequenos objetos utilizando um UAV. O PostDrone University System e o PostDrone University UAV K-263 são, respectivamente, um sistema e um *drone* para serem utilizados no campus de qualquer universidade do mundo. No entanto, com poucas adaptações, essa solução poderia ser utilizada para entrega de objetos em outros contextos de aplicação.

Trabalhos futuros envolvem o aperfeiçoamento dessa solução com o intuito de integrar sistemas PostDrone University instalados em múltiplas instituições, com UAVs de diferentes capacidades de transporte e autonomia de voo. Assim, diversas instituições de ensino ou pesquisa poderiam instalar o PostDrone University em várias localizações de uma cidade e cada um dos sistemas poderia interagir entre si criando, assim, uma rede de *drones* entre diferentes instituições de uma mesma cidade.

4.6.1 Comentários finais

Este capítulo apresentou as etapas do desenvolvimento de um sistema capaz de gerenciar e executar entregas utilizando um UAV. O capítulo seguinte apresentará os resultados dos testes executados em todas as etapas do desenvolvimento dos *softwares*, dos testes que validaram a montagem do UAV responsável pelas entregas e, testes da solução em funcionamento, onde o UAV executa as entregas de modo autônomo.

Capítulo 5

PostDrone University: Resultados Práticos

5.1 Introdução

Os veículos aéreos não tripulados – Unmanned Aerial Vehicles (UAVs) –, também conhecidos como *drones*, são aeronaves operadas remotamente, semiautônomas ou autônomas, que podem ser empregadas para realizar, por exemplo, monitoramento ambiental, filmagens aéreas e, até mesmo, para a entrega de objetos. Por possuírem características de voo que permitem a decolagem e pouso verticais e permanecer em voo pairado, os UAVs de asas rotativas são os mais indicados para os serviços de entrega, principalmente em ambientes onde os espaços para decolagem e pouso sejam pequenos.

Algumas empresas já estudam a utilização de *drones* executando a entrega de objetos, a empresa de comércio eletrônico Amazon desenvolve um sistema, denominado Amazon Prime Air [90], para entrega de produtos em áreas dentro de um raio de 16Km de seus centros de distribuição. Empresas de transporte como FedEx, U7PS e DHL [91] também têm estudado o uso de multirrotores para dinamizar suas entregas. Pesquisas acadêmicas também exploram aplicações deste tipo, como a solução apresenta pela empresa AMP Electric Vehicles em conjunto com a University of Cincinnati Research Institute (UCRI), denominado de HorseFly [92]. Todos os trabalhos citados utilizam *drones* de asas rotativas, em sua maioria com seis motores ou mais.

Nesse contexto, este capítulo apresenta os resultados dos testes executados em um sistema automatizado de entregas utilizando um UAV de asas rotativas do tipo multirotor, onde os usuários finais do sistema, tais como, professores, pesquisadores, analistas, técnicos e alunos, podem utilizar o sistema para enviar objetos entre unidades do campus de uma universidade.

Este capítulo está organizado da seguinte maneira: a Seção 5.2 descreve a solução desenvolvida e seus dois principais componentes, ou seja, o UAV e o sistema de entrega; a Seção

5.3 apresenta a metodologia dos testes executados; a Seção 5.4 apresenta os desafios e discute os resultados dos testes e a Seção 5.5 apresenta as considerações finais.

5.2 PostDrone University

PostDrone University é uma solução de entrega automatizada de pequenos objetos – e.g. documentos, apostilas, livros, correspondências, amostras para análises laboratoriais e componentes eletrônicos – entre unidades do campus de uma universidade por meio de um *drone*.

A solução é composta por um UAV (PostDrone University UAV K-263) e um sistema (PostDrone University System). O sistema, por sua vez, é composto por quatro programas desenvolvidos no decorrer desta pesquisa, i.e. o servidor do sistema (PostDrone University System Server), a ferramenta administrativa do sistema (PostDrone University System Admin), o aplicativo cliente do sistema desenvolvido para dispositivos móveis (PostDrone University Android App) e o servidor de comunicação MAVLink (PostDrone University MAVLink Communication Server), que funciona como uma interface entre o servidor do sistema e o UAV K-263. A Figura 15 mostra o diagrama com a arquitetura do sistema.

Cada unidade do campus de uma universidade deve possuir um heliponto associado e, desse modo, as entregas são realizadas nesses helipontos, não sendo necessário conhecimento prévio da posição geográfica do heliponto por parte dos usuários finais do sistema, visto que os helipontos vinculados às unidades devem ser previamente cadastrados pelo administrador do sistema com seus dados de latitude, longitude, altitude e, ainda, com a altitude mínima de segurança para voos entre os helipontos, garantindo que o UAV K-263 sempre voe mais alto que o maior obstáculo existente entre os helipontos (Figura 20**Erro! Fonte de referência não encontrada.**). Além dos helipontos associados às unidades, também existe um heliponto base, de onde o *drone* parte para executar a entrega e volta ao final da entrega.

O UAV K-263 (Figura 12) é um *drone* de asas rotativas do tipo multirotor, i.e. um quadricóptero, desenvolvido para realizar a entrega de pequenos objetos. Optamos por desenvolver um UAV multirotor para a solução PostDrone University, visto que esse tipo de UAV não precisa de uma área grande para realizar decolagens e pousos e, portanto, parece mais adequado para emprego no campus de universidades, embora outros UAVs de asas rotativas possam ser utilizados no sistema, tais como: helicóptero, tricóptero, hexacóptero ou octacóptero.

O UAV K-263 possui um controlador de voo e estabilidade clone da placa ArduPilot Mega desenvolvida pela empresa 3DR, i.e. HK Pilot Mega, composto de diversos sensores (GPS, barômetro, acelerômetros, inclinômetro e bússola) capazes de manter a estabilidade durante o voo e também responsável pelo controle automatizado das missões que o *drone* é capaz de executar para efetuar as entregas.

A linguagem de programação Java foi utilizada no desenvolvimento de todos os programas que compõem este sistema. A solução, ainda, faz uso de dois serviços providos gratuitamente pela Google:

- i) o serviço de mapas, i.e. o Google Maps, responsável pela imagem do mapa apresentado no PostDrone Admin, que identifica a localização e posição do drone durante o voo;
- ii) o serviço de notificações, i.e. Google Cloud Messaging (GCM), responsável por notificar os usuários do sistema quanto às requisições das quais faça parte.

As tecnologias utilizadas na comunicação de dados entre os diferentes componentes da solução PostDrone University são variadas. A transferência de dados entre o servidor de comunicação MAVLink e o UAV K-263 é bidirecional e garantida por meio do equipamento de telemetria instalado no UAV K-263 e no servidor de comunicação MAVLink, ambos utilizam o protocolo de comunicação para pequenos veículos não tripulados denominado de Micro Air Vehicle Link (MAVLink). O servidor do sistema envia as mensagens de missões e demais comandos ao servidor de comunicação MAVLink por meio de requisições ao *WebService*, enquanto o servidor de comunicação MAVLink envia os dados telemétricos do UAV K-263 (i.e. atitude, altitude, direção, localização e estado da bateria) ao servidor do sistema por meio de *sockets*. A comunicação partindo dos *softwares* clientes, i.e. a ferramenta de administração e o aplicativo móvel, para o servidor do sistema é feita por requisições ao *WebService*, enquanto a comunicação partindo do servidor do sistema para o aplicativo, utiliza o serviço GCM. Já a comunicação partindo do servidor do sistema para a ferramenta administrativa, utiliza *sockets* para enviar os dados telemétricos recebidos do servidor de comunicação MAVLink.

5.3 Metodologia

Os testes da solução apresentados na Tabela 5 serão aplicados de três formas: testes no PostDrone System, para garantir que o sistema está funcionando corretamente na gestão de entregas e que a comunicação entre os componentes está ocorrendo sem problemas; testes de montagem do *drone* e de execução autônoma de tarefas; e testes de integração, quando a solução será posta efetivamente em prática, com usuários utilizando o aplicativo móvel para cadastrar suas requisições de entrega e o PostDrone System enviando as missões para que o UAV K-263 possa iniciar as entregas.

Tabela 5. Testes executados.

Testes	Objetivos
PostDrone App	Validar o funcionamento do aplicativo e testar o envio de notificações.
PostDrone Communication Server	Validar a comunicação bidirecional entre o servidor e o <i>drone</i> , bem como envio de missões MAVLink.
PostDrone Admin	Validar a exibição dos dados telemétricos recebidos.
PostDrone System	Validar o funcionamento de todos os componentes do sistema.
Testes de Telemetria	Aferir a distância máxima de comunicação entre os módulos de telemetria.
Voos iniciais com <i>drone</i>	Validar a montagem e ajustes iniciais no <i>drone</i> e na APM.
Primeiros voos autônomos	Verificar o comportamento do <i>drone</i> em dois modos de voo autônomos.
Pousos autônomos	Aferir a precisão do <i>drone</i> ao pousar nos helipontos.
Voos com carga	Determinar a capacidade máxima de transporte do <i>drone</i> .
Testes de integração	Validar as funcionalidades dos sistemas e a integração destes com o <i>drone</i> .

5.3.1 Testes de *Software*

Para validar as funcionalidades do aplicativo móvel foram executados testes envolvendo apenas o servidor do sistema e o aplicativo móvel instalado em dois aparelhos celulares e um dispositivo virtual em execução no computador. Durante os testes, três usuários cadastravam requisições de entrega, enquanto os dados eram conferidos no BD, com o intuito de garantir que todos os estados das requisições estavam corretos.

A comunicação bidirecional entre o servidor de comunicação e o *drone* é garantida pelo equipamento de telemetria instalado em ambos. Para iniciar a transferência de dados o servidor de comunicação deve estabelecer uma conexão serial com o *drone* e tratar as mensagens recebidas, para transmitir ao servidor do sistema via *sockets*, neste teste os dados recebidos pelo servidor de comunicação são tratados e gravados em arquivos para análise posterior. Com a troca de informações entre o *drone* e o servidor de comunicação funcionando corretamente, foi possível testar a apresentação dos dados telemétricos do *drone* na interface do aplicativo administrativo. Testes nas funcionalidades do sistema administrativo (gerenciar unidades, helipontos, usuários e entregas) também foram efetuados.

Para permitir a correta execução dos testes em todos os componentes de *software* da solução, uma função que simula o funcionamento do *drone* executando as entregas foi utilizada, esta função recebe as informações da entrega que deve ser iniciada, e, após um tempo determinado informa ao sistema que a entrega foi efetuada. O sistema foi mantido em funcionamento por um intervalo de seis horas, com quatro usuários cadastrando suas requisições e acompanhando todo o processo, validando a utilização do sistema. O sistema também efetuou corretamente o gerenciamento das unidades, helipontos, usuários e das requisições cadastradas, inclusive com vários cancelamentos requisitados pelos usuários ou pelo administrador.

5.3.2 Testes de Voo

Antes de iniciar os testes de voo o sistema de telemetria deveria ser testado, uma vez que a transferência de dados por este meio apresentou diversos problemas, especialmente no envio de comandos ao *drone*. Inicialmente o aplicativo responsável por configurar os módulos (ar e terra) foi utilizado para analisar graficamente a qualidade do sinal enquanto o *drone* é afastado da estação de controle até que a perda de sinal ocorresse. Os testes seguintes são executados com o *software* Mission Planner, onde é possível acompanhar os dados telemétricos do *drone* durante o afastamento de sua estação de controle.

Durante todo o processo de montagem do *drone* alguns testes foram necessários para garantir a correta instalação dos componentes e o bom funcionamento do equipamento, bem como determinar o melhor posicionamento dos componentes, garantindo que o voo seja o mais estável quanto possível. Estes testes consistiam de levantar voo com o *drone* usando o modo de voo “Stabilize”, que mantém o *drone* em voo estável facilitando o controle do mesmo por meio de um sistema de rádio transmissão comum na prática do aeromodelismo, e manter o *drone* em voo pairado para verificar o comportamento do mesmo, ao final pousar o *drone* em segurança.

Os resultados do teste anterior levaram à conclusão que o *drone* não está corretamente balanceado e a disposição de alguns componentes está colocando o voo em risco, após as correções de montagem que mitigam tais riscos, novas sessões de testes de voo foram executadas.

Para executar os primeiros voos autônomos do *drone*, dois modos de voo autônomos foram testados, i.e. Loiter e RTL, o modo Loiter quando ativado mantém o UAV K-263 em voo pairado no ponto exato em que o comando é enviado, e, mesmo que rajadas de vento forcem o UAV K-263 a mudar de posição, a HK Pilot Mega impede que isso ocorra, isso é possível graças às informações obtidas dos sensores (barômetro, acelerômetro, inclinômetro e do GPS) instalados na HK Pilot Mega; o modo de voo RTL (*Return to Launch*), quando acionado, altera a altitude do UAV K-263 para um valor de segurança definido pelo sistema, por padrão esse valor é de 15 metros, voa em direção às coordenadas do ponto onde seus motores foram ligados para em seguida pousar com segurança, esse recurso é possível pois o sistema grava a posição geográfica do UAV K-263 no momento em que os motores são ligados. Por questão de segurança, esse parâmetro do modo de voo RTL deve ser superior à altura do obstáculo mais alto existente no ambiente onde o *drone* será utilizado, se tomarmos como exemplo o ambiente da Figura 20, esse valor deveria ser de no mínimo 125 metros.

Com o *drone* equilibrado e executando voos em modo autônomo de modo satisfatório, foram feitos testes de voos totalmente autônomos, onde o *drone* deveria levantar voo, executar alguma tarefa e pousar em um ponto determinado. Durante estes testes foi utilizado o *software* Mission Planner para o envio das missões e acompanhamento dos voos. As missões consistiam dos comandos apresentados na Tabela 6.

Tabela 6. Comandos das Missões.

Comando	Função	Parâmetros
Takeoff	Levantar voo	Altitude
Loiter_Time	Permanecer em voo pairado em um ponto definido por um determinado tempo	Latitude, longitude, altitude e tempo (Segundos)
Land	Pousar em um ponto definido	Latitude, longitude e altitude
RTL	Coloca o <i>drone</i> em uma altitude definida previamente e voa em direção às coordenadas do ponto onde os motores foram ligados para em seguida pousar	Não existem parâmetros para este comando

Foram utilizadas placas de madeira medindo 2,4 m x 2,1 m, como helipontos, estas placas foram colocadas aleatoriamente em um campo de futebol e suas coordenadas foram utilizadas para o cadastro dos helipontos por meio do aplicativo administrativo.

Para testar as missões geradas pelo PostDrone System, um arquivo de waypoints²⁰ contendo as partes de uma missão (Figura 17) foi gerado pelo PostDrone System para em seguida ser importado pelo *software* Mission Planner, a importação não apresentou problemas e a missão foi gravada na memória da HK Pilot Mega, a Figura 24 apresenta as etapas da missão gerada.

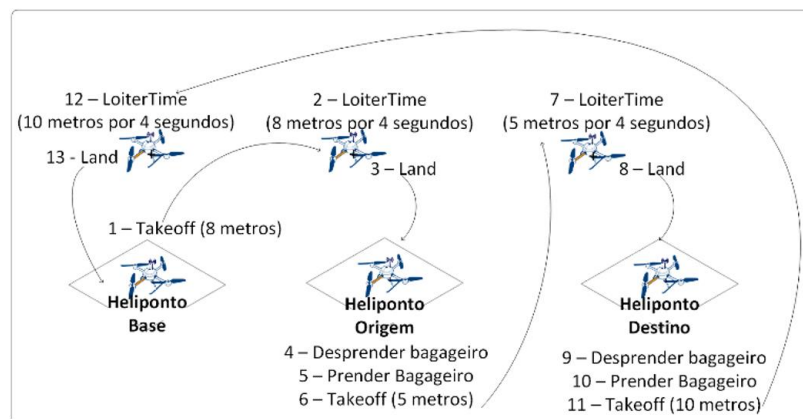


Figura 24. Passos para executar uma entrega.

Inicialmente os passos 2, 7 e 12 não eram utilizados e os voos apresentaram um comportamento perigoso, uma vez que o *drone* levantava voo até a altitude determinada pelo comando Takeoff para em seguida voar em trajetória descendente até o local de pouso determinado, essa trajetória não respeitava o limite mínimo de altitude entre os helipontos e o *drone* sempre executa o pouso mantendo o movimento horizontal, o que causa o capotamento do mesmo. Com a adição do comando “Loiter_time” entre cada decolagem e pouso, o *drone* segue em linha reta, mantendo a altitude determinada nos comandos Takeoff e Loiter_time e somente após 4 segundos posicionado sobre o heliponto, o *drone* inicia o processo de pouso, as posições dos helipontos podem ser observadas na Figura 26a, enquanto os deslocamentos do *drone* podem

²⁰ Arquivo de Waypoints: é um arquivo de texto puro contendo as Ids dos comandos e seus respectivos parâmetros, separados por tabulações, esse arquivo deve ser interpretado, transformado em comandos MAVLink que serão enviados ao *Drone*.

ser observados na Figura 26b, Figura 26c e Figura 26d. Pode-se observar na Figura 25 que o *drone* levanta voo do heliponto base até atingir oito metros de altitude, durante o deslocamento para o heliponto de origem ele mantém a altitude em 8 metros para em seguida pousar no heliponto de origem; permanece pousado por um tempo e levanta voo até alcançar a altitude determinada de cinco metros, mantendo essa altitude enquanto desloca-se para o heliponto de destino e executa o pouso; após alguns segundos levanta voo até alcançar 10 metros de altitude e mantém-se nesta altitude durante o deslocamento até o heliponto base, quando pouisa e termina a missão.

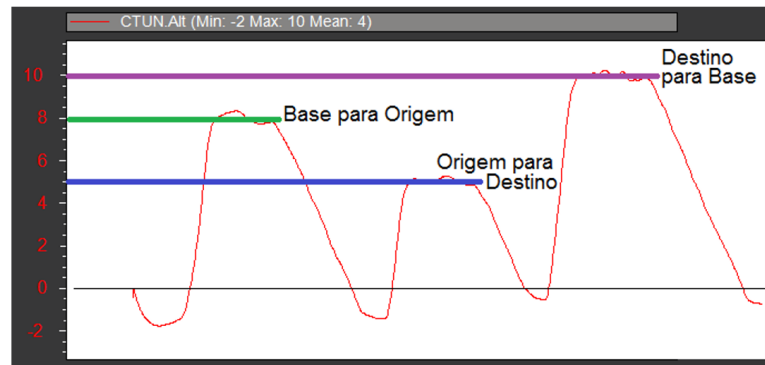


Figura 25. Altitude do *drone* durante a execução de uma missão autônoma.

Esta missão foi utilizada em várias sessões de testes, com objetivos diferentes, como: validar a execução da missão, aferir a precisão no pouso e também obter dados relacionados ao consumo de energia e conseqüentemente à autonomia do *drone*.

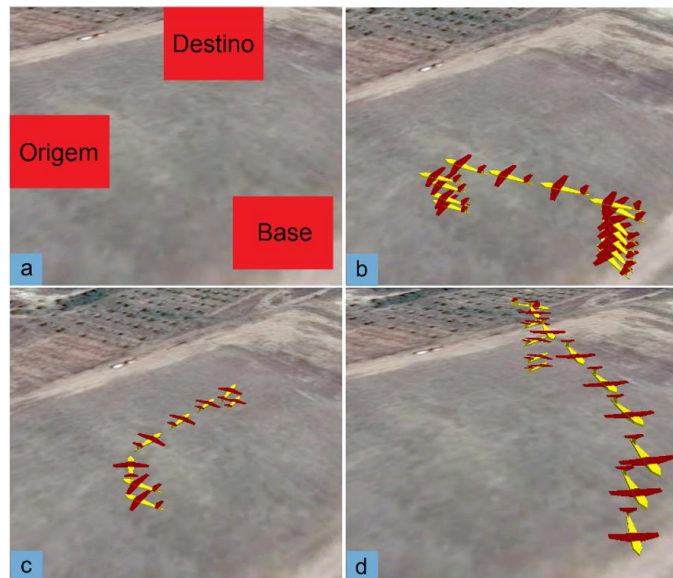


Figura 26. Detalhes dos testes de entrega automatizadas. a) Helipontos; b) Base para Origem – 28 metros; c) Origem para Destino – 33 metros; d) Destino para Base – 43 metros.

5.3.3 Testes de Integração

O teste final de uso é o teste de integração, que, diferente dos testes anteriores no Sistema, utiliza o *drone* para executar as entregas. Para a execução do teste o sistema já possui uma série de informações cadastradas (unidades, helipontos e usuários) e os usuários utilizam a solução integralmente, desde o cadastro de requisições por meio de dispositivos móveis, ao recebimento da entrega executada pelo *drone*. O ambiente escolhido é um campo de futebol com cinco helipontos espalhados pelo mesmo, sendo um deles o heliponto base. Quatro usuários são os responsáveis pelos cadastros das requisições durante os testes. Para executar este teste a função que simulava o *drone* foi desativada, e o servidor de comunicação passou a enviar a missão ao *drone*, por meio do protocolo MAVLink, que em seguida iniciará a entrega.

5.4 Resultados

Apesar do bom funcionamento na execução dos testes no servidor do sistema e no aplicativo móvel, surgiu um problema não detectado durante a modelagem do sistema: o aplicativo móvel armazena apenas a última notificação recebida, desta forma, as notificações não abertas pelo usuário eram substituídas pelas notificações mais recentes, interrompendo o fluxo de estados para as requisições relacionadas às notificações perdidas, com isso, tais requisições jamais estarão aptas a iniciar uma entrega. Para resolver o problema foi criado um “Mural de Notificações” que armazena todas as notificações recebidas até que o usuário decida o que fazer para cada uma delas. O serviço de notificações do Google apresentou o resultado esperado, efetuando a entrega das notificações poucos segundos após o servidor do sistema solicitar.

A arquitetura inicial do sistema não previa a utilização de um servidor apenas para a comunicação com *drone*, essa tarefa era executada pelo servidor do sistema, entretanto, a instalação do equipamento de telemetria em um servidor seria inviável em ambientes de produção, onde os servidores são instalados em um *Data Center*. A solução encontrada foi a criação de um servidor responsável apenas pela comunicação com o *drone*, o que trouxe maior flexibilidade ao sistema, uma vez que a solução agora poderia receber *drones* que utilizem protocolos de comunicação diferentes do MAVLink, bastando implementar os comandos relacionados ao novo protocolo suportado no servidor de comunicação. A comunicação entre o *drone* e o servidor de comunicação apresentou os resultados esperados quanto ao tratamento dos dados recebidos e a transmissão via *sockets* ao servidor do sistema, porém, obtivemos números ruins quanto à distância máxima para transmissão dos dados, esse problema está diretamente relacionado à baixa qualidade do equipamento de telemetria. Durante os testes iniciais apenas o recebimento dos dados telemétricos funcionava, após diversos testes com o equipamento de

telemetria, constatou-se que o fabricante inverteu dois dos cinco pinos de comunicação que ligam o módulo de telemetria à HK Pilot Mega (pinos Out e GND), após corrigir tais pinos o sistema passou a enviar comandos e missões ao *drone* corretamente, entretanto, a perda de sinal a poucos metros de distância não foi resolvida (Figura 27), concluímos que o equipamento de telemetria poderia ter outros defeitos não detectados.

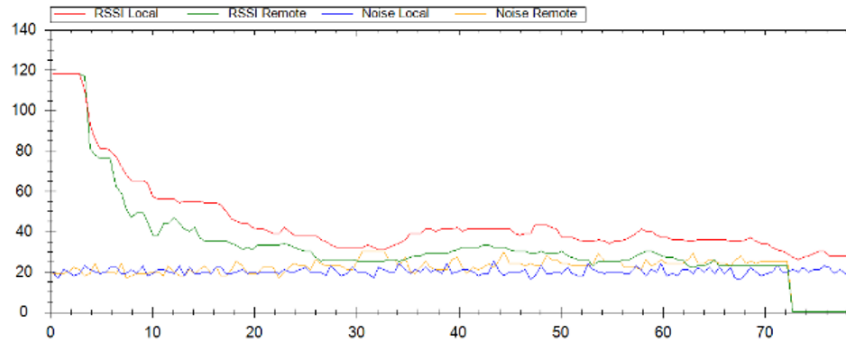


Figura 27. Gráfico de qualidade do sinal nos módulos de telemetria.

A exibição dos dados telemétricos no aplicativo administrativo²¹ representou corretamente as atitudes do *drone* (Figura 28), porém, como a HK Pilot Mega envia as informações telemétricas relacionadas à localização do *drone* cinco vezes a cada segundo, a requisição de uma nova imagem do mapa ao serviço de mapas do Google nesta frequência resulta em negação de serviço por excesso de requisições, para evitar isso foi inserida uma função que analisa se as novas coordenadas recebidas diferem das coordenadas obtidas nos segundos anteriores em pelo menos um metro, apenas quando essa condição é satisfeita que uma nova imagem do mapa é solicitada.

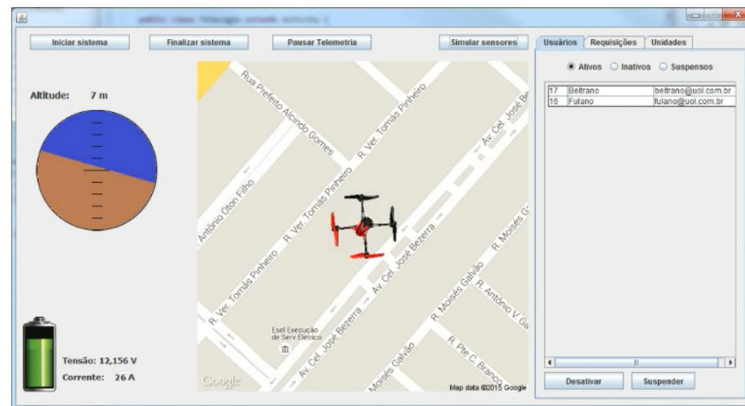


Figura 28. Tela da ferramenta administrativa em funcionamento.

Ainda como parte dos testes de exibição dos dados telemétricos, um arquivo de LOG é criado ao iniciar cada voo, armazenando todos os dados recebidos do *drone* durante o voo, assim, caso o administrador queira visualizar os acontecimentos de uma entrega, basta carregar esse arquivo na ferramenta administrativa, que serão apresentadas informações daquele voo e, assim

²¹ Vídeo do teste: <https://youtu.be/gfieGsnUtqM>

como nas caixas-pretas dos aviões comerciais, auxiliar na obtenção de informações que expliquem possíveis incidentes e acidentes.

O comportamento do sistema simulando a solução em funcionamento não apresentou problemas durante todo o teste, validando a comunicação entre os componentes de *software* da solução e também com os serviços do Google utilizados. As informações armazenadas no BD demonstram que o servidor do sistema fez o correto gerenciamento das requisições, inclusive quanto às requisições canceladas, seja pelos usuários envolvidos ou pelo administrador do sistema.

Durante os primeiros voos, com o intuito de validar a montagem do *drone*, observou-se que o voo nivelado exigia maior força por parte de dois motores, mesmo em situações de pouca incidência de vento, demonstrando que o *drone* não estava devidamente equilibrado. Outro detalhe de montagem que chamou a atenção foram os fios soltos (Figura 29) que poderiam prejudicar os voos ou causar um acidente caso entrassem em contato com as hélices. A solução encontrada para minimizar esses efeitos indesejados foi uma melhor organização dos componentes, equilibrando o peso dos mesmos próximo ao centro de gravidade do *drone* e mantendo todos os fios resguardados no interior dos suportes dos motores e sob uma cobertura de acrílico (Figura 12), que protege a eletrônica embarcada e serve de apoio para componentes como: GPS, módulo de telemetria e o receptor de sinais de rádio usado para o controle do *drone*.



Figura 29. Primeira versão do *drone*.

Os primeiros testes com missões autônomas, enviadas pelo Mission Planner, validaram os recursos presentes da HK Pilot Mega, capazes de controlar o *drone* em voo e também de executar tarefas de modo autônomo. Nestes testes, deve-se observar principalmente o momento do pouso autônomo, executado após a ativação do modo de voo RTL, uma pequena diferença entre o local da decolagem (Figura 30a) e o ponto exato do pouso (Figura 30b), apesar de pequena, essa diferença²² pode prejudicar o processo automatizado responsável por prender e desprender a carga transportada pelo UAV K-263.

²² Vídeo do teste: <https://www.youtube.com/watch?v=Ty1jKE2dVq0>

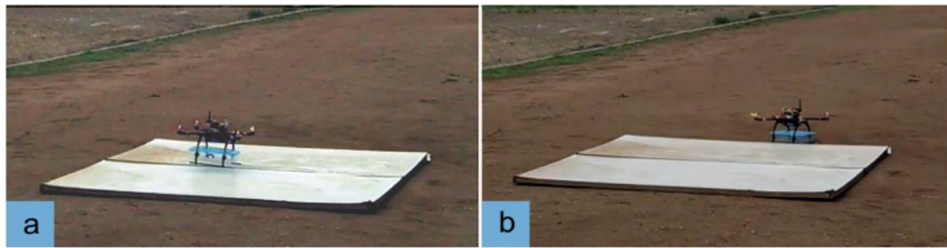


Figura 30. a) Pousou autônomo no heliponto; b) pousou autônomo fora do heliponto.

Durante uma segunda etapa de voos autônomos o *drone* passou a voar em direções diferentes das determinadas na missão, até mesmo o comando RTL levava o *drone* para direções não esperadas. Várias sessões de testes, totalizando 36 minutos de voo, foram necessárias até que o problema fosse encontrado e solucionado. A bateria do GPS estava totalmente descarregada, fazendo com que o GPS assumisse as configurações de fábrica e enviasse dados incompatíveis com a HK Pilot Mega. Após a correção deste problema as novas sessões de testes não apresentaram problemas de posicionamento.



Figura 31. *Drone* em missão autônoma, executando o comando Loiter_time sobre um dos helipontos.

A primeira série de testes com missões autônomas, totalizou oito entregas, estes voos foram importantes para aferir a precisão do *drone* ao pousar nos helipontos, a Figura 31 apresenta o *drone* em Loiter_time, preparando para o pouso no heliponto. Como cada entrega exige que sejam executados três pousos, houve um total de 24 pousos. Em 19 deles o *drone* pousou corretamente dentro da área do heliponto (2,4 m x 2,1 m) e em cinco casos o *drone* pousou fora do heliponto, com distância nunca superior a 20 cm. Durante esta sessão de voos a quantidade de satélites detectados variou, tendo como número mínimo detectado de 10 satélites e máximo de 12 satélites, quanto à precisão do GPS (HDOP – *Horizontal Dilution of Precision*) os dados obtidos como mínima, média e máxima foram de: 1,56m, 1,38m e 1,24m respectivamente.

Em virtude da margem de erro no pouso (Figura 30b), o processo automatizado responsável por prender e desprender o bagageiro do *drone* depende de tecnologias que garantam o pouso na área determinada para troca automática dos bagageiros, como essa funcionalidade

ainda não foi implementada, optou-se por utilizar quatro pequenos servos motores que recebem comandos da HK Pilot Mega para prender ou desprender o bagageiro e o usuário efetua a troca dos bagageiros. Por questão de segurança, o mais adequado é que o *drone* pouse no heliponto, desprenda o bagageiro e permaneça por um tempo determinado sem que os motores estejam em funcionamento, enquanto isso o usuário retira o bagageiro recebido e coloca o novo bagageiro. A alternativa encontrada é inserir o comando “DoServoRepeat” após a execução de cada comando “Land”. Esse comando tem como parâmetros a quantidade e sentido do movimento do servo para desprender o bagageiro, e um tempo para que o servo volte à posição anterior prendendo o bagageiro, nos testes o tempo determinado foi de 60 segundos. Após repetidos testes o *drone* pousava e após poucos segundos levantava voo novamente em direção ao próximo destino, não respeitando os 60 segundos determinados no comando DoServoRepeat. Após a leitura de documentos técnicos relacionados ao protocolo de comunicação MAVLink²³, ficou claro a impossibilidade de parar os motores do *drone* durante a execução da missão, isso ocorre porque existem três tipos de comandos no protocolo MAVLink:

- Comandos de navegação: responsáveis por controlar o voo do *drone*;
- Comandos de ação: são ordens para que o *drone* execute alguma ação durante o voo, como por exemplo iniciar uma filmagem;
- Comandos condicionais: permitem criar condições para iniciar ou parar os comandos de ação.

Os únicos comandos capazes de controlar o voo do *drone* são os comandos de navegação, desta forma, os comandos de ação ou condicionais não afetam a execução da missão. Por isso, nos testes executados, após o servo desprender o bagageiro o *drone* levanta voo e 60 segundos depois, mesmo em voo, o *drone* movimentava novamente os servos para prender o bagageiro. Várias alternativas foram testadas, porém, em nenhuma delas os motores deixaram de girar, oferecendo riscos durante a troca de bagageiros.

Durante os testes de voo autônomo o *drone* passou a apresentar problemas entrando em modo “FailSafe” e pousando rapidamente, algumas vezes causando acidentes²⁴ que danificaram componentes do *drone*, como: hélices, base de acrílico, braço de servo, estrutura de alumínio e trem de pouso. Após a leitura dos vários arquivos de LOG observou-se que o sistema embarcado encontrava discrepâncias entre os valores de altitude obtidos pelo barômetro e pelo GPS (áreas circuladas de vermelho da Figura 32) e, por questão de segurança ele enviava o comando de pouso.

²³ http://planner.ardupilot.com/wiki/common-mavlink-mission-command-messages-mav_cmd/

²⁴ Vídeo do teste: <https://youtu.be/RxAbU6NVUMQ>

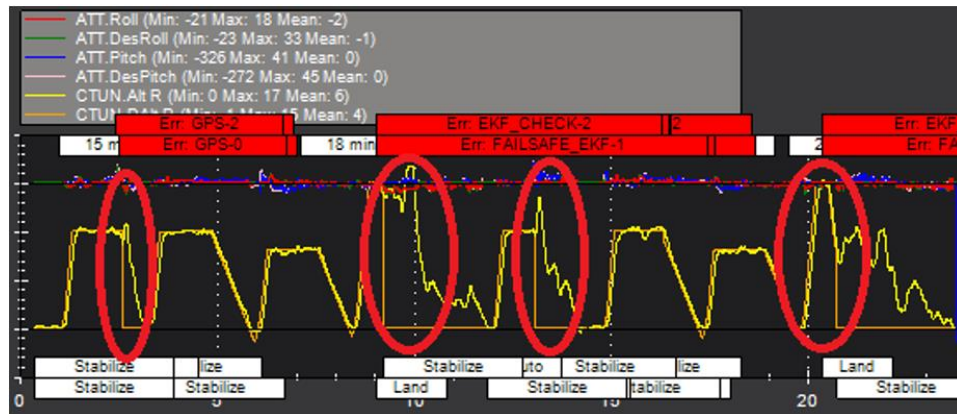


Figura 32. Discrepâncias entre os valores do barômetro e do GPS.

Para solucionar este problema foi efetuada uma atualização do firmware da HK Pilot Mega, durante este procedimento o *bootloader* da placa foi danificado, fazendo com que a HK Pilot Mega deixasse de funcionar. A solução adotada para resolver o problema com o *bootloader* e continuar com os testes de voo foi a troca do microcontrolador ATmega 2650 da placa HK Pilot Mega. Após a substituição do ATmega 2650, testes de voo não programados foram executados para garantir que esse procedimento não danificou a placa de controle, durante estes testes observou-se que o problema com o barômetro persistiu. Durante as análises dos arquivos de LOG ficou claro que um filtro EKF (*Extended Kalman Filter*) detecta a diferença entre a altitude informada pelo barômetro e a altitude informada pelo GPS, e por isso determina o pouso do *drone*. A solução encontrada para que os testes pudessem ser executados foi reduzir gradativamente a precisão deste filtro até que o modo *FailSafe* não seja ativado e, após cada missão ser finalizada desligar a bateria do *drone*, para que a nova missão seja iniciada com a medição correta do barômetro.

Após os testes de carga, onde foram feitas oito entregas com cargas de pesos variados, essa versão do UAV K-263, apresentou uma média de consumo total de 1,2A para executar uma missão em tempo médio de 2 minutos e três segundos, ver Tabela 7.

Tabela 7. Consumo do *drone* com diferentes cargas.

Peso do <i>Drone</i>	Tempo de execução	Corrente Consumida
Sem Bagageiro	2:13	1,09 A
Sem carga	2:08	1,29 A
100 g	1:56	1,35 A
150 g	1:56	1,31 A
200 g	Não concluiu	1,1 A

Durante estes testes o céu permaneceu nublado, o que interferiu negativamente nos dados obtidos pelo GPS, apresentando número mínimo de oito satélites e máximo de 12 satélites, o que resultou em valores de HDOP mínimo, médio e máximo de: 1,98, 1,45m e 1,38m respectivamente, tais valores fizeram com que o *drone* pousasse sempre fora do heliponto a uma

distância nunca superior a 30 cm. Em relação ao comportamento de voo não houve alteração e o *drone* efetuou as entregas sem apresentar problemas relacionados ao padrão de voo esperado.

O *drone* com sua bateria (12V e 5A) e o bagageiro instalado pesam 1.782 gramas, após avaliar os dados dos voos com essa configuração, estima-se que o *drone* seja capaz de executar até quatro entregas para cada bateria carregada. A expectativa inicial de carga máxima foi estimada seguindo as informações do fabricante do motor, que afirma ser de 632 gramas por motor se utilizada uma hélice de 9 x 4,7 e que o motor seja alimentado por uma bateria de três células (11,1 V), para isso seria necessário uma corrente máxima de 10,2A, porém, a carga útil máxima conseguida nos testes com a execução completa de uma entrega foi de 150 gramas com picos de corrente na ordem de 14A por motor, ou seja, o *drone* com a carga útil pesou 1.932 gramas (483 gramas por motor).

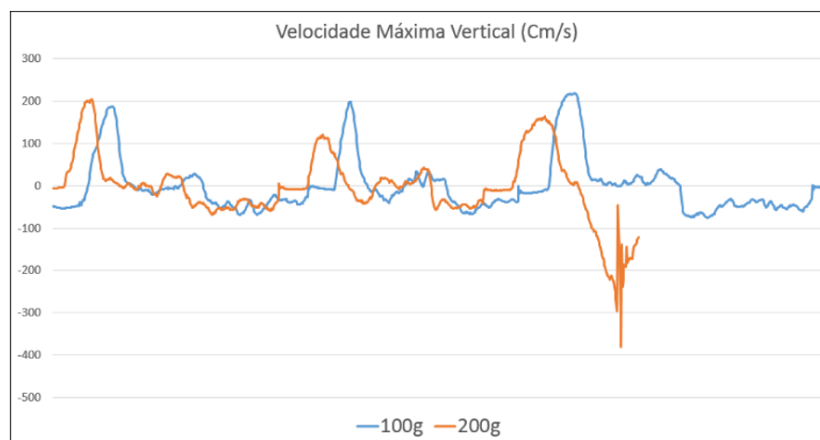


Figura 33. Gráfico de velocidade vertical durante acidente com carga de 200g.

Nos três testes com carga útil de 200 g (495,5 gramas por motor) o *drone* não concluiu a entrega, efetuando sempre um pouso forçado²⁵ entre o heliponto de destino e o heliponto base, alguns deles danificando partes do *drone*. Durante a queda o *drone* apresentou velocidade vertical descendente de até 3,8 metros por segundo, como apresentado no gráfico da Figura 33, enquanto a velocidade vertical descendente máxima de entregas onde não ocorreram acidentes é de 75 cm/s. A Figura 34 apresenta as velocidades horizontais obtidas com as cargas testadas, vale observar apenas que em nenhum dos três voos com carga de 200g o *drone* conseguiu concluir a entrega.

²⁵ Vídeo do teste: <https://youtu.be/2hxVOralkBg>

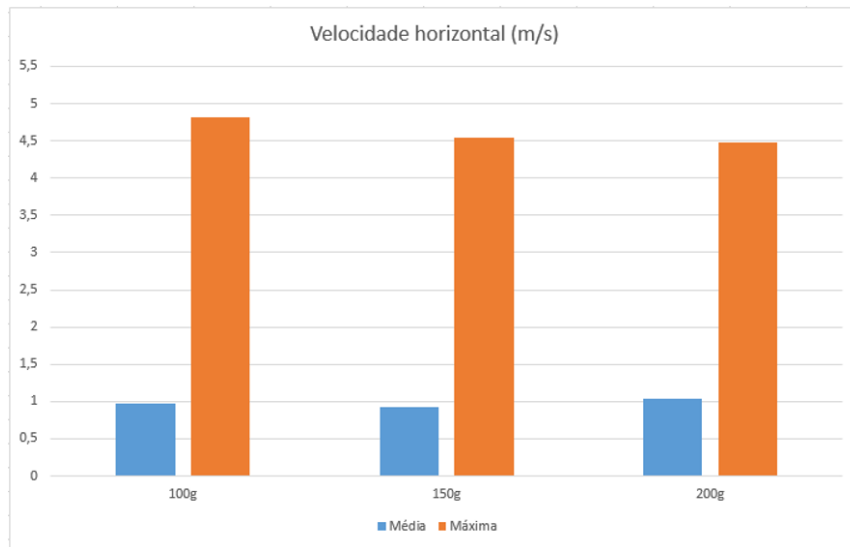


Figura 34. Gráfico de velocidade horizontal do *drone* com variadas cargas.

Durante a análise dos arquivos de LOG ficou claro que para manter o voo com carga 200g o *drone* consumia 60A (Figura 35), atingindo a corrente máxima de funcionamento de cada motor, que é de 15A, quando os motores passam a não funcionar corretamente mesmo durante a tentativa de pouso, causando acidentes.

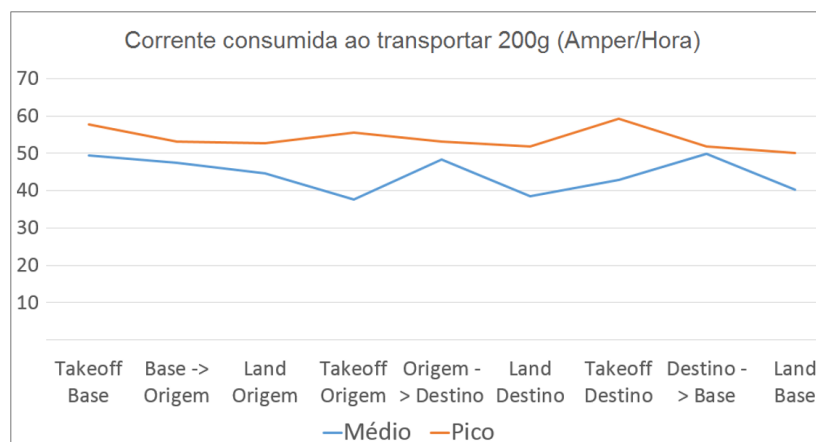


Figura 35. Gráfico de consumo de corrente com a maior carga transportada.

O teste de integração validou todas as funcionalidades da solução, desde a gerência das requisições de entregas até o envio das missões por meio do equipamento de telemetria, porém, por problemas específicos do equipamento de telemetria utilizado, a distância para transferência dos dados ficou muito reduzida, principalmente para o envio de missões.

Tabela 8. Resumo das seções de voos executadas.

Voos	Tempo mm:ss	Modos de voo usados				Observações
		1	2	3	4	
01	13:20	x	x		x	
02	03:00	x	x		x	
03	03:12	x	x		x	
04	17:57	x	x		x	Erros no GPS
05	11:40	x	x		x	Erros no GPS
06	14:43	x	x		x	Testes após correções no GPS – Erros no GPS
07	07:02	x	x		x	Testes após correções no GPS – Erros no GPS
08	02:52	x		x	x	Testes após correções no GPS – Problema resolvido
09	18:59	x		x	x	Missões autônomas – Erros no barômetro causaram um acidente
10	40:15	x	x	x	x	O <i>drone</i> entra em modo <i>Failsafe</i> após erros de comparação entre o barômetro e o GPS
11	18:31	x	x	x	x	O <i>drone</i> entra em modo <i>Failsafe</i> após erros de comparação entre o barômetro e o GPS
12	04:03	x		x		O <i>drone</i> entra em modo <i>Failsafe</i> após erros de comparação entre o barômetro e o GPS – Após estes voos a APM perdeu o <i>bootloader</i> durante um Update de Firmware
13	06:13	x	x	x		Testes após a troca do microcontrolador da HK Pilot Mega – Valores muito diferentes para altitude entre o GPS e o barômetro
14	41:10	x		x		Ajustes de sensibilidade do filtro EKF
15	17:17	x		x		Teste de integração
16	24:20	x		x		Aferir a precisão nos pousos
17	18:17	x		x		Aferir a precisão nos pousos
18	10:55	x		x		Teste de carga – o tempo nublado interferiu negativamente na precisão dos pousos
Tempo total de voos: 05:33:46						

Modos de voo: 1) Stabilize; 2) Loiter; 3) Auto; 4) RTL.

5.5 Desafios e Discussão

Como não foi encontrada uma solução que desligue os motores do *drone* após o pouso, para que o bagageiro pudesse ser substituído, os testes com carga foram executados com o bagageiro sempre preso aos servos, e o fio de controle responsável por enviar os comandos de prender ou desprender o bagageiro foi desligado da HK Pilot Mega, desta forma, o bagageiro não foi substituído durante os testes. Duas alternativas podem ser aplicadas para resolver esse problema: como o código-fonte do ArduCopter²⁶ é aberto, pode-se inserir comandos de navegação capazes de desarmar os motores por um determinado tempo e armar novamente; ou, fragmentar a entrega em três fases, ou seja, para cada entrega seriam executadas três missões, são elas:

1. Takeoff do heliponto base; Loiter_Time sobre o heliponto de origem; Land; Desprender o bagageiro;

²⁶ Sistema embarcado para controle de veículos aéreos de asas rotativas instalado na placa HK Pilot Mega.

2. Prender o bagageiro; Takeoff do heliponto de origem; Loiter_time sobre o heliponto de destino; Land; Desprender o bagageiro;
3. Prender o bagageiro; Takeoff do heliponto de destino; Loiter_time sobre o heliponto base; Land.

Ao final de cada missão o servidor de comunicação aguarda um tempo determinado antes de enviar a nova missão, até que a entrega seja concluída, porém, por problemas de *hardware* no equipamento de telemetria, as missões não podem ser enviadas a distâncias superiores a um metro e meio, inviabilizando testes com essa alternativa.

Os problemas de *hardware* encontrados nos módulos de telemetria não inviabilizaram a validação do sistema, uma vez que um novo kit de telemetria resolverá todos os problemas, podendo inclusive optar por kits de grande alcance. Foram encontrados kits de telemetria que garantem a comunicação em um raio de até 65Km.

Os valores resultantes dos testes de carga também não inviabilizam o uso da solução, bastando que o responsável por implementar a solução, faça a correta escolha do tipo de *drone* a ser usado, uma vez que o sistema pode utilizar qualquer tipo de *drone* de asas rotativas, ou seja, se o objetivo for transportar pequenos e leves objetos, um quadricóptero será capaz de executar a tarefa, caso exista a necessidade de transportar objetos maiores e pesados, deve-se optar por um hexacóptero, octacóptero ou superior. Outro dado importante a ser levado em conta no momento de escolher o *drone* é a distância esperada de atuação do mesmo, caso sejam necessários grandes deslocamentos, mesmo para transportar pequenos e leves objetos, deve-se escolher um hexacóptero, octacóptero ou superior, pois ele requisitará uma bateria maior e mais pesada para cumprir a entrega.

Os testes no transporte de carga mostraram que, caso seja iniciada uma entrega com carga superior à suportada pelo *drone*, em algum momento ele fará um pouso de emergência, assim, para aumentar a confiabilidade do sistema, sensores capazes de aferir o peso da carga devem ser instalados no *drone* ou no heliponto, assim, sempre que um usuário inserir no bagageiro cargas superiores à capacidade máxima determinada, o *drone* não deve iniciar a entrega e o sistema deverá ser informado que o usuário tentou enviar uma carga maior que a suportada.

Apesar de existirem *drones* comercializados em kits praticamente prontos para voar, tal escolha seria muito onerosa financeiramente, por isso optou-se por comprar as partes que comporiam o *drone* e o mesmo foi montado durante a pesquisa. Ainda por questões econômicas a opção escolhida foi um quadricóptero de pequeno porte, porém, após a execução dos testes de carga, concluímos que o *drone* montado não é suficiente para que o sistema seja posto em prática, sendo necessário a construção de um novo *drone*, respeitando as seguintes características mínimas: deve ter número de motores igual ou superior a seis; para reduzir peso final do *drone* o material utilizado para o frame e para as hélices deve ser a fibra de carbono; os motores devem suportar

pelo menos 1,5 Kg; e, utilizar baterias de seis células com corrente de no mínimo 10A. Tais alterações aumentam a carga útil do *drone* e também sua autonomia.

Novas tecnologias nos materiais que compõem as baterias podem trazer melhores resultados em relação à autonomia, já existem no mercado baterias de LiPO (Polímeros de Lítio) utilizando o grafeno como anodo, aumentando a densidade de elétrons e possibilitando a criação de baterias com maior capacidade sem aumento de peso e volume. Por suas características condutivas, baterias que utilizam o grafeno possuem uma baixa resistência interna, possibilitando maiores taxas de carga e descarga sem danificar a bateria.

Drones de asas rotativas com número de motores igual ou superior a cinco apresentam uma característica que reforça a segurança em voo, a possibilidade de continuar em voo e pousar em segurança caso surjam problemas em um dos motores ou hélices.

5.6 Conclusão

Os resultados dos testes de integração validaram a solução, porém, problemas de *hardware* na telemetria e barômetro, forçaram a execução de voos de testes não programados. A capacidade de carga do *drone* ficou abaixo do esperado, mas os testes foram suficientes para determinar os limites do *drone* em uso. Como o sistema pode utilizar qualquer *drone* de asas rotativas que utilize o protocolo de comunicação MAVLink, cabe ao responsável pela instalação do sistema, a escolha pelo *drone* adequado às suas necessidades.

Trabalhos futuros envolvem o controle e as entregas sendo efetuadas por vários *drones* simultaneamente, desta forma o sistema deverá efetuar todos os controles já implementados e também garantir que não ocorram colisões entre os *drones* em voo. Para aumentar a segurança no voo dos *drones*, sensores de impacto deverão ser instalados em pelo menos seis direções (frente, trás, esquerda, direita, acima e abaixo), assim, caso um obstáculo seja detectado, o *drone* deverá entrar em um modo de segurança que pode, entre outras coisas, ativar a função RTL do *drone*.

A fila de execução de entregas poderá receber melhorias que otimizem a utilização das baterias, por exemplo: o sistema deverá estimar se a autonomia do *drone* em um dado momento é suficiente para executar a próxima entrega e, caso contrário, o sistema deverá percorrer as entregas da fila em busca de alguma entrega que possa ser executada com segurança com a carga da bateria em uso no *drone*. O retorno do *drone* para o heliponto base, ao final de cada entrega, reduz consideravelmente a autonomia do *drone*, para evitar esse gasto desnecessário o *drone* deve finalizar a missão ao pousar no heliponto de destino e a próxima entrega deve iniciar deste heliponto.

5.6.1 Comentários finais

Os resultados obtidos neste capítulo provaram que o sistema gerenciou corretamente as entregas cadastradas e que o UAV foi capaz de receber as informações de uma entrega e executá-la de modo autônomo, retornando em seguida ao heliponto base onde deve aguardar novas informações de entrega. Entretanto, novos requisitos foram levantados durante os testes e podem ser implementados posteriormente, tais requisitos poderão melhorar o sistema tornando-o mais eficiente tanto na velocidade de entrega como na autonomia.

Capítulo 6

Considerações Finais

Neste capítulo serão apresentadas as considerações finais desta dissertação de mestrado, sinalizando a importância e relevância de cada passo do estudo, bem como, sintetizando os resultados logrados com a realização desta pesquisa.

6.1 Trabalho realizado

Após o levantamento do estado da arte dos veículos não tripulados, ficou claro que os estudos envolvendo veículos terrestres e aquáticos eram em sua maioria de cunho militar ou utilizados em competições entre estudantes. Como o objetivo principal da pesquisa é a implementação de uma solução que possa ser colocada em prática, escolhemos os veículos do tipo aéreo, uma vez que são os que apresentam maior versatilidade para executar aplicações com finalidades variadas. Focamos as pesquisas seguintes nas interfaces utilizadas no controle de veículos aéreos não tripulados e as aplicações práticas deste tipo de veículo, onde constatamos que as interfaces de controle encontradas requerem conhecimentos específicos de aviação e controle de voo por parte do usuário. Das diversas aplicações possíveis com o uso de *drones*, optamos por desenvolver um sistema de entregas automatizadas que não exige dos usuários nenhum conhecimento prévio de aviação ou controle de aeronaves, para em seguida executar testes de todo o sistema em funcionamento.

6.2 Resultados

- **Capítulo 2:** o levantamento do estado da arte dos veículos não tripulados serviu de base para o entendimento do domínio de veículos não tripulados, permitindo a identificação e análise dos principais desafios de *hardware* e *software* deste domínio. Este capítulo apresentou também diversos controladores de UVs, entre eles a ArduPilot Mega, escolhida para o controle do UAV utilizado nesta pesquisa.

- **Capítulo 3:** com base nos levantamentos do estado da arte das interfaces de usuário para controle de veículos aéreos não tripulados, foi possível entender os tipos de interfaces utilizadas, que tipo de interface é melhor aproveitada para cada tipo de veículo e quais os desafios e limitações das atuais interfaces em uso. Com base nesta pesquisa foi possível constatar que dispositivos comuns a outros domínios são amplamente utilizados como interfaces de controle de veículos aéreos não tripulados, e que essa adaptação pode gerar diversos problemas, podendo inclusive aumentar as chances de que acidentes ocorram;
- **Capítulo 4:** este capítulo apresentou os trabalhos relacionados às entregas automatizadas que utilizem UAVs, suas principais características, limitações e desafios, bem como apresenta as etapas do desenvolvimento do sistema PostDrone University, um sistema, composto por quatro programas, que é capaz de gerir e efetuar entregas automatizadas por meio de um UAV do tipo multirrotor. Apresenta também as etapas de construção e testes de voo do quadricóptero denominado UAV K-263.
- **Capítulo 5:** este capítulo teve como objetivo a execução dos testes que validarão a solução PostDrone University, três tipos de testes foram executados e documentados: testes de funcionamento dos sistemas desenvolvidos (Servidor do Sistema, Servidor de Comunicação MAVLink, Ferramenta Administrativa e Aplicativo Móvel), testes de voo (controlado e autônomo) do UAV K-263 e testes de integração das soluções desenvolvidas, com o sistema enviando as missões de entrega ao UAV K-263, para que este efetue a entrega. Os primeiros testes apontaram possíveis ajustes necessários nos *softwares* desenvolvidos, inclusive a necessidade de inserir um novo componente à solução, i.e., o Servidor de Comunicação MAVLink, e apresentou requisitos não-funcionais não previstos durante a fase de levantamento dos requisitos, como por exemplo, a altitude mínima de segurança entre os helipontos, que exigiu novas funcionalidades no cadastro de unidades. Os primeiros testes de voo foram essenciais para avaliar a montagem do UAV K-263, enquanto os voos autônomos confirmaram o correto funcionamento do sistema embarcado na placa HK Pilot Mega, bem como foi possível estimar as margens de erro apresentadas nos pousos automatizados do UAV K-263. Questões relativas à segurança na operação serão anotadas para futura documentação de uso da solução.

6.3 Perspectivas e Trabalhos Futuros

Apesar do crescente uso de *drones* nas mais variadas aplicações, poucas soluções são apresentadas para sistemas de entrega por meio de *drones*, desta forma muitos requisitos, funcionais e não-funcionais, ainda não foram elencados para tais aplicações, reforçando a necessidade de novas pesquisas neste domínio de aplicação.

Os resultados desta pesquisa sinalizam alguns novos desafios que podem ser objetos de pesquisa em trabalhos futuros, como:

- **Integração do PostDrone University entre várias instituições:** diversas instituições poderão utilizar a solução PostDrone University interagindo entre elas, assim, a universidade “A” poderá enviar objetos para a universidade “B”, com isso o *drone* da Universidade “A” deverá executar missões dentro da área de abrangência da universidade “B”;
- **Utilização um sistema embarcado para tomada de decisões:** A solução apresentada neste trabalho não dispõe de recursos embarcados no *drone* que permitam a tomada de decisões complexas caso surjam problemas durante uma entrega, por exemplo: mesmo com a instalação de sensores de obstáculos no *drone*, se durante uma entrega for detectado um obstáculo, ele deverá executar apenas uma ação (pouso ou RTL). Trabalhos futuros podem dar continuidade a esta solução, inserindo recursos que permitam a tomada de decisão durante as entregas, assim, caso surjam problemas, novos algoritmos deverão reformular o percurso do *drone* para que a entrega seja efetuada;
- **Utilização de um ou mais *drones* em uma mesma instituição:** a solução deverá controlar as entregas, inclusive selecionando o *drone* mais adequado à entrega a ser iniciada e, um sistema embarcado no *drone* deverá ser capaz de identificar as situações de risco que surgirem, para evitar acidentes, principalmente entre os *drones* em voo;
- **Otimizar as entregas com o intuito de economizar a carga da bateria:** o sistema poderá receber também melhorias para garantir um melhor aproveitamento das baterias, sabemos que a menor distância entre dois pontos é uma reta, por isso que o *drone* executa os voos entre os helipontos em linha reta, entretanto, o sistema deverá levar em consideração se o consumo de carga da bateria para voar respeitando a altitude mínima de segurança entre os helipontos será menor que o consumo caso o *drone* desvie dos obstáculos para efetuar as entregas. O sistema também deverá estimar se a autonomia do *drone* em um dado momento é suficiente para executar a próxima entrega e, caso contrário, deve escolher outra entrega da fila que possa ser executada com a carga da bateria em uso no *drone*.

Referências Bibliográficas

- [1] N. Tesla, “METHOD OF AND APPARATUS FOR CONTROLLING MECHANISM OF MOVING VESSELS OR VEHICLES,” 1898.
- [2] F. J. Pereda, H. G. de Marina, J. F. Jimenez, and J. M. Giron-Sierra, “A development project of autonomous marine surface vehicles for sea demining,” in *Control Automation Robotics Vision (ICARCV), 2010 11th International Conference on*, 2010, pp. 573–578.
- [3] G. Bruzzone, M. Bibuli, and M. Caccia, “Autonomous mine hunting mission for the Charlie USV,” in *OCEANS, 2011 IEEE - Spain*, 2011, pp. 1–6.
- [4] B. Schulz, R. Hughes, E. Matson, R. Moody, and B. Hobson, “The development of a free-swimming UUV for mine neutralization,” in *OCEANS, 2005. Proceedings of MTS/IEEE*, 2005, pp. 1443–1447 Vol. 2.
- [5] W. H. Wang, X. Q. Chen, A. Marburg, J. G. Chase, and C. E. Hann, “A Low-Cost Unmanned Underwater Vehicle Prototype for Shallow Water Tasks,” in *Mechtronic and Embedded Systems and Applications, 2008. MESA 2008. IEEE/ASME International Conference on*, 2008, pp. 204–209.
- [6] A. Bouhraoua, N. Merah, M. AlDajani, and M. Elshafei, “Design and implementation of an unmanned ground vehicle for security applications,” in *Mechatronics and its Applications (ISMA), 2010 7th International Symposium on*, 2010, pp. 1–6.
- [7] C.-H. Kim, K.-M. Jeung, Y.-S. Choi, S.-U. Lee, C.-H. Kim, and T. Jeong, “Design of the unmanned ground vehicle for bird expellant in an airport,” in *Control, Automation and Systems (ICCAS), 2011 11th International Conference on*, 2011, pp. 1094–1097.
- [8] M. Jew, A. El-Osery, and S. Bruder, “Implementation of an FPGA-based aided IMU on a low-cost autonomous outdoor robot,” in *Position Location and Navigation Symposium (PLANS), 2010 IEEE/ION*, 2010, pp. 1043–1051.
- [9] H. Miyanaka, N. Wada, T. Kamegawa, N. Sato, S. Tsukui, H. Igarashi, and F. Matsuno, “Development of an unit type robot ‘KOHGA2’ with stuck avoidance ability,” in *Robotics and Automation, 2007 IEEE International Conference on*, 2007, pp. 3877–3882.
- [10] D. Wooden, M. Malchano, K. Blankespoor, A. Howardy, A. A. Rizzi, and M. Raibert, “Autonomous navigation for BigDog,” in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2010, pp. 4736–4741.
- [11] X. Righetti, S. Cardin, D. Thalmann, and F. Vexo, “Immersive flight for surveillance applications,” in *3D User Interfaces, 2007. 3DUI '07. IEEE Symposium on*, 2007, p. -.
- [12] G. D. Chandler, D. K. Jackson, A. W. Groves, O. A. Rawashdeh, N. A. Rawashdeh, W. T. Smith, J. D. Jacob, and J. E. Lumppp, “A Low-Cost Control System for a High-Altitude UAV,” in *Aerospace Conference, 2005 IEEE*, 2005, pp. 1–12.
- [13] Z. Gong, Y. Deng, Z. Jiang, and L. Chen, “Aero-robotic vehicles for low-altitude remote-detecting on disaster scene,” in *Mechatronics and Automation, 2005 IEEE International Conference*, 2005, vol. 4, pp. 1941–1946 Vol. 4.

- [14] A. M. Brandt and M. B. Colton, "Haptic collision avoidance for a remotely operated quadrotor UAV in indoor environments," in *Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on*, 2010, pp. 2724–2731.
- [15] S. Yu, S. Ma, B. Li, and Y. Wang, "An amphibious snake-like robot with terrestrial and aquatic gaits," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, 2011, pp. 2960–2961.
- [16] A. Kalantari and M. Spenko, "Design and experimental validation of a hybrid aerial and terrestrial robot," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 2013.
- [17] A. Cadena, "Design and construction of an Autonomous Underwater Vehicle for the launch of a small UAV," in *Technologies for Practical Robot Applications, 2009. TePRA 2009. IEEE International Conference on*, 2009, pp. 78–83.
- [18] S. Bhatia, H. S. Dhillon, and N. Kumar, "Alive human body detection system using an autonomous mobile rescue robot," in *India Conference (INDICON), 2011 Annual IEEE*, 2011, pp. 1–5.
- [19] N. A. Rawashdeh, L. M. Alkurdi, and H. T. Jasim, "Development of a low cost differential drive Intelligent Ground Vehicle," in *Mechatronics and its Applications (ISMA), 2012 8th International Symposium on*, 2012, pp. 1–5.
- [20] P. Appelqvist, J. Knuuttila, and J. Ahtiainen, "Development of an Unmanned Ground Vehicle for task-oriented operation - considerations on teleoperation and delay," in *Advanced intelligent mechatronics, 2007 IEEE/ASME international conference on*, 2007, pp. 1–6.
- [21] J. Gebauer, P. Koci, and P. Sofer, "Multicopter potentialities," in *Carpathian Control Conference (ICCC), 2012 13th International*, 2012, pp. 194–197.
- [22] S. Gupte, P. I. T. Mohandas, and J. M. Conrad, "A survey of quadrotor Unmanned Aerial Vehicles," in *Southeastcon, 2012 Proceedings of IEEE*, 2012, pp. 1–6.
- [23] S. Noda, S. Machida, and H. Lim, "Mechanism and control of four rotor flying robot," in *Control, Automation and Systems (ICCAS), 2011 11th International Conference on*, 2011, pp. 1152–1157.
- [24] I. Sa and P. Corke, "System identification, estimation and control for a cost effective open-source quadcopter," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, 2012, pp. 2202–2209.
- [25] J. Mclean, G. Parker, and N. Seal, "Basic control for four rotor autonomous aerial agent," in *Automation Congress, 2008. WAC 2008. World*, 2008, pp. 1–8.
- [26] H. A. Ruff, N. Rao Edala, J. Geist, P. K. Kumar, S. Narayanan, M. Draper, and M. W. Haas, "An architecture for modeling uninhabited aerial vehicles," in *Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference on*, 1999, vol. 1, pp. 744–749 vol.1.
- [27] A. Albers, S. Trautmann, T. Howard, T. A. Nguyen, M. Frietsch, and C. Sauter, "Semi-autonomous flying robot for physical interaction with environment," in *Robotics Automation and Mechatronics (RAM), 2010 IEEE Conference on*, 2010, pp. 441–446.
- [28] A. Gademer, F. Mainfroy, L. Beaudoin, L. Avanthey, V. Germain, C. Cheron, S. Monat, and J.-P. Rudant, "Faucon noir UAV project development of a set of tools for managing, visualizing and mosaicing centimetric UAV images," in *Geoscience and Remote Sensing Symposium, 2009 IEEE International, IGARSS 2009*, 2009, vol. 3, pp. III–228–III–231.

- [29] S. Divakar, "Multi-colored line following and obstacle avoiding helicopter," in *Electronics Computer Technology (ICECT), 2011 3rd International Conference on*, 2011, vol. 4, pp. 135–139.
- [30] B. T. M. Leong, S. M. Low, and M. P.-L. Ooi, "Low-Cost Microcontroller-based Hover Control Design of a Quadcopter," *Procedia Eng.*, vol. 41, no. 0, pp. 458–464, 2012.
- [31] J. Amahah, "The design of an unmanned aerial vehicle based on the ArduPilot," *Georg. Electron. Sci. J. Comput. Sci. Telecommun.*, no. 5, p. 22, 2009.
- [32] M. A. Zafar, A. Rauf, Z. Ashraf, and H. Akhtar, "Design and development of effective manual control system for unmanned air vehicle," in *Computer Research and Development (ICCRD), 2011 3rd International Conference on*, 2011, vol. 1, pp. 349–353.
- [33] N. Berezny, L. de Greef, B. Jensen, K. Sheely, M. Sok, D. Lingenbrink, and Z. Dodds, "Accessible aerial autonomy," in *Technologies for Practical Robot Applications (TePRA), 2012 IEEE International Conference on*, 2012, pp. 53–58.
- [34] W. S. Ng and E. Sharlin, "Collocated interaction with flying robots," in *RO-MAN, 2011 IEEE*, 2011, pp. 143–149.
- [35] M. S. Alvissalim, B. Zaman, Z. A. Hafizh, M. A. Ma'sum, G. Jati, W. Jatmiko, and P. Mursanto, "Swarm quadrotor robots for telecommunication network coverage area expansion in disaster area," in *SICE Annual Conference (SICE), 2012 Proceedings of*, 2012, pp. 2256–2261.
- [36] N. Michael, E. Stump, and K. Mohta, "Persistent surveillance with a team of MAVs," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, 2011, pp. 2708–2714.
- [37] C. Blum and V. V. Hafner, "An Autonomous Flying Robot for Network Robotics," in *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on*, 2012, pp. 1–5.
- [38] P. Rudol, M. Wzorek, G. Conte, and P. Doherty, "Micro Unmanned Aerial Vehicle Visual Servoing for Cooperative Indoor Exploration," in *Aerospace Conference, 2008 IEEE*, 2008, pp. 1–10.
- [39] P. T. Nathan, H. A. F. Almurib, and T. N. Kumar, "A review of autonomous multi-agent quad-rotor control techniques and applications," in *Mechatronics (ICOM), 2011 4th International Conference On*, 2011, pp. 1–7.
- [40] J. M. de la Cruz, E. Besada-Portas, L. Torre-Cubillo, B. Andres-Toro, and J. A. Lopez-Orozco, "Evolutionary path planner for UAVs in realistic environments," in *Proceedings of the 10th annual conference on Genetic and evolutionary computation*, 2008, pp. 1477–1484.
- [41] K. Culligan, M. Valenti, Y. Kuwata, and J. P. How, "Three-Dimensional Flight Experiments Using On-Line Mixed-Integer Linear Programming Trajectory Optimization," in *American Control Conference, 2007. ACC '07*, 2007, pp. 5322–5327.
- [42] P. P.-Y. Wu, D. Campbell, and T. Merz, "On-board multi-objective mission planning for Unmanned Aerial Vehicles," in *Aerospace conference, 2009 IEEE*, 2009, pp. 1–10.
- [43] X. Yang, M. Ding, and C.-P. Zhou, "Fast Marine Route Planning for UAV Using Improved Sparse A* Algorithm," in *Genetic and Evolutionary Computing (ICGEC), 2010 Fourth International Conference on*, 2010, pp. 190–193.
- [44] S. G. Fowers, D.-J. Lee, B. J. Tippetts, K. D. Lillywhite, A. W. Dennis, and J. K. Archibald, "Vision Aided Stabilization and the Development of a Quad-Rotor Micro UAV," in *Computational Intelligence in Robotics and Automation, 2007. CIRA 2007. International Symposium on*, 2007, pp. 143–148.

- [45] A. Khozouhee, C. Brennan, E. Gonçalves, and E. Urhiae, “AAPS – Automated Aero-Painting System,” 2012. [Online]. Available: http://www.ecs.umass.edu/ece/sdp/sdp12/moritz/AAPS_Final_Report.pdf. [Accessed: 19-Sep-2012].
- [46] J. T. Hing, J. Menda, K. Izzetoglu, and P. Y. Oh, “An Indoor Study to Evaluate a Mixed-reality Interface for Unmanned Aerial Vehicle Operations in Near Earth Environments,” in *Proceedings of the 10th Performance Metrics for Intelligent Systems Workshop*, 2010, pp. 214–221.
- [47] J. T. Hing and P. Y. Oh, “Development of an Unmanned Aerial Vehicle Piloting System with Integrated Motion Cueing for Training and Pilot Evaluation,” *J. Intell. Robot. Syst.*, vol. 54, no. 1–3, pp. 3–19, 2009.
- [48] F. Saghafi, S. Pouya, and S. M. K. Zadeh, “Intelligent landing of autonomous aerial vehicles using fuzzy logic control,” in *Aerospace conference, 2009 IEEE*, 2009, pp. 1–9.
- [49] M. Shaker, M. N. R. Smith, S. Yue, and T. Duckett, “Vision-Based Landing of a Simulated Unmanned Aerial Vehicle with Fast Reinforcement Learning,” in *Emerging Security Technologies (EST), 2010 International Conference on*, 2010, pp. 183–188.
- [50] P. I. T. M. Das, S. Swami, and J. M. Conrad, “An algorithm for landing a Quadrotor Unmanned Aerial Vehicle on an oscillating surface,” in *Southeastcon, 2012 Proceedings of IEEE*, 2012, pp. 1–4.
- [51] C. S. Murdoch and S. N. Reynoso, “Design and Implementation of a MPPT circuit for a Solar UAV,” *Lat. Am. Trans. IEEE (Revista IEEE Am. Lat.*, vol. 11, no. 1, pp. 108–111, 2013.
- [52] W. Cheng-long, L. Zhen-ye, F. Ming-yu, and B. Xin-Qian, “Amphibious hovercraft course control based on adaptive multiple model approach,” in *Mechatronics and Automation (ICMA), 2010 International Conference on*, 2010, pp. 601–604.
- [53] R. R. Murphy, K. S. Pratt, and J. L. Burke, “Crew roles and operational protocols for rotary-wing micro-UAVs in close urban environments,” in *Human-Robot Interaction (HRI), 2008 3rd ACM/IEEE International Conference on*, 2008, pp. 73–80.
- [54] L. Zong-jian and P. Xiao-dong, “Low Altitude Photogrammetry of Unmanned Airship,” in *Information Engineering and Computer Science, 2009. ICIECS 2009. International Conference on*, 2009, pp. 1–4.
- [55] a. Zul Azfar and D. Hazry, “Simple GUI design for monitoring of a remotely operated quadrotor unmanned aerial vehicle (UAV),” *2011 IEEE 7th Int. Colloq. Signal Process. its Appl.*, pp. 23–27, Mar. 2011.
- [56] R. K. Rangel, K. Heinz Kienitz, and M. P. Brandão, “Development of a multi-purpose portable electrical UAV system, fixed amp; rotative wing,” in *Aerospace Conference, 2011 IEEE*, 2011, pp. 1–9.
- [57] P. Ulbrich, R. Kapitza, C. Harkort, R. Schmid, and W. Schröder-Preikschat, “I4Copter: an adaptable and modular quadrotor platform,” in *Proceedings of the 2011 ACM Symposium on Applied Computing*, 2011, pp. 380–386.
- [58] S. Divakar, “Cell phone controlled rocker-bogie suspension type rover with a scooping arm,” in *Ubiquitous Robots and Ambient Intelligence (URAI), 2011 8th International Conference on*, 2011, pp. 822–827.
- [59] J. Stowers, M. Hayes, and A. Bainbridge-Smith, “Altitude control of a quadrotor helicopter using depth map from Microsoft Kinect sensor,” in *Mechatronics (ICM), 2011 IEEE International Conference on*, 2011, pp. 358–362.

- [60] B. A. Rocca, S. Preidikman, C. G. Gebhardt, and J. C. Massa, "Dynamics of Micro-Air-Vehicles with Flapping Wings: A Multibody System Approach," *Lat. Am. Trans. IEEE (Revista IEEE Am. Lat.)*, vol. 11, no. 1, pp. 183–189, 2013.
- [61] A. Cavaleri, B. Motella, M. Pini, and M. Fantino, "Detection of spoofed GPS signals at code and carrier tracking level," in *Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), 2010 5th ESA Workshop on*, 2010, pp. 1–6.
- [62] T.-H. Kim, C. S. Sin, and S. Lee, "Analysis of effect of spoofing signal in GPS receiver," in *Control, Automation and Systems (ICCAS), 2012 12th International Conference on*, 2012, pp. 2083–2087.
- [63] O. Pozzobon, L. Canzian, M. Danieleto, and A. D. Dalla Chiara, "Anti-spoofing and open GNSS signal authentication with signal authentication sequences," in *Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), 2010 5th ESA Workshop on*, 2010, pp. 1–6.
- [64] C. Xi-jun, C. Ke-jin, X. Jiang-ning, and L. Bao, "Analysis on Forgery Patterns for GPS Civil Spoofing Signals," in *Computer Sciences and Convergence Information Technology, 2009. ICCIT '09. Fourth International Conference on*, 2009, pp. 353–356.
- [65] D. Cavett, M. Coker, R. Jimenez, and B. Yaacoubi, "Human-Computer Interface for Control of Unmanned Aerial Vehicles," in *Systems and Information Engineering Design Symposium, 2007. SIEDS 2007. IEEE*, 2007, pp. 1–6.
- [66] Y. Kang and M. Yuan, "Software design for mini-type ground control station of UAV," in *Electronic Measurement Instruments, 2009. ICEMI '09. 9th International Conference on*, 2009, pp. 4–740.
- [67] M. Quigley, M. A. Goodrich, and R. W. Beard, "Semi-autonomous human-UAV interfaces for fixed-wing mini-UAVs," in *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, 2004, vol. 3, pp. 2457–2462 vol.3.
- [68] B. Kayayurt, I. Yayla, A. Yapici, and C. Kucukoguz, "Ground control station avionics software development in ANKA UAV," in *Digital Avionics Systems Conference (DASC), 2011 IEEE/ALAA 30th*, 2011, pp. 5B6–1–5B6–7.
- [69] F. Crescenzo, G. Miranda, F. Persiani, and T. Bombardi, "A First Implementation of an Advanced 3D Interface to Control and Supervise UAV (Uninhabited Aerial Vehicles) Missions," *Presence*, vol. 18, no. 3, pp. 171–184, 2009.
- [70] M. Jasinski and A. Nawrat, "Graphical human-machine interface for QB systems," in *Human System Interactions, 2008 Conference on*, 2008, pp. 360–364.
- [71] A. Akce, M. Johnson, O. Dantsker, and T. Bretl, "A Brain #x2013;Machine Interface to Navigate a Mobile Robot in a Planar Workspace: Enabling Humans to Fly Simulated Aircraft With EEG," *Neural Syst. Rehabil. Eng. IEEE Trans.*, vol. 21, no. 2, pp. 306–318, 2013.
- [72] W. Karlen, S. Cardin, D. Thalmann, and D. Floreano, "Enhancing pilot performance with a SymBodic system," in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, 2010, pp. 6599–6602.
- [73] Z. Al-Khatib, J. Yu, H. G. Al-Khakani, and S. Kombarji, "A wireless multivariable control scheme for a quadrotor hovering robotic platform using IEEE® 802.15.4," *IEEE Publ. Student Appl. Pap.*, 2009.
- [74] K. Pfeil, S. L. Koh, and J. LaViola, "Exploring 3D Gesture Metaphors for Interaction with Unmanned Aerial Vehicles," in *Proceedings of the 2013 International Conference on Intelligent User*

- Interfaces*, 2013, pp. 257–266.
- [75] A. Zul Azfar and D. Hazry, “Simple GUI design for monitoring of a remotely operated quadrotor unmanned aerial vehicle (UAV),” in *Signal Processing and its Applications (CSPA), 2011 IEEE 7th International Colloquium on*, 2011, pp. 23–27.
 - [76] T. Wypych and F. Kuester, “System for interactive management of aerial imaging campaigns,” in *Aerospace Conference, 2013 IEEE*, 2013, pp. 1–8.
 - [77] A. A. Sandor, B. G. Soos, G. Cserey, and G. Szederkenyi, “Ergonomic remote control technique for horizontal rotors equipped UAVs,” in *Industrial Electronics, 2009. IECON '09. 35th Annual Conference of IEEE*, 2009, pp. 2223–2228.
 - [78] M. Hou, G. Ho, G. R. Arrabito, S. Young, and S. Yin, “Effects of display mode and input method for handheld control of micro aerial vehicles for a reconnaissance mission,” *Human-Machine Syst. IEEE Trans.*, vol. 43, no. 2, pp. 149–160, 2013.
 - [79] J. M. Peschel, “Towards physical object manipulation by small unmanned aerial systems,” in *Safety, Security, and Rescue Robotics (SSRR), 2012 IEEE International Symposium on*, 2012, pp. 1–6.
 - [80] H. Il Son, A. Franchi, L. L. Chuang, J. Kim, H. H. Bulthoff, and P. R. Giordano, “Human-Centered Design and Evaluation of Haptic Cueing for Teleoperation of Multiple Mobile Robots,” *Cybern. IEEE Trans.*, vol. 43, no. 2, pp. 597–609, 2013.
 - [81] T. M. Lam, M. Mulder, and M. M. van Paassen, “Haptic interface in UAV tele-operation using force-stiffness feedback,” in *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*, 2009, pp. 835–840.
 - [82] A. Ruesch, A. Y. Mersha, S. Stramigioli, and R. Carloni, “Kinetic scrolling-based position mapping for haptic teleoperation of unmanned aerial vehicles,” in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, 2012, pp. 3116–3121.
 - [83] J. Smisek, M. M. van Paassen, M. Mulder, and D. A. Abbink, “Neuromuscular analysis based tuning of haptic shared control assistance for UAV collision avoidance,” in *World Haptics Conference (WHC), 2013*, 2013, pp. 389–394.
 - [84] H. Vargas, E. Preza, and R. Velazquez, “A Combined Wiimote-Camera Tracking System for Small Aerial Vehicles,” in *Electronics, Robotics and Automotive Mechanics Conference, 2009. CERMA '09.*, 2009, pp. 226–231.
 - [85] S. Kasahara, R. Niiyama, V. Heun, and H. Ishii, “exTouch: Spatially-aware Embodied Manipulation of Actuated Objects Mediated by Augmented Reality,” in *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction*, 2013, pp. 223–228.
 - [86] I. Maza, F. Caballero, R. Molina, N. Peña, and A. Ollero, “Multimodal Interface Technologies for UAV Ground Control Stations,” *J. Intell. Robot. Syst.*, vol. 57, no. 1–4, pp. 371–391, 2010.
 - [87] B. Markwalter, “Flights of Fancy: Products in the Unmanned Systems Marketplace,” *IEEE Consum. Electron. Mag.*, vol. 4, no. 2, pp. 46–48, Apr. 2015.
 - [88] L. De Miranda, H. Hornung, and M. C. Baranauskas, “Adjustable interactive rings for iDTV,” *IEEE Trans. Consum. Electron.*, vol. 56, no. 3, pp. 1988–1996, Aug. 2010.
 - [89] L. C. de Miranda and M. C. C. Baranauskas, “Artefato Físico de Interação de Televisão Digital,” PI10134662, 2010.
 - [90] G. Kimchi, D. Buchmueller, S. A. Green, B. C. Beckman, S. Isaacs, A. Navot, F. Hensel, A. Bar-Zeev, S. Sylvain, and J.-M. Rault, “UNMANNED AERIAL VEHICLE

- DELIVERY SYSTEM,” US2015/0120094A1, 2015.
- [91] M. Heutger and M. Kückelhaus, “UNMANNED AERIAL VEHICLES IN LOGISTICS: A DHL perspective on implications and use cases for the logistics industry,” Troisdorf, 2014.
- [92] E. Ackerman, “The Way to Make Delivery *Drones* Work Is Using...Trucks? - IEEE Spectrum.” [Online]. Available: <http://spectrum.ieee.org/automaton/robotics/drones/amp-electric-truck-delivery-drones>. [Accessed: 10-Feb-2016].
- [93] Y.-F. Lin, W.-M. Lu, K.-H. Chen, and J.-I. Guo, “Vision-based landing system design for a small UAV,” in *2015 IEEE International Conference on Consumer Electronics - Taiwan*, 2015, pp. 496–497.
- [94] E. Peeters, E. Teller, and W. G. Patrick, “PROVIDING EMERGENCY MEDICAL SERVICES USING UNMANNED AERIAL VEHICLES,” US 9.051.043 B1, 2015.
- [95] B. T. Boshielo, B. Mendelow, P. Muir, and J. Robertson, “Development of e-Juba, a preliminary proof of concept unmanned aerial vehicle designed to facilitate the transportation of microbiological test samples from remote rural clinics to National Health Laboratory Service laboratories,” *South African Med. J.*, vol. 97, p. 1215+, Jul. 2007.
- [96] O. A. Yakimenko, N. J. Slegers, E. A. Bourakov, C. W. Hewgley, A. B. Bordetsky, R. P. Jensen, A. B. Robinson, J. R. Malone, and P. E. Heidt, “Mobile system for precise aero delivery with global reach network capability,” in *Control and Automation, 2009. ICCA 2009. IEEE International Conference on*, 2009, pp. 1394–1398.
- [97] M. R. Haque, M. Muhammad, D. Swarnaker, and M. Arifuzzaman, “Autonomous Quadcopter for product home delivery,” in *Electrical Engineering and Information Communication Technology (ICEEICT), 2014 International Conference on*, 2014, pp. 1–5.
- [98] W. J. Eggington and P. M. Stevens, “The Cyclocraft and its applications,” in *Aerospace and Electronics Conference, 1994. NAECON 1994., Proceedings of the IEEE 1994 National*, 1994, pp. 1272–1279 vol.2.
- [99] E. Jones, “New approaches for autonomous logistics aircraft and ground systems,” in *Autonomous Systems, 2007 Institution of Engineering and Technology Conference on*, 2007, pp. 1–7.
- [100] M. Ozartan, S. Akgul, and B. Akca, “A different view to future use of unmanned aerial vehicles,” in *Unmanned Aircraft Systems (ICUAS), 2013 International Conference on*, 2013, pp. 167–172.