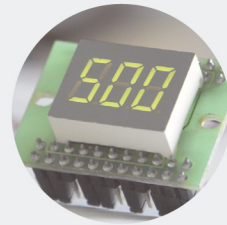




Calibrac#807;a#771;o dos par#226;metros extr#305;́nsecos de um LiDAR num UAV

ANDR#233; FILIPE MARTINS FERREIRA

novembro de 2018



Calibração de parâmetros extrínsecos de um LiDAR num UAV

ANDRÉ FILIPE MARTINS FERREIRA

Outubro de 2018



Calibração dos parâmetros extrínsecos de um LiDAR num UAV

André Filipe Martins Ferreira
Nº 1110336

Mestrado em Engenharia Eletrotécnica e de Computadores
Área de Especialização de Sistemas Autónomos
Departamento de Engenharia Electrotécnica
Instituto Superior de Engenharia do Porto

2018



Dissertação, para satisfação parcial dos requisitos do Mestrado em
Engenharia Eletrotécnica e de Computadores

Candidato: André Filipe Martins Ferreira

N^o 1110336

Orientador: José Miguel Soares De Almeida

Co-Orientador: André Miguel Pinheiro Dias

Mestrado em Engenharia Eletrotécnica e de Computadores

Área de Especialização de Sistemas Autónomos

Departamento de Engenharia Electrotécnica

Instituto Superior de Engenharia do Porto

31 de Outubro de 2018

Agradecimientos

Esta página foi intencionalmente deixada em branco.

Resumo

Quando um sensor LiDAR é montado numa plataforma móvel, seja esta terrestre ou aérea, é necessário determinar a posição absoluta e a orientação do sensor para utilizar os dados obtidos, seja para modelação 3D, calculo de áreas de zonas de interesse ou, como na maioria das vezes na área da robótica, para mapeamento 3D.

Os veículos responsáveis por transportar estes sensores são, normalmente capacitados de sistemas de GNSS que fornecem informações geográficas e de INS que registam a orientação dos veículos.

Esta dissertação propõe endereçar o problema de estimar a posição e orientação relativa do sensor LiDAR em relação a um UAV em diferentes cenários.

Nesse sentido, pretende-se que a abordagem tomada seja capaz de detetar planos, em múltiplas orientações, numa nuvem de pontos geo-referenciada, extrair os parâmetros que os representam e utilizá-los para estimar os parâmetros extrínsecos do sensor.

Palavras-Chave: Nuvem de Pontos, LiDAR, Parâmetros Extrínsecos, Detecção de Planos

Esta página foi intencionalmente deixada em branco.

Abstract

When a LiDAR sensor is mounted on a mobile vehicle, been aerial or terrestrial, it is necessary to determine the absolute position and orientation of the sensor to use the obtained data, either for 3D modeling, measurement of interest zones or, as usually on robotics field, for 3D mapping.

Vehicles that have to transport those sensors are also endowed with GNSS modules that provide geographical information and INS modules that record the orientation of those vehicles.

This thesis proposes to address the estimation problem of the relative position and orientation between the LiDAR sensor and the UAV robot in multiple scenarios.

Therefore, it is intended to the took approach to be able to detect planes in multiple orientations on a georeferenced pointcloud, extract the parameters that represent them and use them to estimate the extrinsic parameters of the sensor.

Keywords: PointCloud, LiDAR, Extrinsic Parameters, Plane Detection

Esta página foi intencionalmente deixada em branco.

Conteúdo

Agradecimentos	i
Resumo	iii
Abstract	v
Lista de Figuras	ix
Lista de Tabelas	xi
Lista de Siglas e Acrónimos	xiii
1 Introdução	1
1.1 Motivação	2
1.2 Objetivos	3
1.3 Estrutura	4
2 Estado da Arte	5
2.1 Métodos de calibração entre câmara e INS	5
2.2 Métodos de calibração entre LiDAR e sistema visível	6
2.2.1 Calibração com recurso a alvo	6
2.2.2 Calibração sem alvo	8
2.3 Métodos de calibração entre LiDAR e INS	9
3 Fundamentos Teóricos	13
3.1 <i>Earth-Centered Earth-Fixed</i>	13
3.2 Transformação de referenciais	13
3.2.1 Transformação Linear	14

3.2.2	Transformação Angular	14
3.3	Equação geral do Plano	15
3.4	RANSAC	16
3.5	LiDAR	17
3.5.1	Scan mechanisms	17
3.6	Influências de erros nos parâmetros	17
4	Algoritmo	19
4.1	Associação de pontos a Posição do UAV	20
4.2	Divisão da Nuvem de Pontos pela atitude do UAV	20
4.3	Construção da Nuvem de Pontos	21
4.4	Deteção de planos na Nuvem de pontos	21
4.5	Estimação dos parâmetros de calibração	22
5	Implementação	25
5.1	Arquitetura de hardware	25
5.1.1	UAV	25
5.1.2	Velodyne VLP-16	26
5.1.3	INS	26
5.2	Correlação temporal dos dados	27
5.3	Biblioteca PCL	28
5.3.1	RANSAC	28
6	Resultados e Análise	31
6.1	Cenário simulado	31
6.1.1	Trajectoria e Parâmetros reais	32
6.1.2	Resultados da Calibração	33
6.1.3	Alteração de Altura	36
6.2	Dados reais	38
6.2.1	ERL Emergency Robots 2017	38
6.2.2	FADEUP	38
7	Conclusão e Trabalho Futuro	41
	Bibliografia	43

Lista de Figuras

1.1	Espectro Eletromagnético	1
1.2	Alguns dos robôs desenvolvidos pelo LSA	3
2.1	Alvo de calibração com vista da câmara e do LiDAR	6
2.2	Imagens do padrão com a projeção dos pontos do LiDAR 2D	7
2.3	Imagem obtida pela câmara à direita, dados do LiDAR com atribuição da cor correspondente.	8
2.4	Influência dos erros nos ângulos de <i>boresight</i> em dois voos paralelos	11
3.1	Ângulos de euler aplicados na navegação de um UAV	14
3.2	Ponto P num plano α com vetor normal $\vec{n} = (a, b, c)$	15
3.3	Representação de três planos alvo perpendiculares entre si a) e a sua reprojeção em três vistas diferentes b), c) e d).	18
4.1	Algoritmo para a calibração dos parâmetros extrínsecos do LiDAR	19
4.2	Reprojeção de planos após divisão dos dados	20
4.3	Nuvem de pontos geo-referenciada	21
5.1	UAV STORK	25
5.2	Velodyne VLP-16	26
5.3	Análise de um registo de dados yaw	27
6.1	Superfícies virtuais utilizadas para obter dados simulados.	31
6.2	Trajetória executada na simulação.	32
6.3	Evolução do erro dos parâmetros extrínsecos	34
6.4	Evolução do erro da distância de todos os pontos ao plano respetivo	34
6.5	Comparação visual dos planos reprojctados	35

6.6	Trajectoria executada na simulacao.	36
6.7	Evolucao do erro dos parametros extrinsecos	37
6.8	Evolucao do erro da distancia de todos os pontos ao plano respetivo	37
6.9	Torre del Sale	38
6.10	FADEUP	39
6.11	Nuvem de pontos do campo de futebol da FADEUP	39
6.12	Nuvem de pontos do campo de futebol da FADEUP	40

Lista de Tabelas

- 5.1 Características do Velodyne VLP-16 26
- 6.1 Parâmetros extrínsecos do *Light Detection And Ranging* (LiDAR) simulado 33
- 6.2 Parâmetros iniciais da calibração do LiDAR 33
- 6.3 Resultados de 36 iterações consecutivas 33

Esta página foi intencionalmente deixada em branco.

Lista de Siglas e Acrónimos

CRAS Centro de Robótica e Sistemas Autónomos

ECEF *Earth-Centered Earth-Fixed*

FADEUP *Faculdade de Desporto da Universidade do Porto*

FOV *Field of View*

GNSS *Global Navigation Satellite System*

GPS *Global Position System*

IMU *Inertial Measurement Unit*

INESC TEC Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência

INS *Inertial Navigation System*

ISEP Instituto Superior de Engenharia do Porto

LASER *Light Amplification by Stimulated Emission of Radiation*

LiDAR *Light Detection And Ranging*

LSA Laboratório de Sistemas Autónomos

PCL *Point Cloud Library*

RADAR *Radio Detection and Ranging*

RANSAC *RANdom SAmple Consensus*

RPY *Roll Pitch Yaw*

UAV *Unmanned Aerial Vehicle*

Esta página foi intencionalmente deixada em branco.

Capítulo 1

Introdução

O uso da tecnologia de detecção e medição de distâncias pelo uso de ondas eletromagnéticas tem cerca de um século com a construção do primeiro radar, do inglês *Radio Detection and Ranging*. Enquanto que o radar utiliza ondas rádio e micro-ondas, com a frequência da radiação entre os 3 MHz e os 110 GHz, o LiDAR, do inglês *Light Detection And Ranging*, utiliza outro segmento de frequências do espectro eletromagnético, com os comprimentos de onda entre os 10 μm e os 250 nm, correspondendo a um intervalo entre a luz infravermelha e a ultravioleta, passando pela visível, como se pode observar na figura 1.1.

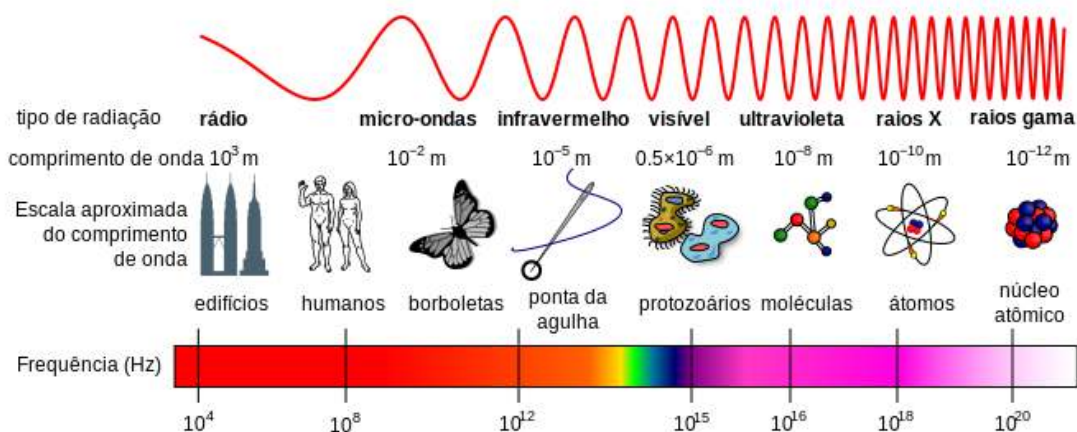


Figura 1.1: Espectro Eletromagnético

As primeiras aplicações do LiDAR vieram da meteorologia, nos anos 60, quando o Centro Nacional de Pesquisas Atmosféricas, nos Estados Unidos da América, usou a

tecnologia para medir nuvens. Na década seguinte, na missão Apollo 15, a NASA (*National Aeronautics and Space Administration*) utilizou a tecnologia LiDAR para mapear a superfície lunar [1], tendo levado a um maior conhecimento dessa tecnologia por parte do público geral. Desde então, o uso do LiDAR tem vindo a ser utilizado na agricultura e silvicultura, topografia em minas a céu aberto, inspeções de construções e estruturas, mapeamento de ambientes urbanos, gestão de recursos, modelos digitais de elevação, carros autónomos, entre outros fins.

Na maioria dos casos apresentados, o LiDAR vai a bordo de um *Unmanned Aerial Vehicle* (UAV) para maior área de cobertura, devido a maior manobrabilidade e facilidade de acesso por via aérea. Para análise da área inspeccionada, é necessário obter os dados numa nuvem de pontos que abrange a mesma, sendo por isso fundamental o conhecimento da posição exata do LiDAR no momento da obtenção de dados.

Nesta dissertação propõe-se então endereçar o problema da calibração dos parâmetros extrínsecos do LiDAR em relação à navegação do UAV que o transporta, para a obtenção de uma nuvem de pontos necessária na extração de dados sobre a área sobrevoada, recorrendo apenas a superfícies da mesma.

1.1 Motivação

O Laboratório de Sistemas Autónomos (LSA), do Instituto Superior de Engenharia do Porto (ISEP), tem estado ativamente envolvido em projetos de robótica aplicada devido à sua ligação com o Centro de Robótica e Sistemas Autónomos (CRAS), do Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência (INESC TEC). Os sistemas autónomos desenvolvidos abrangem o meio aquático, aéreo e terrestre tal como demonstrado por alguns desses robôs presentes na figura 1.2.

Esta dissertação insere-se nos projetos com UAVs que necessitam do mapeamento de uma área de interesse, como por exemplo, barragens e minas de céu aberto e consiste na calibração dos parâmetros extrínsecos do LiDAR para obtenção da nuvem de pontos com melhor qualidade, sem necessidade de pré-calibração e estruturação do meio.



Figura 1.2: Alguns dos robôs desenvolvidos pelo LSA

1.2 Objetivos

Nesta dissertação é endereçado o problema da calibração dos parâmetros extrínsecos de um LiDAR, fator crítico para a produção de um mapa para as aplicações descritas em 1.

Assim sendo, o desenvolvimento do método de calibração pressupõe a realização dos seguintes objetivos:

- Análise do estado de arte da calibração de extrínsecos de um LiDAR;
- Desenvolvimento de um método de calibração dos parâmetros extrínsecos de um LiDAR com base num ambiente não estruturado;
- Validação do uso do método desenvolvido em diferentes cenários.

1.3 Estrutura

Esta dissertação encontra-se organizada em sete capítulos. No segundo capítulo é apresentado um estudo acerca de diferentes abordagens já desenvolvidas pela comunidade científica sobre métodos de calibração entre câmaras, INS e LiDAR.

O capítulo 3 aborda conceitos e fundamentos teóricos necessários para a compreensão desta dissertação.

No capítulo 4 é explicado o algoritmo desenvolvido para a calibração de parâmetros extrínsecos de um LiDAR, bem como uma explicação de cada um dos passos executados.

O capítulo seguinte, capítulo 5, é descrita a arquitetura de *hardware* utilizada na obtenção de nuvens de pontos em cenários reais.

A validação do algoritmo e a apresentação dos seus resultados são realizados no capítulo 6, onde esta abordagem é testada em cenários reais e simulados.

Por fim, no capítulo 7 são apresentadas conclusões sobre o projeto desenvolvido e são apresentadas algumas linhas de trabalho futuro a realizar.

Capítulo 2

Estado da Arte

No seguinte capítulo são endereçados alguns métodos de calibração de parâmetros extrínsecos, sendo estes separados em três grupos, nomeadamente, calibração da câmara em relação ao sistema inercial de navegação, ou *Inertial Navigation System* (INS), calibração de um LiDAR em relação a uma câmara, e, por fim, um LiDAR em relação ao INS.

Existem vários procedimentos publicados em relação à calibração de extrínsecos de múltiplas câmaras [2, 3], múltiplos LiDARs [4–9] ou combinações dos dois sensores [10–12].

2.1 Métodos de calibração entre câmara e INS

Mohamed Mostafa refere que é um fator importante que os parâmetros de calibração se mantenham constantes ao longo da calibração, sendo isso garantido desde que os sensores estejam rigidamente posicionados um em relação aos outros [13].

Na mesma publicação, estudou dois algoritmos de calibração dos ângulos de *boresight*, isto é, as diferenças dos eixos rotacionais entre dois sensores. Um deles resume-se em determinar a matriz de orientação da imagem utilizando pontos de controlo no solo, retirar a matriz da unidade de medida inercial, em inglês *Inertial Measurement Unit* (IMU), no instante em que a imagem foi obtida e calcular o desvio angular entre as duas matrizes. O segundo consiste na introdução dos três valores angulares no processo de ajuste em conjunto com todas as imagens obtidas.

2.2 Métodos de calibração entre LiDAR e sistema visível

O estado da arte de calibração de sistemas LiDAR-câmara podem classificados em:

- Calibração com recurso a um alvo: seja este um xadrez típico de calibração de intrínsecos das câmaras [14–17], um alvo triangular [18], um alvo circular com um [19] ou mais círculos [20], ou um alvo de calibração em forma de V [11, 21].
- Calibração sem alvo: tais abordagens baseiam-se em corresponder características encontradas nos dados dos dois sensores, tais como arestas, cantos ou linhas e, depois disso, minimizar o erro de reprojeção associado a elas. Numa primeira abordagem ao problema, poderia levar a pensar que é necessário uma sobreposição do *Field of View* (FOV) dos sensores, tal como acontece em [22–25]. No entanto, Napier et al. em [26] mostra que é possível calibrar sem alvo e sem sobreposição.

2.2.1 Calibração com recurso a alvo

Tal como dito anteriormente, Wasielewski et al. em [21] utiliza um padrão em forma de V, com uma face do padrão de cor preta e a outra de cor branca, tal como é apresentado na figura 2.1.

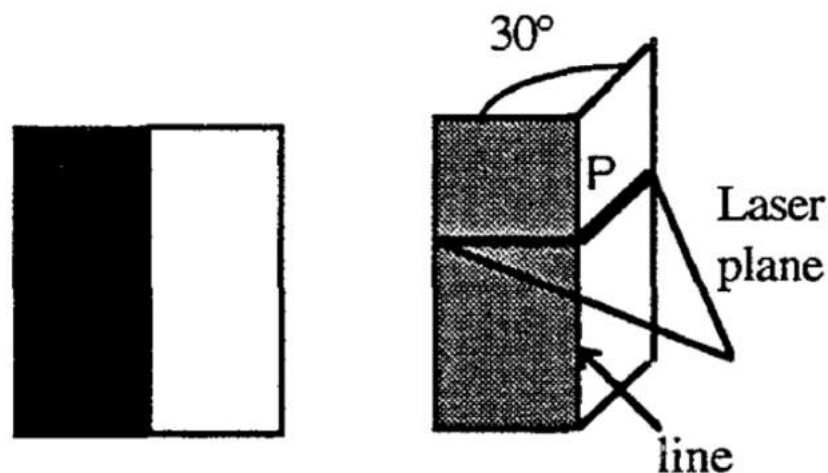


Figura 2.1: Alvo de calibração com vista da câmara e do LiDAR [21]

O formato e a diferença de cor do padrão permite a deteção da aresta tanto na imagem como nos dados do LiDAR e assim fazer a correspondência para determinar os ângulos de *boresight*.

Em [14], o autor utilizou diferentes orientações do alvo no seu método de calibração com um LiDAR de apenas uma linha, isto é, um LiDAR 2D. Inicialmente, esse alvo é paralelo ao plano da imagem e, depois de escolhido aleatoriamente um eixo de rotação, é efetuada uma rotação em torno desse eixo com uma certa amplitude angular, sendo esse processo repetido para 24 orientações diferentes. Apesar do padrão ser facilmente detectado por processamento de imagem, o mesmo não acontece com o sistema de LASER, do inglês *Light Amplification by Stimulated Emission of Radiation*, sendo por isso extraído manualmente.

Na figura 2.2 é possível ver imagens do xadrez com os pontos do LASER projetados.

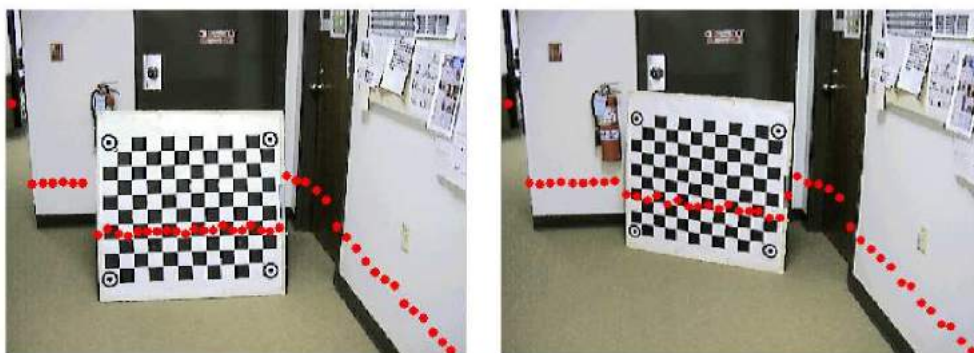


Figura 2.2: Imagens do padrão com a projeção dos pontos do LiDAR 2D [14]

Uma alternativa ao algoritmo anterior foi apresentada em [16], resolvendo o problema da rotação entre sensores como um problema $p3p$ [27], utilizando 3 de N planos detectados. O passo seguinte baseia-se na aplicação do método *RANdom SAmples Consensus* (RANSAC) às diferentes soluções, utilizando $N-3$ planos para determinar qual a melhor.

Tal como nos trabalhos anteriores, Unnikrishnan em [15] utilizou um xadrez para calibração de extrínsecos entre uma câmara e um LiDAR, no entanto, o sistema de laser era 3D, correspondendo a um registo de dados ilustrado na figura 2.3. O algoritmo de calibração requer 20 a 30 pontos do LiDAR no alvo e que os cantos do xadrez estejam perceptíveis na imagem visível. Depois de selecionada manualmente a área correspondente ao alvo, são então calculadas as matrizes de transformação dos dois sensores em relação a este e, com isso, são obtidos os parâmetros extrínsecos dos dois sensores.

O algoritmo desenvolvido por Geiger [17] faz a deteção de planos nos dados do LiDAR para fazer a correspondência com os alvos xadrez utilizados para calibração da câmara. Depois de segmentados os planos, são calculadas hipóteses de transformações entre os dados do LiDAR e da câmara por associação aleatória de planos. Fazem uma avaliação

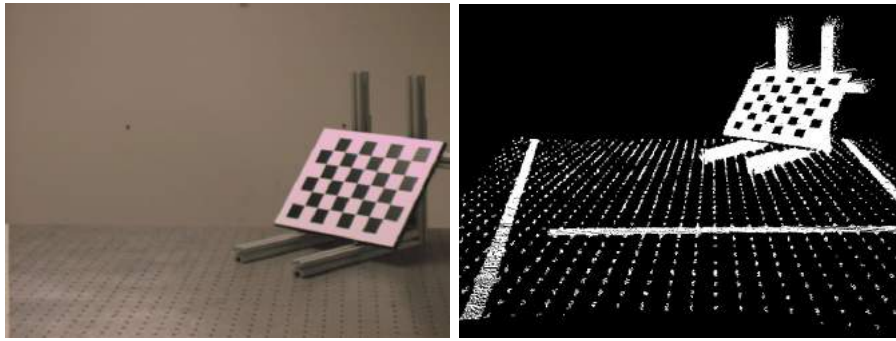


Figura 2.3: Imagem obtida pela câmara à direita, dados do LiDAR com atribuição da cor correspondente.

quantitativa de todas as soluções por distância do centroide do xadrez detetado na imagem após transformação aos pontos vizinhos no plano do LiDAR. Por fim, são suprimidos os resultados semelhantes até chegar a uma única solução e, caso não seja possível, é requerida a interferência do utilizador.

2.2.2 Calibração sem alvo

Nem sempre o alvo é um objeto dedicado à calibração como nas publicações anteriores, por vezes, são estruturas já existentes no ambiente em redor dos sensores.

É o caso do trabalho desenvolvido em [23] que se distingue dos restantes por tirar vantagem de um triedro arbitrário, tal como duas paredes adjacentes e o chão onde estas se encontram. A calibração é formulada como um problema de mínimos quadrados não linear, em inglês *nonlinear least square problem*, e resolvido pelo método Levenberg-Marquardt [28].

Pandey et al. em [22], utilizou o cenário de uma garagem ocupada por veículos para testes *indoor* e, devido aos sensores estarem no topo de um carro, para cenários exteriores utilizava as ruas por onde o veículo circulava. Para resolver a calibração entre os dois sensores, tinham em consideração o valor da intensidade de um ponto, dado pelo LiDAR, e o valor na escala de cinza obtido da projeção desse ponto no campo de visão da câmara e formular assim um problema de informação mútua.

O algoritmo desenvolvido por Levinson et al. em [24] propõe uma solução em tempo real para comparação dos dados do LiDAR e da câmara. O processamento de imagem passa por definir cada pixel como o valor máximo da diferença entre ele e os seus 8 vizinhos. Relativamente aos dados do LiDAR, são considerados como pontos possíveis

de ser uma descontinuidade, aqueles que se encontram mais próximos que pelo menos dois dos seus vizinhos. Ao contrário de uma situação de pós-processamento, em que era realizada uma comparação com todos os dados anteriores, o autor utiliza uma janela temporal para reduzir o tempo computacional e tornar possível o processamento em tempo real.

Napier et al. em [26] utilizam um LiDAR 2D para fazer o varrimento do ambiente em redor a um veículo, fazendo depois a reconstrução 3D utilizando a trajetória percorrida. Com isto, eliminam a necessidade da sobreposição do campo de visão dos dois sensores no momento da obtenção de dados. Relativamente à calibração, o método é bastante semelhante ao do autor anterior.

Em [25], Liao et al. desenvolveram um algoritmo que não necessita sobreposição entre o campo de visão dos dois sensores visto que utilizam a transformação entre um par de dados consequentes de cada sensor individualmente. Utilizam a transformação entre a câmara e o LiDAR em dois instantes diferentes e consecutivos para calcular a relação entre os dois sistemas.

2.3 Métodos de calibração entre LiDAR e INS

As publicações seguintes utilizam apenas os sistemas LiDAR para dar posições de pontos, *Global Position System* (GPS) para posicionamento e INS de modo a obter atitudes. Quando combinados, é possível obter uma nuvem de pontos geo-localizada num referencial global tal como o *Earth-Centered Earth-Fixed* (ECEF).

A equação de geo-localização para este sistema é a seguinte:

$$X_{ECEF} = T_{Nav}^{ECEF} + R_{Nav}^{ECEF} \times R_{INS}^{Nav} \times (T_{LiDAR}^{INS} + R_{LiDAR}^{INS} \times X_{LiDAR}) \quad (2.1)$$

em que:

- X_b é a coordenada $\{X, Y, Z\}$ no referencial b ;
- T_b^n é a transformação linear do referencial b para n ;
- R_b^n é a matriz de rotação do referencial b para n .

Skaloud and Lichti, em [29], desenvolveram um modelo matemático para calibração dos ângulos de *boresight*, assumindo que a transformação linear entre os dois sensores é conhecida e, portanto, retirada dos parâmetros de calibração. Para o desenvolvimento

desse modelo, tiveram em conta a equação 2.1 para cada ponto $X_i = \{x_i, y_i, z_i\}$ e a restrição necessária para que esses pertençam a uma superfície planar comum j , apresentada em 3.6.

$$s_{1_j}x_i + s_{2_j}y_i + s_{3_j}z_i + s_{4_j} = 0 \quad (2.2)$$

De modo a automatizar o processo, em [30] desenvolveram métodos autónomos de remoção de vegetação e deteção de planos.

Em ambas as publicações, foram utilizados telhados de zonas urbanas para obtenção dos planos necessários para a calibração. Devido à existência de edifícios com diferentes orientações, as normais dos planos detetados terão também distintas direções, o que é necessário para uma boa descorrelação dos parâmetros de calibração.

Rieger, em [31], utilizou um processo similar mas adaptado a um veículo terrestre, utilizando a plataforma comercial RiPROCESS para obtenção dos valores de calibração.

Chan et al. em [32] não se baseiam só em planos como os autores anteriores, detetam também linhas aéreas de transporte de energia ou comunicações, na nuvem de pontos. Para isso, utilizam o modelo matemático de uma catenária, que é a representação matemática das linhas em suspensão, desprezando efeitos de forças exteriores tais como o vento.

Em [33], Le Scouarnec propõe um método livre de interferências dos erros de GPS e imune aos bias do INS utilizando vetores de direção ao invés da posição dos pontos obtidos. Os vetores de direção utilizados na calibração, provém de linhas de varrimento, obtidas pelo método de RANSAC. Linhas pertencentes ao mesmo plano têm o seu vetor de direção ortogonal à normal do respetivo plano, respeitando a equação 3.6, tendo como $X_i = \{u, v, w\}$ as componentes vectoriais das linhas. Visto que X_i representa orientações e não posições, deixa de haver a necessidade de posicionamento global.

Morales et al. construíram um LiDAR 3D, utilizando um LiDAR 2D, aplicando uma rotação no seu centro ótico [34]. Visto que a ligação do LiDAR 2D ao sistema de rotação não é ideal, existe uma distorção na nuvem de pontos final. De modo a eliminar a distorção, aplicam a calibração dos parâmetros intrínsecos do LiDAR 3D, sendo estes os extrínsecos do sistema LASER 2D em relação ao sistema de rotação.

O efeito de um valor errado em cada ângulo de *boresight* não influencia de igual forma a reprojeção do ponto lido do LiDAR, tal como demonstra Li et al., em [35]. Com base nessa observação, o autor destaca o problema da escolha dos planos para a calibração, dando o exemplo de dois varrimentos paralelos em que num deles, o sensor passa exatamente na vertical no vértice de um telhado e, conseqüentemente, no segundo varrimento, o mesmo telhado fica na oblíqua em relação ao sensor. Na figura 2.4 é

ilustrado o efeito de uma má calibração no caso anterior, em que a posição real do telhado são as linhas a preto, é representada a vermelho a reprojeção dos pontos do primeiro varrimento, enquanto que os pontos a azul referem-se ao segundo varrimento.

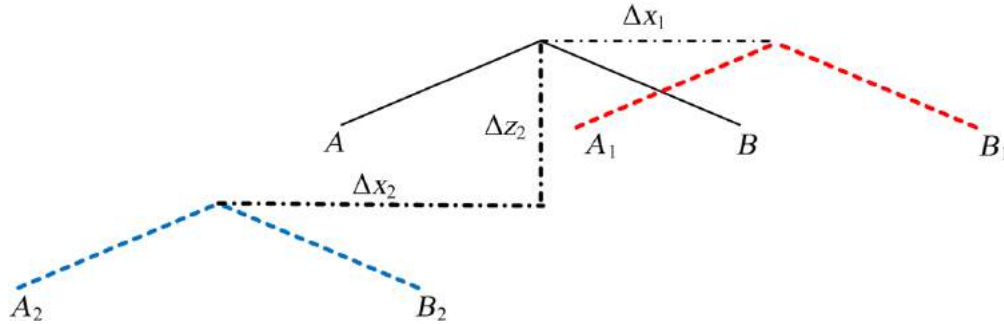


Figura 2.4: Influência dos erros nos ângulos de *boresight* em dois voos paralelos. A_1 , B_1 e A_2 , B_2 são as reprojeções do telhado A , B do primeiro e segundo varrimento, respetivamente. Δx e Δz representam os erros em posição nos respetivos eixos [35]

Esta página foi intencionalmente deixada em branco.

Capítulo 3

Fundamentos Teóricos

Neste capítulo serão apresentados métodos e conceitos necessários para uma melhor compreensão dos capítulos seguintes, **mais propriamente, fundamentos associados à representação de transformação de referenciais, métodos de estimação de parâmetros de planos**

3.1 *Earth-Centered Earth-Fixed*

O referencial ECEF é um sistema de coordenadas cartesiano em que a origem se situa no centro de massa do planeta Terra, daí se obtém *Earth Centered*. Este mantém-se fixo à rotação da Terra, isto é, um ponto representado nesse referencial sofre a mesma rotação que o planeta e por isso é *Earth Fixed* [36].

O eixo Z tem como sentido positivo o polo norte e o plano XY é coplanar com o plano equatorial. Relativamente ao eixo X, o sentido positivo deste aponta para a interseção com o meridiano de Greenwich com a linha do equador. Por fim, o eixo Y segue a regra da mão direita, apontando para o ponto com a longitude de 90° .

3.2 Transformação de referenciais

Na equação 2.1 vemos a transformação de um ponto do LiDAR para um ponto no referencial ECEF. Para obter tal transformação, são necessárias matrizes de movimentos angulares e de movimentos lineares entre referenciais.

3.2.1 Transformação Linear

Este último é representado por um vetor de dimensão 3 visto que se trata de um movimento em três dimensões, tal como se pode ver em 3.1.

$$T_b^n = [X, Y, Z]' \quad (3.1)$$

3.2.2 Transformação Angular

Num mundo tridimensional, qualquer orientação pode ser obtida pela composição de três rotações a partir de um referencial conhecido. De igual forma, qualquer matriz de rotação pode ser decomposta em três matrizes, como é representado em 3.2.

$$R = R_x(\alpha)R_y(\beta)R_z(\gamma) \quad (3.2)$$

No contexto desta dissertação, são utilizados os ângulos de Euler, *roll* (α), *pitch*, (β) e *yaw* (γ) [37] para descrever as orientações dos sensores, sendo que representam a rotação sobre os eixos x , y e z respetivamente, tal como se pode observar na figura 3.1

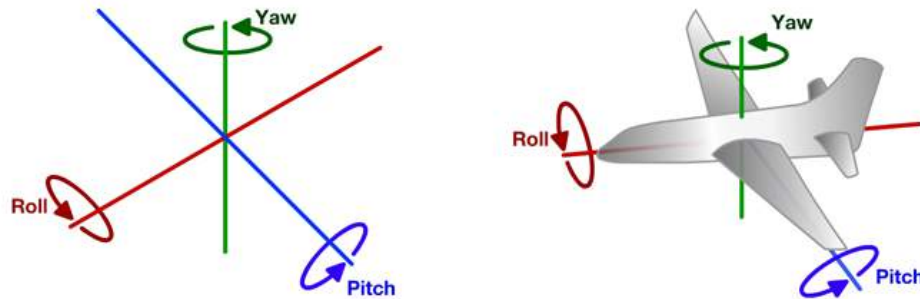


Figura 3.1: Ângulos de euler aplicados na navegação de um UAV [38]

A rotação de um ângulo α sobre o eixo dos xx é representada na equação 3.3.

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (3.3)$$

No caso do eixo dos yy , a rotação de um ângulo β corresponde à equação 3.4.

$$R_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & -\sin(\beta) \\ 0 & 1 & 0 \\ \sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \quad (3.4)$$

Por fim, numa rotação de um ângulo γ sobre o eixo dos zz , utiliza-se a equação 3.5.

$$R_z(\gamma) = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.5)$$

3.3 Equação geral do Plano

Seja α um plano com o vetor normal $\vec{n} = (a, b, c)$ e um ponto P de coordenadas (x, y, z) representado na figura 3.2

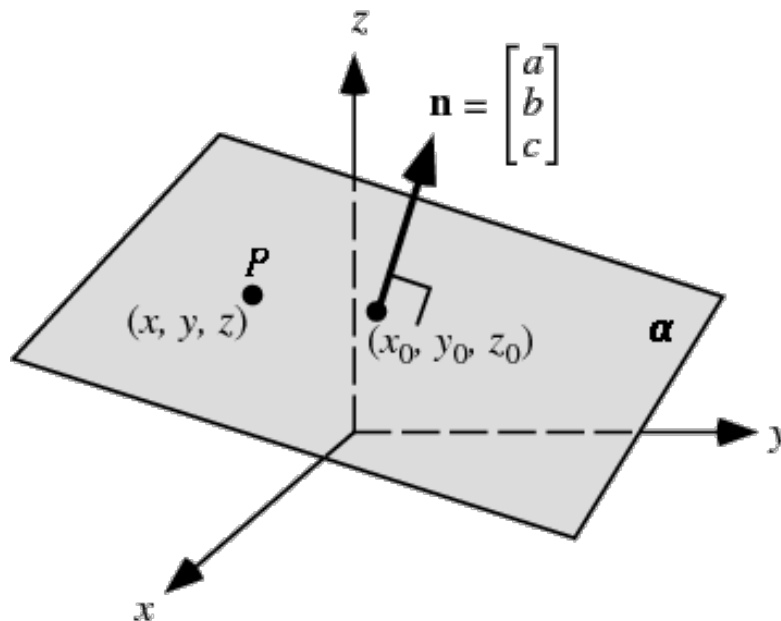


Figura 3.2: Ponto P num plano α com vetor normal $\vec{n} = (a, b, c)$

A equação 3.6 permite descrever qualquer plano utilizando 4 parâmetros, nomeadamente $\alpha = (a, b, c, d)$, sendo a, b e c os valores do seu vetor normal e d a distância do plano à origem do referencial.

$$ax + by + cz + d = 0 \quad (3.6)$$

Para calcular o valor d , recorre-se a um dos pontos pertencentes ao plano, tal como o ponto P no exemplo anterior, e resolve-se a equação 3.6 em ordem a d , resultando na equação 3.7.

$$d = -ax - by - cz \quad (3.7)$$

Depois de definidos todos os parâmetros do plano, é possível verificar se um ponto pertence ou não a este, verificando se o produto escalar entre as coordenadas do ponto a verificar e dos parâmetros do plano é nulo.

O produto escalar [39, 40] entre dois vetores \vec{a} e \vec{b} de dimensão n é definido pela equação 3.8.

$$\vec{a} \cdot \vec{b} = \sum_{i=0}^n a_i b_i = a_1 b_1 + a_2 b_2 + \dots + a_n b_n \quad (3.8)$$

Como se pode constatar, ambos os vetores têm que ter a mesma dimensão, o que não acontece entre as coordenadas de um ponto no espaço e os parâmetros de um plano. Para que isso seja possível, um ponto $Q = (x, y, z)$ é representado em coordenadas homogêneas $Q' = (xw, yw, zw, w)$, onde w toma o valor unitário, resultando no ponto $Q' = (x, y, z, 1)$.

Com esta transformação de coordenadas, torna-se possível calcular o produto escalar entre o ponto Q' e o plano α , utilizando a equação 3.8, resultando na equação 3.6 como se pode verificar em 3.9.

$$\alpha \cdot Q' = ax + by + cz + d(1) = 0 \quad (3.9)$$

3.4 RANSAC

O objetivo principal do algoritmo *RANdom SAmple Consensus* (RANSAC) [41] é calcular parâmetros de modelos matemáticos (planos, cones, cilindros etc.) a partir de amostras de conjuntos de dados. A escolha destas é aleatória, daí se chamar *Random Sample*. Após a primeira amostra recolhida, são estimados os parâmetros com base nos dados contidos nesta e, posteriormente, é verificado se esses parâmetros são os mais indicados para os restantes pontos da amostra. A estimação que apresentar maior consenso, em inglês *Consensus*, entre os dados é considerada como solução.

A correspondência de pontos a *features* a detetar é muitas vezes dificultada pela existência de *outliers*, isto é, pontos que não fazem parte dessa *feature*. Uma das funções do algoritmo RANSAC é filtrar esses *outliers* na estimação de parâmetros dos modelos matemáticos das *features*.

Um esboço do algoritmo do RANSAC é apresentado em algoritmo 1.

Algoritmo 1 : RANSAC

Seja:

S: todo o conjunto de dados

d: distância máxima dos pontos ao modelo

M: modelo matemático a detetar

repita

Escolhe uma amostra aleatória de pontos do conjunto S

Estima um modelo M para os pontos escolhidos

Calcula a distância para todos os pontos de S

Cria o conjunto de *inliers*, ou seja, todos os pontos cuja distância ao modelo seja menor que d

até número máximo de iterações seja alcançado

Seleciona o conjunto com maior número de *inliers* como uma solução ao problema

Estima os parâmetros do modelo, utilizando desta vez todos os *inliers*

Como se pode observar do algoritmo 1, o RANSAC é um método iterativo e não determinístico, ou seja, calcula uma diferente solução em cada execução do algoritmo. Facilmente se conclui que a solução tende a melhorar quanto maior for o número de iterações.

3.5 LiDAR

Mostrar ambientes típicos com scans; Velodyne - Strong, Last, Both return

3.5.1 Scan mechanisms

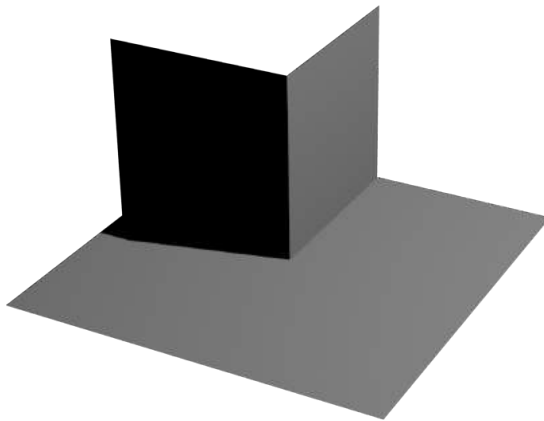
Transformação

3.6 Influências de erros nos parâmetros

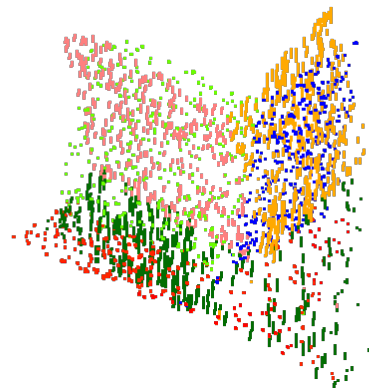
A existência de erros na atitude e posição do LiDAR em relação ao UAV pode resultar em diferenças significativas na reprojeção da nuvem de pontos para diferentes atitudes deste.

Nas figuras 3.3(b), 3.3(c) e 3.3(d) é representado a vista isométrica, vista frontal e vista lateral direita, respectivamente, da nuvem de pontos referente aos planos apresentados em 3.3(a).

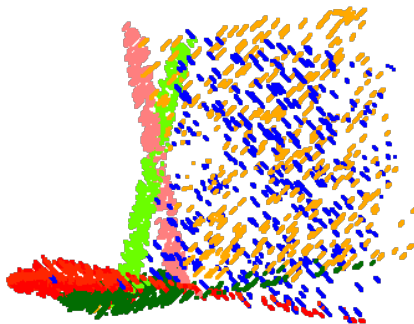
A nuvem de pontos ilustrada provém de dois voos paralelos mas com sentidos opostos e, conseqüentemente, diferentes atitudes do UAV. O primeiro voo deu resultado aos planos azul, verde claro e vermelho e o segundo resultou nos planos amarelos, rosa e verde escuro.



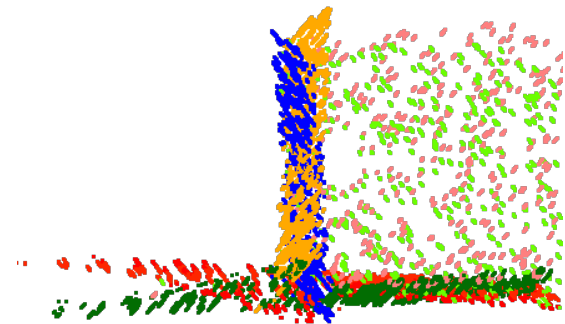
(a) Planos alvo



(b) Perspectiva isométrica da nuvem de pontos



(c) Vista frontal da nuvem de pontos



(d) Vista lateral direita da nuvem de pontos

Figura 3.3: Representação de três planos alvo perpendiculares entre si a) e a sua re-projeção em três vistas diferentes b), c) e d).

Capítulo 4

Algoritmo

Neste capítulo será descrita a abordagem tomada para a calibração dos parâmetros extrínsecos do LiDAR.

O método proposto é dividido em cinco tarefas: 1) Associação temporal dos dados do LiDAR e os valores de posição e atitude do UAV que o transporta; 2) Divisão dos pontos do LiDAR por segmentos de voo; 3) Construção da nuvem de pontos geo-referenciada; 4) Detecção e identificação de planos na nova nuvem de pontos; 5) Estimação dos parâmetros que melhor aproximam os planos que se sobrepõem em diferentes segmentos de voo.

Uma descrição geral da abordagem é apresentada na figura 4.1 em forma de fluxograma.

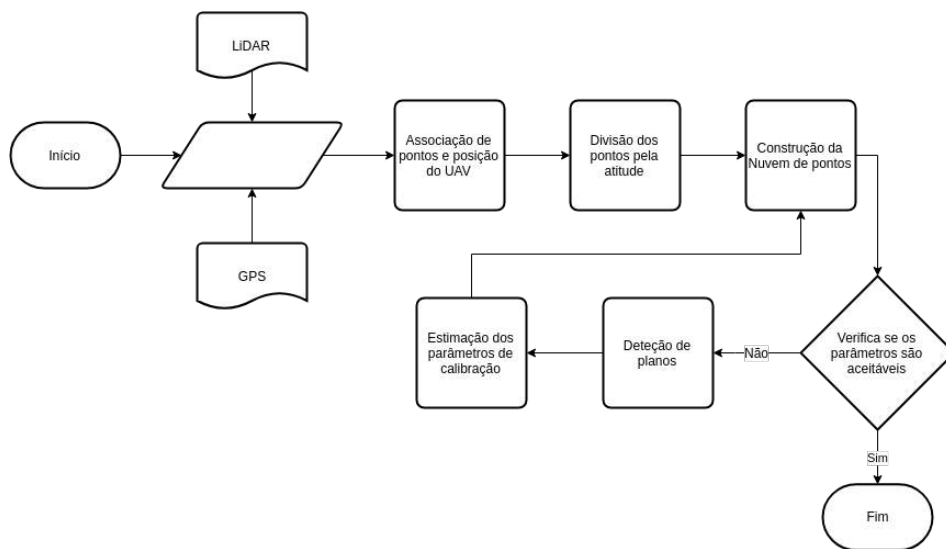


Figura 4.1: Algoritmo para a calibração dos parâmetros extrínsecos do LiDAR

4.1 Associação de pontos a Posição do UAV

Começa-se por ler o ficheiro que contém todas as posições e atitudes do UAV e o ficheiro que contém os pontos obtidos pelo LiDAR.

De seguida, faz-se uma combinação de dados, isto é, associa-se uma posição e atitude a cada ponto dado pelo LiDAR, de modo a ser possível utilizar a equação 2.1 para obter uma nuvem de pontos geo-referenciada.

4.2 Divisão da Nuvem de Pontos pela atitude do UAV

Como se concluiu do sub-capítulo 3.6, valores errados nos parâmetros afetam a re-projeção de planos e, conseqüentemente, dificultam a deteção de planos.

De modo a facilitar tal tarefa, é realizada uma divisão dos dados por direções de voo, ou seja, dependendo do seu valor de atitude. Quando aplicada ao exemplo da figura 3.3, resulta nas re-projeções ilustradas na figura 4.2.

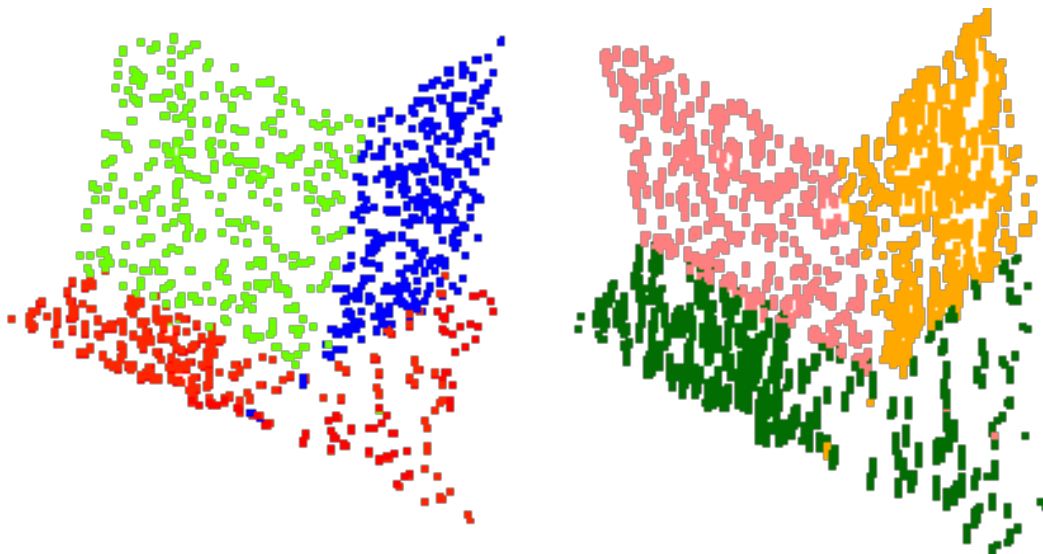


Figura 4.2: Re-projeção de planos após divisão dos dados. À direita a nuvem de pontos obtida para um sentido de voo e à esquerda para o sentido oposto.

De notar que ambas as imagens foram tiradas com a mesma perspetiva.

4.3 Construção da Nuvem de Pontos

Para cada segmento da nuvem de pontos, é aplicada a equação 2.1, transformando uma nuvem de pontos no referencial do LiDAR numa nuvem de pontos geo-referenciada. Para isso, é necessário obter uns parâmetros iniciais para a posição relativa do LiDAR ao INS do UAV de modo a calcular as matrizes R_{LiDAR}^{INS} e T_{LiDAR}^{INS} . O processo de aquisição dos parâmetros iniciais pode dar-se por uma simples medição visual da posição dos sensores.

É possível observar a nuvem de pontos antes e depois da transformação na figura 4.3.

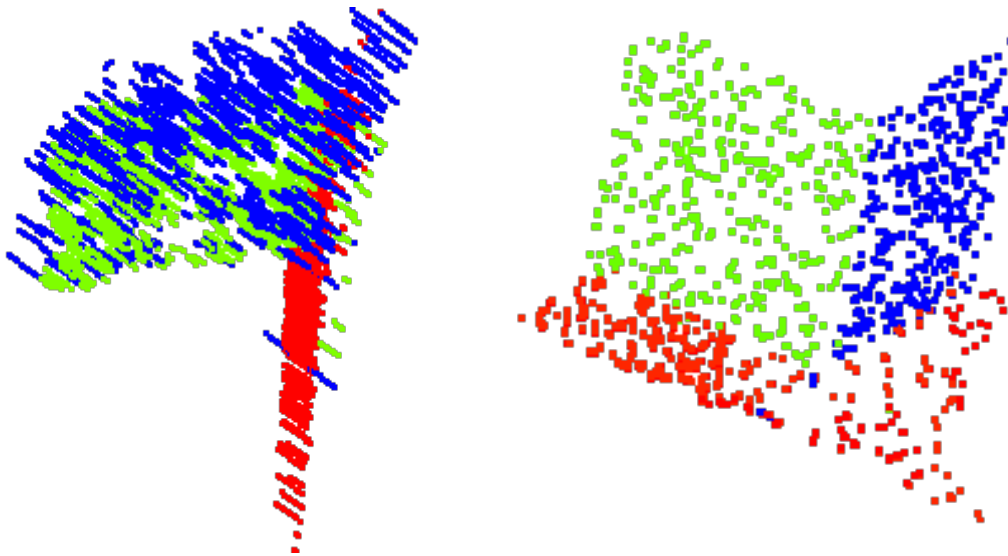


Figura 4.3: Representação da nuvem de pontos no referencial do LiDAR (a) e geo-referenciada (b).

4.4 Detecção de planos na Nuvem de pontos

A tarefa descrita neste sub-capítulo será a de maior importância pois é com base nos resultados desta que a calibração dos parâmetros extrínsecos do LiDAR irá ser realizada.

O processo de encontrar planos numa nuvem de pontos tem sido um problema fundamental na criação de modelos 3D. Uma das soluções é a transformada de Hough, que procura por objetos de uma determinada geometria através de uma acumulação de parâmetros bem sucedidos [42]. No contexto desta dissertação, as nuvens de pontos a serem utilizadas na calibração tem, geralmente, centenas de milhares de pontos o que

iria levar a um tempo de execução e consumo de memória intoleráveis no processo de calibração.

No intuito de utilizar uma abordagem mais heurística e robusta, foi utilizado o algoritmo RANSAC. Em cada execução deste, um ponto e alguns dos seus vizinhos são escolhidos aleatoriamente e, com base nessa seleção, são calculados os parâmetros a , b , c e d do plano que possui menor distância acumulada entre ele e os pontos selecionados.

Este plano é então comparado com todos os pontos da nuvem de pontos utilizando a equação 3.9 e são selecionados todos aqueles que apresentem o resultado dentro de um erro aceitável. De seguida é realizado o último passo do algoritmo RANSAC que passa por uma otimização dos parâmetros obtidos utilizando o conjunto de pontos anterior de modo a reduzir o erro quadrático médio.

Os pontos pertencentes ao plano estimado são então removidos da nuvem de pontos e associados ao novo elemento da lista de planos.

4.5 Estimação dos parâmetros de calibração

Após a análise de todos os segmentos de voo, obteve-se uma lista de planos e os seus respetivos pontos por cada segmento. O passo seguinte consiste em realizar a associação de cada plano entre segmentos, isto é, identificar que planos correspondem à mesma área de varrimento.

Devido à mudança de atitude, nem sempre é possível obter a mesma área de cobertura de um plano para os diferentes sentidos de voo, normalmente devido à oclusão causada por outros objetos. Por esse motivo, a associação não pode ser realizada por localização do centroide do plano.

Dois planos coplanares com centroides diferentes possuem a mesma normal e a mesma distância à origem do referencial escolhido. Devido a esta propriedade, são utilizados os parâmetros a , b , c e d para a associação entre planos. Depois de escolhido um elemento da lista de planos detetados, são identificados aqueles que apresentem a normal com uma pequena variação angular relativa, utilizando o produto escalar, como é descrito na equação 4.1.

$$\cos(\theta_a - \theta_b) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}||\vec{b}|} \quad (4.1)$$

Visto que o módulo do vetor normal é unitário, a diferença angular entre dois planos pode ser calculada pela equação 4.2

$$\theta_a - \theta_b = \arccos(\vec{a} \cdot \vec{b}) \quad (4.2)$$

Esta última comparação não é capaz de distinguir planos paralelos mas não coplanares. Para isso é utilizada a distância à origem do referencial, isto é, aqueles que se encontrem próximos do plano a comparar, são associados a este.

Esta associação é realizada até que todos os planos sejam testados. Planos que não sejam associados a outros, são eliminados do processo de calibração pois não contribuem para a mesma.

Como se pode verificar nas figuras 3.3(c) e 3.3(d), a existência de erro nos parâmetros extrínsecos do LiDAR leva a um efeito simétrico na reprojeção dos pontos. Com base nessa observação, são calculados os planos representados pela média dos parâmetros entre planos associados.

Por fim, depois de obtidos todos os planos que aparecem em pelo menos dois segmentos de voo, é utilizado o produto escalar entre esses planos e os pontos pertencentes a cada um. Essa operação é descrita na equação 4.3 sendo que s representa os argumentos de um plano, S o conjunto de planos e X_{ECEF} o conjunto de pontos que pertencem ao plano s .

$$\forall s \in S, s \cdot X_{ECEF} = 0 \quad (4.3)$$

Para isso é preciso rescrever a equação 2.1 de modo a utilizar as matrizes estimadas de translação e rotação do LiDAR para o INS do veículo, obtendo a equação 4.4.

$$X_{ECEF} = T_{Nav}^{ECEF} + R_{Nav}^{ECEF} \times R_{INS}^{Nav} \times (\hat{T}_{LiDAR}^{INS} + \hat{R}_{LiDAR}^{INS} \times X_{LiDAR}) \quad (4.4)$$

De modo a obter as matrizes estimadas, é necessário um método capaz de resolver problemas *nonlinear least squares*. Para isso utilizou-se o clássico algoritmo de Levenberg-Marquardt, definindo os valores iniciais do algoritmo como os valores utilizados anteriormente para a reprojeção da nuvem de pontos.

Esta página foi intencionalmente deixada em branco.

Capítulo 5

Implementação

5.1 Arquitetura de hardware

De modo a obter um varrimento o mais próximo possível da realidade, houve a necessidade de estudar os detalhes referentes aos sensores. Neste sub-capítulo serão descritos os elementos sensoriais relevantes para o contexto desta dissertação.

5.1.1 UAV

Estes sensores estão rigidamente colocados no UAV STORK, ilustrado na figura 5.1.

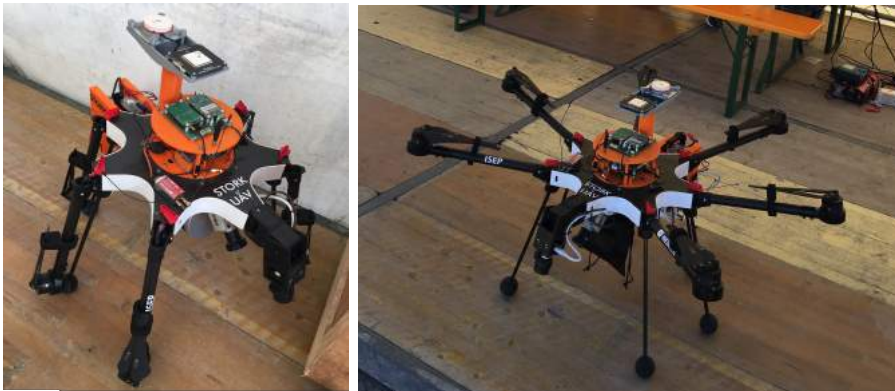


Figura 5.1: UAV STORK

Esta plataforma é um hexa-rotor desenvolvido pelo INESC TEC e pelo LSA tendo como principal diferença, em relação a estruturas convencionais, a posição dos braços que respeita a forma geométrica de um heptágono. O vértice sem nenhum braço associado

representa um espaço extra para a colocação de *payload* tal como um conjunto de câmara visível e térmica, por exemplo.

O STORK tem como 900 mm de diâmetro e uma capacidade de *payload* de 4.9 kg, conseguindo voar 25 minutos contínuos. Tem como principais aplicações a inspeção de ativos elétricos, busca e salvamento, monitoramento ambiental e mapeamento 3D.

5.1.2 Velodyne VLP-16

O sensor LiDAR a bordo do STORK é o Velodyne VLP-16, ilustrado na figura 5.2. É um LiDAR de pequena dimensão capaz de retornar dados em tempo real, em três dimensões e em todo o redor do sensor.



Figura 5.2: Velodyne VLP-16

Na tabela 5.1 encontram-se as características relevantes do Velodyne VLP-16 que serão importantes para a obtenção de dados simulados.

Tabela 5.1: Características do Velodyne VLP-16

Alcance Máximo	100 m
<i>Field of View</i> (Vertical)	+15.0° to -15.0° (30°)
Resolução Angular (Vertical)	2.0°
<i>Field of View</i> (Horizontal)	360°
Resolução Angular (Horizontal/Azimuth)	0.1° – 0.4°
Frequência de dados	~ 300000 Pontos/s

5.1.3 INS

O STORK possui dois sistemas inerciais, sendo um de baixo custo constituído por sensores inerciais e o módulo GNSS u-blox M8T [43] e outro sistema de alta precisão

contendo um STIM300 [44] e o módulo GNSS K501 [45].

Quando pós-processados os dados do sistema de alta precisão, obtém-se um registo de dados com uma frequência de 1000 Hz com valores de *Roll Pitch Yaw* (RPY), posição em Latitude, Longitude e Altitude e ECEF.

5.2 Correlação temporal dos dados

Como podemos concluir dos valores apresentados anteriormente, a frequência de dados não coincidem o que requer uma correlação temporal entre os dados dos dois sistemas. Nesse sentido, foi realizada uma análise de um registo de dados do sensor com menor frequência, o INS.

Na figura 5.3 encontra-se um excerto de um voo em que podemos observar uma variação angular aproximadamente constante entre os segundos 10 e 12.

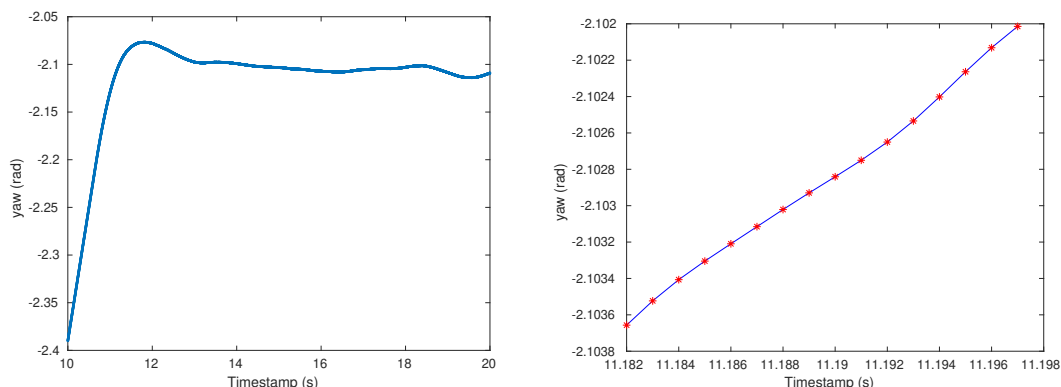


Figura 5.3: Análise de um registo de dados yaw. À direita encontra-se um excerto de 10 segundos de um registo de dados. À esquerda é ilustrado uma pequena janela temporal desse excerto.

No gráfico à direita, vemos que a variação entre dois dados consecutivos, representados a vermelho, pode ser aproximada a uma reta entre eles, sendo esta representada pelas linhas azuis. Com base nessa aproximação, é estimada a equação da reta $y = mx + b$ utilizando as equações em 5.1 sendo que t_2 , t_1 , y_2 e y_1 são os valores de *timestamp* e *yaw* de dois dados consecutivos, respetivamente.

$$\begin{aligned} m &= \frac{y_2 - y_1}{t_2 - t_1} \\ b &= y_2 - m t_2 \end{aligned} \quad (5.1)$$

Com isto, é possível fazer uma aproximação temporal do valor de atitude para cada ponto retornado pelo Velodyne, utilizando sempre o valor de atitude registado imediatamente antes e depois do *timestamp* do ponto do LiDAR.

5.3 Biblioteca PCL

Para processamento da nuvem de pontos do LiDAR, é utilizada a biblioteca *Point Cloud Library* (PCL) [46] pois esta contém funções para segmentação e estimação de parâmetros de modelos matemáticos conhecidos, tais como planos, linhas, esferas, entre outros para além do seu código poder ser usado em C++.

Permite também realizar uma redução de dados da nuvem de pontos original, também referido como *downsampling*, sendo esta tarefa útil para a diminuição do tempo de processamento da nuvem de pontos. Esta função é opcional e a sua utilização é definida pelo utilizador.

5.3.1 RANSAC

Esta biblioteca possui o algoritmo de RANSAC e alguns modelos matemáticos integrados. No entanto, também é possível utilizar variantes ao RANSAC tais como LMEDS - *Least Median of Squares*, MSAC - *M-Estimator Sample Consensus*, RRANSAC - *Randomized RANSAC*, RMSAC - *Randomized MSAC*, MLESAC - *Maximum Likelihood Estimation Sample Consensus* e PROSAC - *PROgressive Sample Consensus*.

Independente da variante escolhida, é necessário uma configuração inicial de alguns parâmetros.

5.3.1.1 Tipo de Modelo

Refere-se ao modelo matemático que o algoritmo irá estimar. Este pode ser definido como um modelo criado pelo utilizador ou como os modelos comuns tais como planos, esferas, etc. No contexto desta dissertação, o uso RANSAC serve para a deteção de planos, sendo por isso esse o modelo definido.

5.3.1.2 Número de iterações

Este valor pode ser obtido de modo empírico tendo como ponto de paragem o tempo total de processamento, a qualidade da estimação obtida ou tendo em conta ambos os fatores.

5.3.1.3 *Threshold* de distância

Com este parâmetro, é dito ao algoritmo qual a distância máxima para um ponto pertencer ao plano.

Por vezes, são observadas pelo LiDAR superfícies que apresentam alguma irregularidade tal como o solo em ambientes naturais, paredes rugosas, entre outros. Portanto, para que cada uma das superfícies mencionadas seja identificada como um único plano, este parâmetro deve ser definido com um valor superior a essa irregularidade.

Esta página foi intencionalmente deixada em branco.

Capítulo 6

Resultados e Análise

Neste capítulo serão apresentados alguns resultados de três cenários diferentes, um simulado e dois reais.

6.1 Cenário simulado

No intuito de avaliar o algoritmo descrito no capítulo 4, utilizou-se um conjunto de 3 superfícies virtuais, ilustradas na figura 6.1, de modo a simular erros na posição do LiDAR.

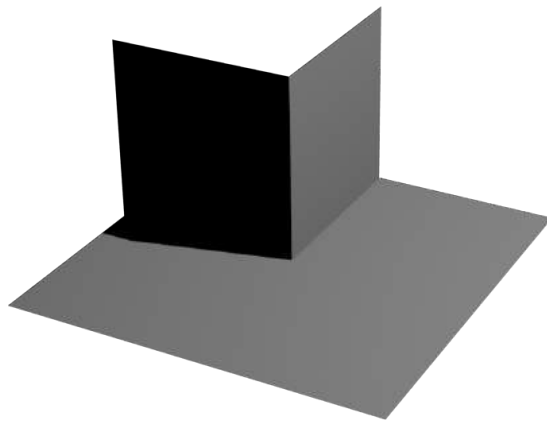


Figura 6.1: Superfícies virtuais utilizadas para obter dados simulados.

A principal vantagem do uso de planos simulados, em relação a um conjunto de dados reais, está na possibilidade de obter uma medida quantitativa do erro final do algoritmo. Ou seja, numa situação real, não há valores de referência para a posição do sensor logo,

não é possível medir a precisão do algoritmo proposto. Nessa situação, apenas numa verificação visual da reprojeção da nuvem de pontos pode determinar o nível qualitativo da calibração executada.

6.1.1 Trajetória e Parâmetros reais

Foi simulada uma trajetória linear de direção constante mas de duplo sentido, tal como é apresentado na figura 6.2. O UAV iniciou a trajetória no ponto (0, 0) deslocando-se no sentido do ponto marcado pelo asterisco vermelho, invertendo depois o sentido de modo a deslocar-se para o asterisco azul.

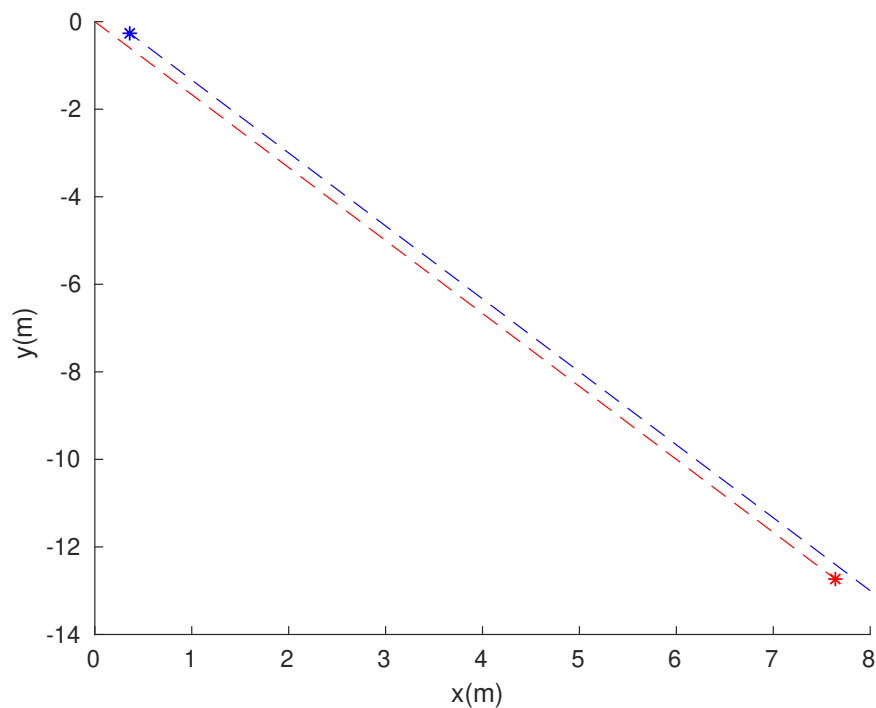


Figura 6.2: Trajetória executada na simulação.

Após a obtenção de uma trajetória, foi definido, para o veículo, uma atitude de 0° em *roll* e 10° em *pitch*. O valor de *yaw*, obteve-se pelo arco-tangente do declive da trajetória, isto é $\text{atan2}(-5, 3) \approx -59^\circ$ para o deslocamento a vermelho e $\text{atan2}(5, -3) \approx -121^\circ$ para o movimento em sentido contrário.

Na tabela 6.1 encontram-se os parâmetros extrínsecos do LiDAR utilizados para a obtenção da nuvem de pontos. Estes são similares à posição do Velodyne em relação ao

STORK e, assim, é possível utilizar os mesmos valores iniciais nos dois tipos de cenário.

Tabela 6.1: Parâmetros extrínsecos do LiDAR simulado

Roll	5°	X	0.05 m
Pitch	77°	Y	0.10 m
Yaw	95°	Z	0.30 m

6.1.2 Resultados da Calibração

O algoritmo de RANSAC foi definido para detetar planos em qualquer orientação e tamanho, desde que possua mais do que 20 pontos, utilizando 500 iterações e aceitando como ponto pertencente ao plano aquele que estiver a uma distância menor que 7 cm.

Tal como foi dito no sub-capítulo 4.3, é necessário obter uns parâmetros extrínsecos iniciais de modo a obter planos reprojitados. Deste modo, foram utilizados os valores contidos na tabela 6.2.

Tabela 6.2: Parâmetros iniciais da calibração do LiDAR

Roll	0°	X	0.0 m
Pitch	75°	Y	0.0 m
Yaw	90°	Z	0.36 m

Ao fim de 36 iterações, obteve-se um erro absoluto angular entre $[0.0006^\circ$ e $0.041^\circ]$ e erro absoluto linear entre $[0.7$ e $13]$ cm, resultando nos valores apresentados na tabela 6.3.

Tabela 6.3: Resultados de 36 iterações consecutivas

Roll	4.999°	X	0.126 m
Pitch	77.021°	Y	0.107 m
Yaw	95.041°	Z	0.430 m

A cada iteração executada, foi registado os valores dos parâmetros extrínsecos de modo a poder ilustrar na figura 6.3 a sua evolução, mostrando uma rápida convergência do valor de *roll* contrastando com os restantes parâmetros extrínsecos.

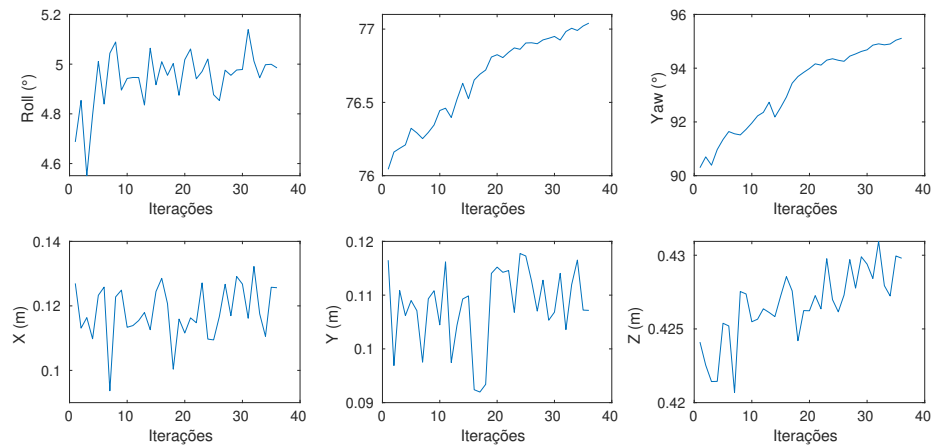


Figura 6.3: Evolução do erro dos parâmetros extrínsecos

Na figura 6.4 encontra-se o resultado em diagrama de caixa, em inglês *box plot* [47], do produto escalar entre os pontos do LiDAR e o seu plano correspondente. É possível observar que após duas iterações os pontos encontram-se a uma distância inferior ao valor definido no algoritmo do RANSAC, o que poderá indicar que este possa ser diminuído.

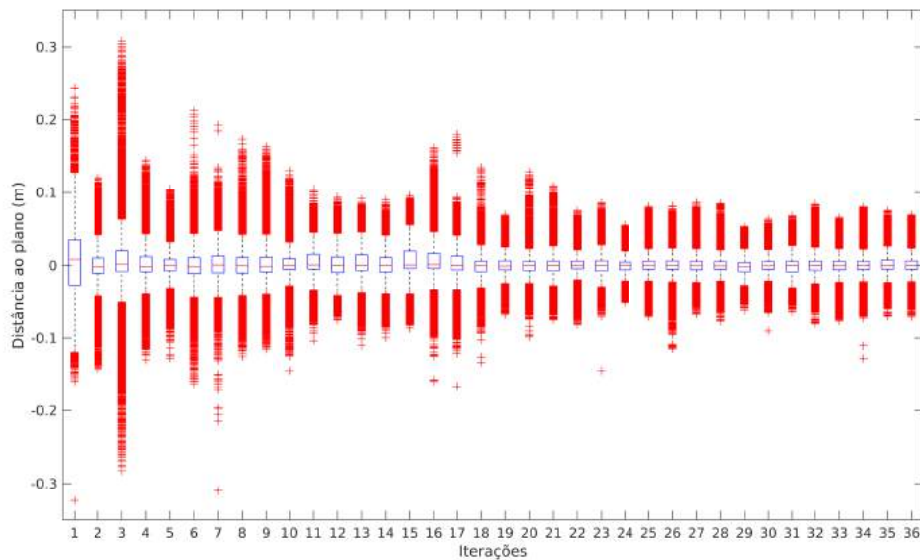


Figura 6.4: Evolução do erro da distância de todos os pontos ao plano respetivo

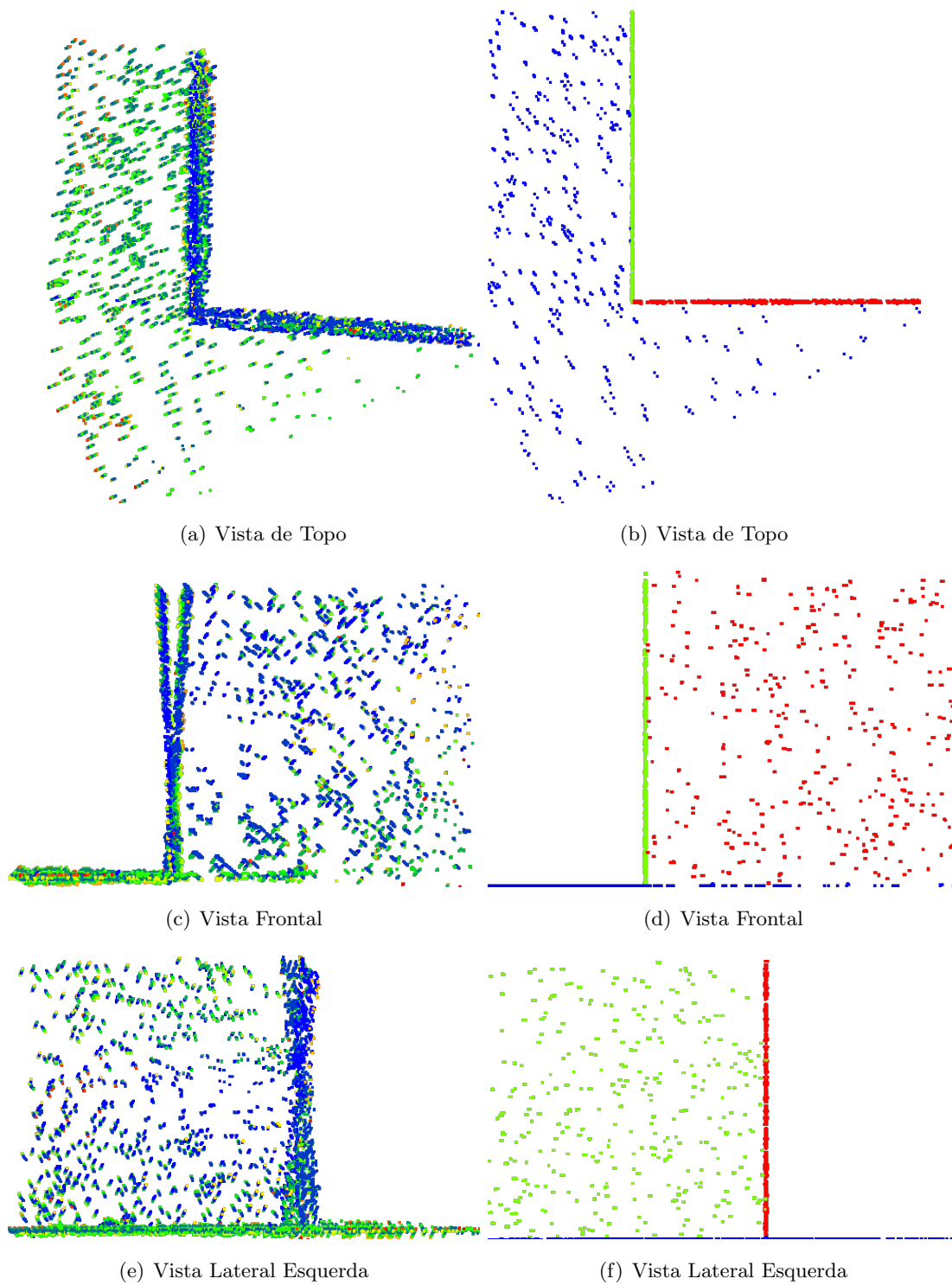


Figura 6.5: Comparação visual dos planos reprojutados. À direita encontram-se os planos antes da calibração e à esquerda o após calibração.

Com o objetivo de efetuar uma comparação visual, na figura 6.5 encontram-se lado a lado o antes e depois da nuvem de pontos com 3 vistas diferentes. As cores apresentadas contém o gradiente Azul-Verde-Amarelo-Vermelho e representam o ID de cada plano. Pode-se observar que nas figuras à direita, correspondentes ao após calibração, consegue-se visualizar as cores azul e vermelho o que corresponde a uma detecção de 3 planos apenas.

6.1.3 Alteração de Altura

Uma segunda simulação foi realizada, alterando a altura da trajetória no segundo sentido de voo para 10 metros acima da trajetória anterior. Na figura 6.6 é ilustrada a trajetória descrita pelo UAV nesta segunda simulação.

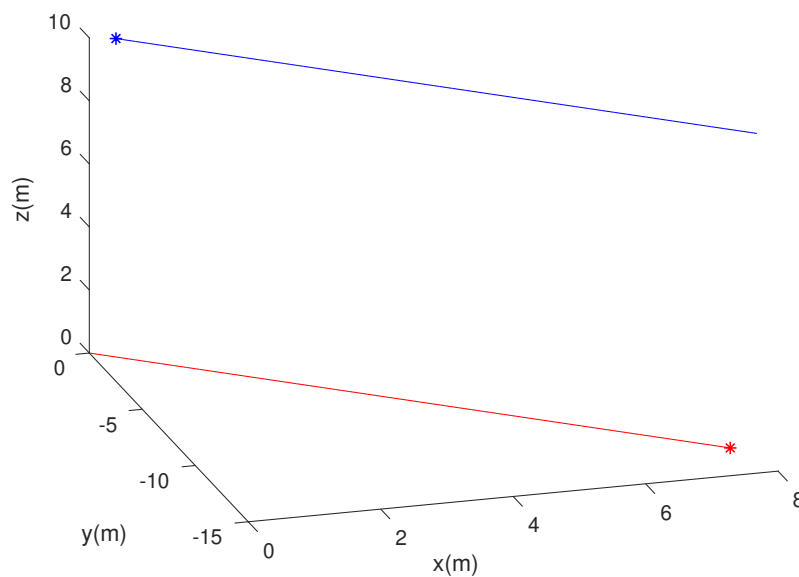


Figura 6.6: Trajetória executada na simulação.

Analisando a figura 6.7, verifica-se que os parâmetros resultantes da estimação convergem para os valores reais. No entanto, observa-se uma convergência que requer menor número de iterações ao invés da simulação 6.1.1.

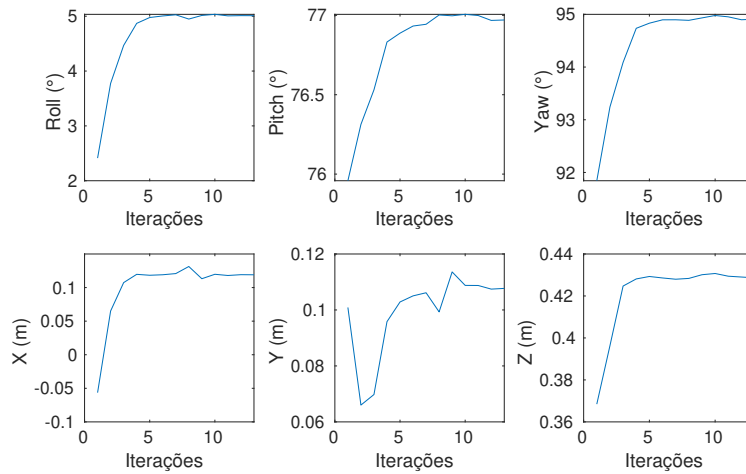


Figura 6.7: Evolução do erro dos parâmetros extrínsecos

É também possível concluir uma maior celeridade de convergência observando o valor de dispersão dos pontos dos planos na figura 6.8.

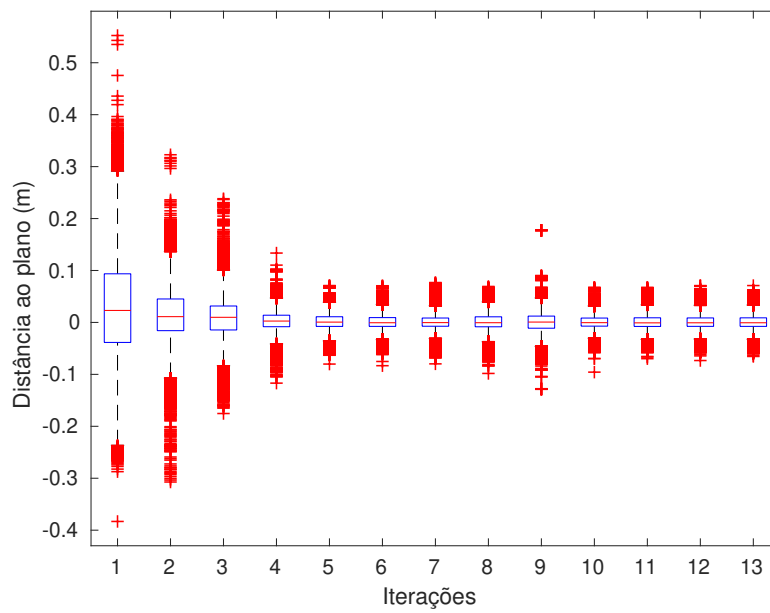


Figura 6.8: Evolução do erro da distância de todos os pontos ao plano respetivo

6.2 Dados reais

6.2.1 ERL Emergency Robots 2017

Os resultados seguintes provém da prova da equipa ISEP/INESC TEC Aerial Robotics Team na competição ERL Emergency Robots [48], em Piombino, Itália. Esta foi inspirada no desastre de Fukushima e envolveu cooperação entre robôs aéreos, terrestres e marinhos. Na figura 6.9 encontra-se a Torre del Sale, edifício que o UAV teve que sobrevoar para detetar possíveis entradas para o robô terrestre e, ao mesmo tempo, realizar um mapeamento do cenário ao seu redor.



Figura 6.9: Torre del Sale [Fonte: STORK]

6.2.2 FADEUP

Neste sub-capítulo serão apresentados os resultados obtidos num voo efetuado no campo de futebol da *Faculdade de Desporto da Universidade do Porto* (FADEUP). Na figura 6.10 é possível observar os diferentes objetos existentes no recinto desportivo, tais como, balizas, redes de proteção e vegetação ao seu redor.

Tal como na simulação efetuada, foi realizado um planeamento de voo de modo a



Figura 6.10: FADEUP [Fonte: Google Earth]

efetuar uma trajetória linear nos dois sentidos. Na nuvem de pontos obtida, ilustrada na figura 6.11, é possível observar a trajetória seguida pelo UAV devido à detecção de pontos na própria estrutura pelo LiDAR.

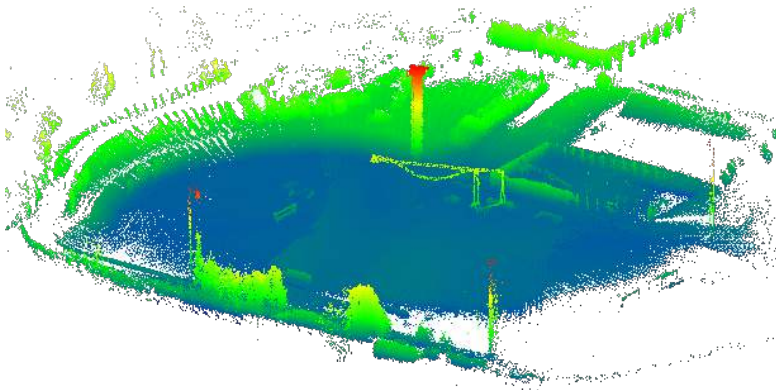


Figura 6.11: Nuvem de pontos do campo de futebol da FADEUP

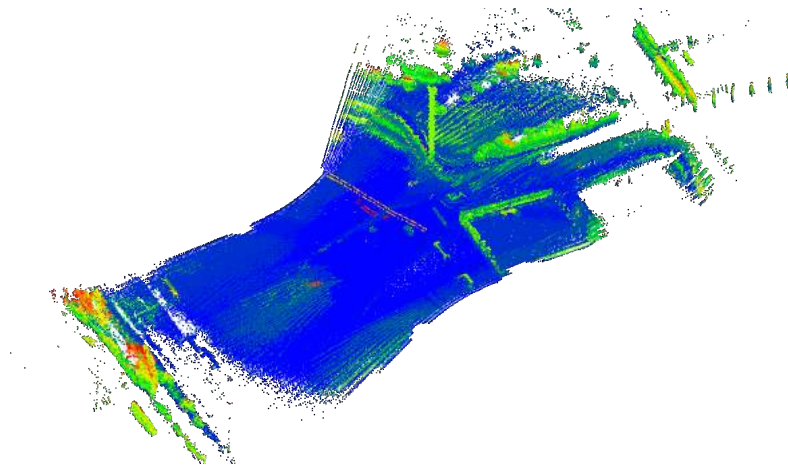


Figura 6.12: Nuvem de pontos do campo de futebol da FADEUP

Capítulo 7

Conclusão e Trabalho Futuro

Nesta dissertação é abordado o desenvolvimento de um algoritmo de calibração dos parâmetros extrínsecos de um LiDAR em relação a um UAV.

Inicialmente foi realizado um estudo com o objetivo de conhecer e entender os métodos e abordagens existentes para a calibração de parâmetros extrínsecos de LiDAR. Essa análise fez com que se atingisse o primeiro objetivo definido para esta dissertação, de maneira que permitiu definir a arquitetura do projeto.

Na abordagem desenvolvida procurou-se tirar proveito de planos presentes na área de varrimento do LiDAR, de forma a obter os parâmetros extrínsecos deste para realizar uma reprojeção da nuvem de pontos com detalhe.

A validação do método desenvolvido consistiu na simulação de um varrimento de 3 planos perpendiculares entre si, respeitando as características do Velodyne, especialmente o FOV vertical e horizontal e a frequência de dados retornados.

Referente ao cenário simulado, a calibração foi alcançada com sucesso referente à orientação do LiDAR, requerendo poucas dezenas de iterações. No entanto, a posição relativa do sensor ao UAV apesar de convergir no mesmo número de iterações, toma valores errados.

Referente a cenários reais, a detecção de planos e a extração dos seus parâmetros apresentou-se bem sucedida. No entanto, não foi possível obter os parâmetros extrínsecos do Velodyne em relação ao STORK.

Com base nos resultados obtidos na calibração em cenário simulado, é possível afirmar que o segundo objetivo foi parcialmente concluído com sucesso, ficando como trabalho futuro simular outros cenários de modo a obter a estimação correta dos parâmetros extrínsecos lineares do LiDAR.

Uma linha de trabalho futuro pretende-se identificar e remover vegetação da nuvem

de pontos visto que esta influencia a extração dos parâmetros dos planos detetados, levando a uma má estimação. Outra linha de trabalho consiste no desenvolvimento de uma melhor abordagem de associação de planos detetados.

Bibliografia

- [1] F. Einaudi, G. K. Schwemmer, B. M. Gentry, and J. B. Abshire, “Lidar Past, Present, and Future in NASA’s Earth and Space Science Programs,” *22nd International Laser Radar Conference (ILRC 2004)*, 2004.
- [2] T. Svoboda, D. Martinec, and T. Pajdla, “A convenient multicamera self-calibration for virtual environments,” *Presence: Teleoperators and Virtual Environments*, vol. 14, no. 4, pp. 407–422, 2005.
- [3] L. Heng, B. Li, and M. Pollefeys, “CamOdoCal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry,” *IEEE International Conference on Intelligent Robots and Systems*, pp. 1793–1800, 2013.
- [4] C. Gao and J. R. Spletzer, “On-line calibration of multiple LIDARs on a mobile vehicle platform,” *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 279–284, 2010.
- [5] M. Sheehan, A. Harrison, and P. Newman, “Self-calibration for a 3D laser,” *International Journal of Robotics Research*, vol. 31, no. 5, pp. 675–687, 2012.
- [6] D. G. Choi, Y. Bok, J. S. Kim, and I. S. Kweon, “Extrinsic calibration of 2D laser sensors,” *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3027–3033, 2014.
- [7] J. Jung, J. Kim, S. Yoon, S. Kim, H. Cho, C. Kim, and J. Heo, “Bore-sight calibration of multiple laser range finders for kinematic 3D laser scanning systems,” *Sensors (Switzerland)*, vol. 15, no. 5, pp. 10292–10314, 2015.
- [8] J. Rowekamper, M. Ruhnke, B. Steder, W. Burgard, and G. D. Tipaldi, “Automatic extrinsic calibration of multiple laser range sensors with little overlap,” *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June, pp. 2072–2077, 2015.

- [9] D. G. Choi, Y. Bok, J. S. Kim, and I. S. Kweon, "Extrinsic Calibration of 2-D Lidars Using Two Orthogonal Planes," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 83–98, 2016.
- [10] L. Meng, F. Sun, and S. S. Ge, "Extrinsic calibration of a camera with dual 2D laser range sensors for a mobile robot," *IEEE International Symposium on Intelligent Control - Proceedings*, pp. 813–817, 2010.
- [11] S. Sim, K. Kwak, J. Kim, and S. H. Joo, "Closed loop-based extrinsic calibration of multi-modal sensors," *International Conference on Control, Automation and Systems*, no. Iccas, pp. 647–651, 2014.
- [12] M. Pereira, D. Silva, V. Santos, and P. Dias, "Self calibration of multiple LIDARs and cameras on autonomous vehicles," *Robotics and Autonomous Systems*, vol. 83, pp. 326–337, 2016.
- [13] M. M. R. Mostafa, "Boresight calibration of integrated inertial/camera systems," *Proceedings of the International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation (KIS 2001), Banff, AB, Canada*, pp. 5–8, 2001.
- [14] Qilong Zhang and R. Pless, "Extrinsic calibration of a camera and laser range finder (improves camera calibration)," *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, pp. 2301–2306, 2004.
- [15] R. Unnikrishnan and M. Hebert, "Fast Extrinsic Calibration of a Laser Rangefinder to a Camera," *Robotics*, vol. 2005, no. July 2005, p. 23, 2005.
- [16] F. Vasconcelos, J. P. Barreto, and U. Nunes, "A minimal solution for the extrinsic calibration of a camera and a laser-rangefinder," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2097–2107, 2012.
- [17] A. Geiger, F. Moosmann, O. Car, and B. Schuster, "Automatic camera and range sensor calibration using a single shot," *2012 IEEE International Conference on Robotics and Automation*, pp. 3936–3943, 2012.
- [18] G. Li, Y. Liu, L. Dong, X. Cai, and D. Zhou, "An algorithm for extrinsic parameters calibration of a camera and a laser range finder using line features," *IEEE International Conference on Intelligent Robots and Systems*, pp. 3854–3859, 2007.

- [19] S. A. Rodriguez F., V. Frémont, and P. Bonnifait, “Extrinsic calibration between a multi-layer lidar and a camera,” *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 214–219, 2008.
- [20] C. Guindel, J. Beltran, D. Martin, and F. Garcia, “Automatic extrinsic calibration for lidar-stereo vehicle sensor setups,” *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, vol. 2018-March, pp. 1–6, 2018.
- [21] S. Wasielewski and O. Strauss, “Calibration of a multi-sensor system laser range-finder/camera,” *IEEE Intelligent Vehicles Symposium*, pp. 472–477, 1995.
- [22] G. Pandey, J. R. McBride, S. Savarese, and R. M. Eustice, “Automatic Extrinsic Calibration of Vision and Lidar by Maximizing Mutual Information,” *Journal of Field Robotics*, vol. 32, no. 5, pp. 696–722, 2015.
- [23] X. Gong, Y. Lin, and J. Liu, “3D LIDAR-camera extrinsic calibration using an arbitrary trihedron,” *Sensors (Switzerland)*, vol. 13, no. 2, pp. 1902–1918, 2013.
- [24] J. Levinson and S. Thrun, “Automatic Online Calibration of Cameras and Lasers,” *Robotics: Science and Systems IX*, 2013.
- [25] Q. Liao, M. Liu, L. Tai, and H. Ye, “Extrinsic Calibration of 3D Range Finder and Camera without Auxiliary Object or Human Intervention,” *CoRR*, vol. abs/1703.0, 2017.
- [26] A. Napier, P. Corke, and P. Newman, “Cross-calibration of push-broom 2D LIDARs and cameras in natural scenes,” in *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3679–3684, IEEE, may 2013.
- [27] R. M. Haralick, C.-n. Lee, K. Ottenberg, and M. Nölle, “Review and Analysis of Solutions of the Three Point Perspective Pose Estimation Problem,” *Ijcv*, vol. 13, no. 3, pp. 331–356, 1994.
- [28] K. Madsen, H. B. Nielsen, and O. Tingleff, “Methods for Non-Linear Least Squares Problems (2nd ed.),” *Informatics and Mathematical Modelling*, p. 60, 2004.
- [29] J. Skaloud and D. Lichti, “Rigorous approach to bore-sight self-calibration in airborne laser scanning,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 61, no. 1, pp. 47–59, 2006.
- [30] J. Skaloud and P. Schaer, “Towards Automated Lidar Boresight Self-Calibration,” *5th International Symposium on Mobile Mapping Technology*, no. May, 2007.

- [31] P. Rieger, N. Studnicka, and M. Pfennigbauer, “Boresight Alignment Method for Mobile Laser Scanning Systems,” *In Proc. of the RSPRS conference*, 2008.
- [32] T. O. Chan, D. D. Lichti, and C. L. Glennie, “Multi-feature based boresight self-calibration of a terrestrial mobile mapping system,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 82, pp. 112–124, 2013.
- [33] R. Le Scouarnec, T. Touzé, J. B. Lacambre, and N. Seube, “A new reliable boresight calibration method for mobile laser scanning applications,” *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, vol. 40, no. 3W1, pp. 67–72, 2014.
- [34] J. Morales, J. L. Martínez, A. Mandow, A. J. Reina, A. Pequeño-Boyer, and A. García-Cerezo, “Boresight calibration of construction misalignments for 3D scanners built with a 2D laser rangefinder rotating on its optical center,” *Sensors (Switzerland)*, vol. 14, no. 11, pp. 20025–20040, 2014.
- [35] D. Li, H. Guo, C. Wang, P. Dong, and Z. Zuo, “Improved bore-sight calibration for airborne light detection and ranging using planar patches,” *Journal of Applied Remote Sensing*, vol. 10, no. 2, p. 024001, 2016.
- [36] J. B.-Y. Tsui, “Earth-Centered, Earth-Fixed Coordinate System,” in *Fundamentals of Global Positioning System Receivers*, pp. 54–72, 2004.
- [37] G. G. Slabaugh, “Computing Euler angles from a rotation matrix,” *denoted as TRTA implementation from httpwww starfireresearch comservices-java3dsamplecodeFlorinE ulers html*, vol. 6, no. 2000, pp. 1–6, 1999.
- [38] T. M. Company, “Pitch, Roll, and Yaw.” [Online]. Disponible: <https://www.touringmachine.com/Articles/aircraft/6/>, 2007. [Acedido: 06-Jun-2018].
- [39] E. W. Weisstein, “Dot product From MathWorld - A Wolfram Web Resource.” [Online]. Disponible: <http://mathworld.wolfram.com/DotProduct.html>. [Acedido: 16-Oct-2018].
- [40] “Inner product *Encyclopedia of Mathematics*.” [Online]. Disponible: http://www.encyclopediaofmath.org/index.php?title=Inner_product&oldid=29549. [Acedido: 16-Oct-2018].

- [41] M. a. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, pp. 381–395, jun 1981.
- [42] D. Borrmann, J. Elseberg, K. Lingemann, and A. Nüchter, “The 3D Hough Transform for plane detection in point clouds: A review and a new accumulator design,” *3D Research*, vol. 2, no. 2, pp. 1–13, 2011.
- [43] u-blox AG, *NEO/LEA-M8 Datasheet*, June 2016. Rev. 3.
- [44] Sensoror AS, *STIM300 Inertia Measurement Unit Datasheet*, May 2013. Rev. 9.
- [45] ComNav, *K501 OEM Board*, 2014.
- [46] R. B. Rusu and S. Cousins, “3D is here: Point Cloud Library (PCL),” in *IEEE International Conference on Robotics and Automation (ICRA)*, (Shanghai, China), May 9-13 2011.
- [47] E. W. Weisstein, “Box-and-whisker plot From MathWorld - A Wolfram Web Resource.” [Online]. Disponível: <http://mathworld.wolfram.com/Box-and-WhiskerPlot.html>. [Acedido: 28-Oct-2018].
- [48] A. F. Winfield, M. Palau Franco, B. Brueggemann, A. Castro, G. Ferri, F. Ferreira, X. Liu, Y. Petillot, J. Roning, F. Schneider, E. Stengler, D. Sosa, and A. Viguria, “euRathlon and ERL Emergency: A Multi-domain Multi-robot Grand Challenge for Search and Rescue Robots,” *Advances in Intelligent Systems and Computing*, vol. 694, pp. 263–271, 2018.