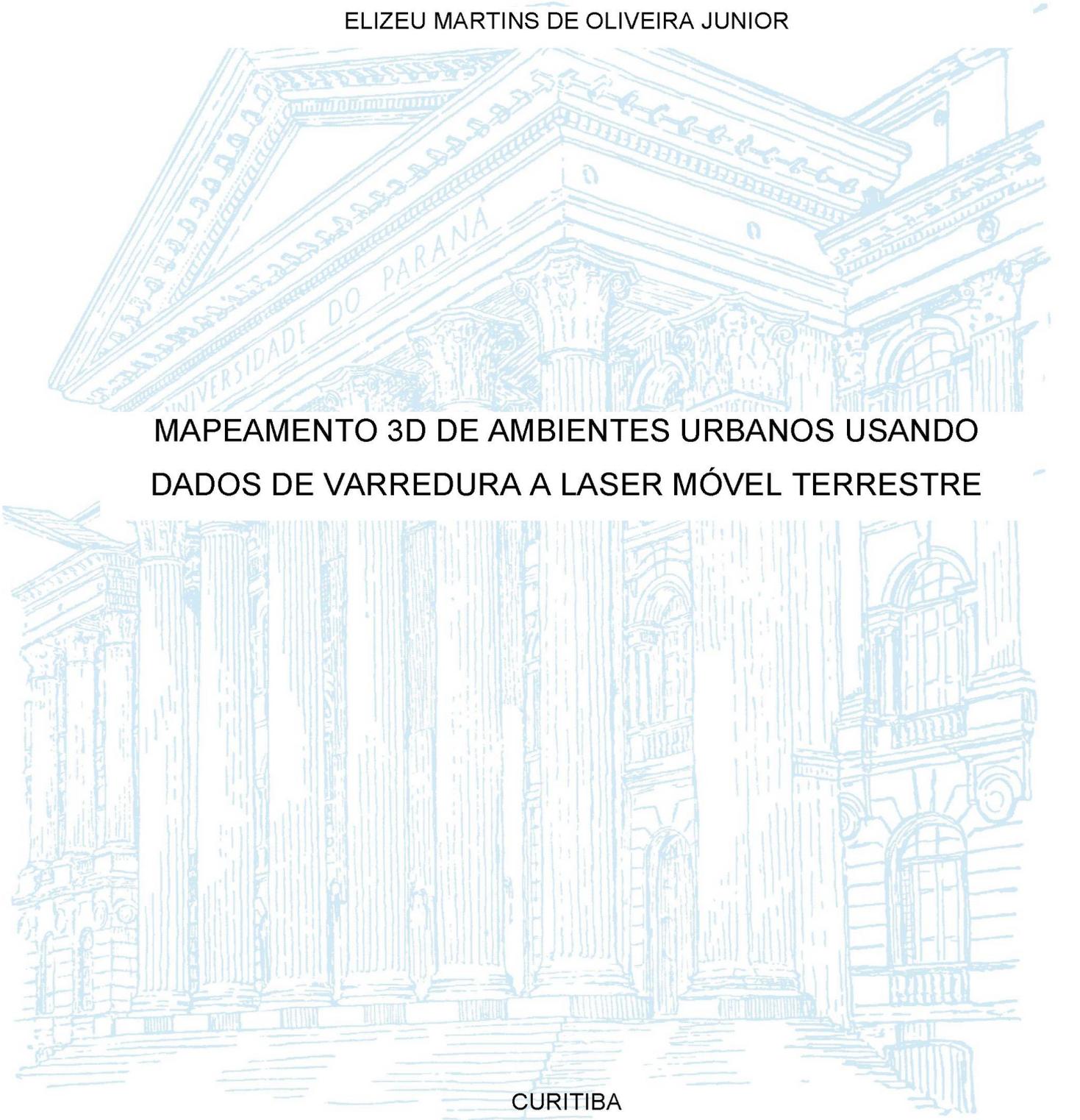


UNIVERSIDADE FEDERAL DO PARANÁ

ELIZEU MARTINS DE OLIVEIRA JUNIOR



MAPEAMENTO 3D DE AMBIENTES URBANOS USANDO
DADOS DE VARREDURA A LASER MÓVEL TERRESTRE

CURITIBA

2020

ELIZEU MARTINS DE OLIVEIRA JUNIOR

**MAPEAMENTO 3D DE AMBIENTES URBANOS USANDO
DADOS DE VARREDURA A LASER MÓVEL TERRESTRE**

Tese apresentada como requisito parcial à obtenção do título de Doutor em Ciências Geodésicas no curso de Pós-Graduação em Ciências Geodésicas, Setor de Ciências da Terra da Universidade Federal do Paraná - UFPR.

Orientador: Prof. Dr. Daniel Rodrigues dos Santos

CURITIBA

2020

Catálogo na Fonte: Sistema de Bibliotecas, UFPR
Biblioteca de Ciência e Tecnologia

O48m Oliveira Junior, Elizeu Martins de
Mapeamento 3D de ambientes urbanos usando dados de varredura a laser móvel terrestre [recurso eletrônico] / Elizeu Martins de Oliveira Junior– Curitiba, 2020.

Tese - Universidade Federal do Paraná, Setor de Ciências da Terra, Programa de Pós-graduação em Ciências Geodésicas.

Orientador: Prof. Dr. Daniel Rodrigues dos Santos

1. Teoria bayesiana de decisão estatística. 2. Sensor Velodyne LiDAR 3D. I. Universidade Federal do Paraná. II. Santos, Daniel Rodrigues dos. III. Título.

CDD: 551.072

Bibliotecária: Roseny Rivelini Morciani CRB-9/1585



MINISTÉRIO DA EDUCAÇÃO
SETOR DE CIÊNCIAS DA TERRA
UNIVERSIDADE FEDERAL DO PARANÁ
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO CIÊNCIAS
GEODÉSICAS - 40001016002P6

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em CIÊNCIAS GEODÉSICAS da Universidade Federal do Paraná foram convocados para realizar a arguição da tese de Doutorado de **ELIZEU MARTINS DE OLIVEIRA JUNIOR** intitulada: **MAPEAMENTO 3D DE AMBIENTES URBANOS USANDO DADOS DE VARREDURA A LASER MÓVEL TERRESTRE**, sob orientação do Prof. Dr. DANIEL RODRIGUES DOS SANTOS, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de doutor está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 31 de Julho de 2020.

Assinatura Eletrônica
11/08/2020 13:09:59.0

DANIEL RODRIGUES DOS SANTOS
Presidente da Banca Examinadora

Assinatura Eletrônica
03/08/2020 12:42:31.0

HENRIQUE CANDIDO DE OLIVEIRA
Avaliador Externo (UNIVERSIDADE ESTADUAL DE CAMPINAS)

Assinatura Eletrônica
03/08/2020 13:25:00.0

MAURICIO GALO
Avaliador Externo (UNIVERSIDADE EST. PAULISTA JÚLIO DE
MESQUITA FILHO/PR. PRUDENT)

Assinatura Eletrônica
03/08/2020 12:59:40.0

IVANDRO KLEIN
Avaliador Interno (INSTITUTO FED. DE EDUC., CIÊNC. E TECNOL.
DE SANTA CATARINA)

Centro Politécnico - Caixa Postal 19001 - CURITIBA - Paraná - Brasil
CEP 81531-980 - Tel: (41) 3361-3153 - E-mail: cpqcg@ufpr.br

Documento assinado eletronicamente de acordo com o disposto na legislação federal Decreto 8539 de 08 de outubro de 2015.
Gerado e autenticado pelo SIGA-UFPR, com a seguinte identificação única: 48502

Para autenticar este documento/assinatura, acesse <https://www.prppg.ufpr.br/siga/visitante/autenticacaoassinaturas.jsp>
e insira o código 48502

DEDICATÓRIA

*Dedico este trabalho à minha esposa
Hellen e a toda minha família*

AGRADECIMENTOS

A minha esposa Hellen pelo amor, compreensão, dedicação, apoio e paciência.

Aos meus pais Jocirlei e Elizeu, toda a minha família e amigos.

Ao meu orientador Professor Dr. Daniel Rodrigues dos Santos, pela amizade, orientação, incentivo, compreensão, conselhos e principalmente paciência.

Aos membros da banca, Dr. Maurício Galo, Dr. Henrique de Oliveira e Dr. Ivandro Klein pelas sugestões ao texto.

Aos colegas do LAPE - Laboratório de Aquisição e Processamento de Dados Espaciais e do PPGCG, Giovana Angélica Ros Miola, Gabriel Dresch, Júlio, Carla, Niarkos, Fabiano Freiman, Kauê, Nicolas e Pedrosa pelo compartilhamento de conhecimentos. Em especial ao Nadisson Luis Pavan que sempre contribuiu com as discussões sobre a matemática.

Também agradeço as famílias Nandes e Lazarin que me acolheram em Curitiba sempre que precisei.

Aos Professores da UNEMAT em especial aos professores Érico Fernando de Oliveira Martins e Rodrigo Bruno Zanin pelo apoio e incentivo.

A todos os professores do Programa de Pós-Graduação em Ciências Geodésicas e funcionários da UFPR.

A coordenação e a secretária Monica Kleuser.

Agradeço à Universidade Federal do Paraná pela oportunidade de aperfeiçoar a minha formação profissional, a CAPES e ao CNPq pelo fornecimento da bolsa de estudos.

“Na Terra não há nada tão grande como o homem;
no homem não há nada tão grande como a mente.”

William Rowan Hamilton

RESUMO

O vasto potencial das representações digitais de objetos por uma coleção de nuvens de pontos 3D têm chamado a atenção de pesquisadores há mais de duas décadas com aplicações em diversas áreas da Ciência e Tecnologia como: Planejamento Urbano, Engenharias, Geociências, Visão Computacional, Robótica, entre outras. Isso fez com que as nuvens de pontos 3D se popularizassem e novas tecnologias surgissem demandando técnicas eficientes para lidar com essas informações para fins de mapeamento tridimensional de ambientes urbanos. Na presente pesquisa é apresentado um método para mapeamento de ambientes urbanos utilizando nuvens de pontos LiDAR (*Light Detection and Ranging*) 3D obtidas por um sensor LiDAR com uma solução estocástica para a correspondência entre planos a partir de uma Matriz de Probabilidades *a priori* e sua posterior atualização por Inferência Bayesiana, além de uma abordagem inédita na determinação dos parâmetros de transformação entre os pares de nuvens de pontos LiDAR 3D com a álgebra de Dual Quaternions. Caso as observações gerem um Espaço Canônico bidimensional, superfícies quase-planares são usadas para criar uma zona de busca por pontos virtuais que, dentro da álgebra de Dual Quaternions, determinam os parâmetros de transformação ótimos e diante a indisponibilidade dessas regiões, um método de aproximação é proposto. A redução média dos valores da REMQ (Raiz do Erro Médio Quadrático) ponto a plano após a aplicação dos algoritmos NDT (*Normal Distribution Transform*), ICP (*Iterative Closest Point*) e o método proposto são de 77%, 80% e 87%, respectivamente. Já o erro de fechamento da componente translacional calculado com o método proposto representa apenas 0,47% e 0,68% comparado aos encontrados pelos algoritmos NDT e ICP, respectivamente.

Palavras-chave: Registro de nuvens de pontos LiDAR 3D; Sensor Velodyne LiDAR 3D; Dual Quaternions; Inferência Bayesiana.

ABSTRACT

The great potential of digital representations of objects through a collection of 3D point clouds has attracted the attention of researchers for over two decades with applications in several areas of Science and Technology such as Urban Planning, Engineering, Geosciences, Computer Vision, Robotics, among others. This made the 3D point clouds to become popular and caused the emergence new technologies requiring efficient techniques to handle such information for tridimensional urban mapping purposes. In this research is proposed a novel method for mapping urban environments using LiDAR (*Light Detection and Ranging*) 3D point clouds obtained by a LiDAR sensor with a stochastic solution for the correspondence between plans in a twofold way: firstly a *priori* Probability Matrix is build relaxation labeling wise and is subsequently updated using Bayesian Inference. Besides that, a new approach is proposed to calculate the transformation parameters between a pair of LiDAR 3D point clouds with Dual Quaternions algebra. If the observations generate a two-dimensional Canonical Space, quasi-planar surfaces are used to create a search zone for virtual points, which determine the optimal transformation parameters within the same Dual Quaternions algebra, in the case, these surfaces are not available an approximation method is used. The average reduction in the REMQ (Root Mean Square Error) point by plane values after the application of the NDT (Normal Distribution Transform), ICP (Iterative Closest Point) and the proposed method are 77%, 80% and 87%, respectively. The translational component of the loop closure error calculated with the proposed method represents only 0.47% and 0.68% compared to those found by the NDT and ICP algorithms, respectively.

Keywords: 3D LiDAR Point Cloud Registration; Velodyne 3D LiDAR sensor, Dual Quaternions, Bayesian Inference.

LISTA DE FIGURAS

FIGURA 1 – REPRESENTAÇÃO DO DESLOCAMENTO DE UM ROBÔ DE MODO CINEMÁTICO TRANSPORTANDO UM SENSOR LIDAR EM UM AMBIENTE URBANO.	21
FIGURA 2 – RESULTADO DO REGISTRO LOCAL ENTRE AS NUVENS DE PONTOS LIDAR 3D DE REFERÊNCIA (VERMELHO) E DE PESQUISA (AZUL) COM O MÉTODO PROPOSTO NESTA PESQUISA.	23
FIGURA 3 – SENSOR VELODYNE HDL-32.....	33
FIGURA 4 – DESCRIÇÃO DO SISTEMA DE REVOLUÇÃO APRESENTADO PELO SENSOR VELODYNE HDL-32 COM DESTAQUE AOS 32 LASERS E SUAS POSIÇÕES E ORIENTAÇÕES.....	34
FIGURA 5 – ILUSTRAÇÃO DO SISTEMA DE COORDENADAS DO SENSOR VELODYNE HDL-32.....	35
FIGURA 6 - TRANSFORMAÇÃO DE CORPO RÍGIDO ENTRE PONTOS COM A ÀLGEBRA DE DUAL QUATERNIONS.	44
FIGURA 7 – POSIÇÕES DO SENSOR.....	54
FIGURA 8 – A) IMAGEM AÉREA DO CAMPUS NORTE DA UNIVERSIDADE DE MICHIGAM; B) IMAGEM DO ROBÔ SEGWAY USADO PARA A OBTENÇÃO DOS DADOS.....	58
FIGURA 9 - FLUXOGRAMA DO MÉTODO PROPOSTO PARA MAPEAMENTO TRIDIMENSIONAL EM AMBIENTES URBANOS.....	60
FIGURA 10 - FLUXOGRAMA DO MÉTODO DE REGISTRO LOCAL PROPOSTO PARA O CASO DE UM ESPAÇO CANÔNICO TRIDIMENSIONAL TER SIDO GERADO PELOS PLANOS OBSERVADOS.....	65
FIGURA 11 - FLUXOGRAMA DO MÉTODO DE REGISTRO LOCAL PROPOSTO PARA O CASO DE UM ESPAÇO CANÔNICO BIDIMENSIONAL TER SIDO GERADO PELOS PLANOS OBSERVADOS.....	67
FIGURA 12 – FLUXOGRAMA DO MÉTODO DE REGISTRO LOCAL PROPOSTO PARA O CASO DE UM ESPAÇO CANÔNICO UNIDIMENSIONAL OU INEXISTENTE SENDO GERADO PELOS PLANOS.....	68

FIGURA 13 - PAR DE NUVENS DE PONTOS LIDAR 3D COM DESTAQUE AOS VETORES NORMAIS DOS PLANOS DETECTADOS.....	70
FIGURA 14 - PAR DE NUVENS DE PONTOS LIDAR 3D E OS CENTROIDES DOS PLANOS EXTRAÍDOS, COM DESTAQUE AOS VETORES DE DISTÂNCIA ENTRE C_1 E OS CENTROIDES C_j DOS PLANOS NA NUVEM DE PESQUISA.....	72
FIGURA 15 – PLANOS DETECTADOS PELO ALGORITMO DE CRESCIMENTO DE REGIÃO (VERDE E PRETO) E <i>CLUSTERS</i> USADOS NO PROCESSO DE REGISTRO LOCAL (AZUL E VERMELHO).....	87
FIGURA 16 – <i>CLUSTERS</i> CORRESPONDENTES NA NUVEM DE PONTOS LIDAR 3D DE REFERÊNCIA (VERMELHO) E PESQUISA (AZUL).....	90
FIGURA 17 - FLUXOGRAMA DO MÉTODO DE APROXIMAÇÃO <i>SLIDING</i>	91
FIGURA 18 - PAR DE PLANOS ANTES (A) E APÓS (B) A APLICAÇÃO DOS PARÂMETROS DE TRANSFORMAÇÃO PRELIMINARES À NUVEM DE PONTOS LIDAR 3D DE PESQUISA (VERMELHO) NO PROCESSO DE <i>SLIDING</i> . NUVEM DE PONTOS LIDAR 3D DE REFERÊNCIA EM AZUL.....	92
FIGURA 19 – PLANOS CHAVE ANTES (A) E APÓS (B) A APLICAÇÃO DE TODOS OS COMPONENTES TRANSLACIONAIS NA APROXIMAÇÃO <i>SLIDING</i>	93
FIGURA 20 – FLUXOGRAMA DO MÉTODO DE REFINAMENTO GLOBAL APLICADO NESTA PESQUISA COM BASE NO ALGORITMO LUM.....	95
FIGURA 21 – VISTA AÉREA DA ÁREA DE ESTUDO USADA NESTA PESQUISA.....	97
FIGURA 22 – AVALIAÇÃO DA DESEMPENHO DAS CORRESPONDÊNCIAS ENTRE PLANOS PRESENTES EM ATÉ 8 NUVENS À FRENTE.....	100
FIGURA 23 – AVALIAÇÃO DO USO DA INFERÊNCIA BAYESIANA NO PROCESSO DE CORRESPONDÊNCIA ENTRE PLANOS.....	100
FIGURA 24 – VALORES DA REMQ PONTO A PLANO ANTES E APÓS O REGISTRO LOCAL PARA O 1º TRECHO.....	102
FIGURA 25 - VALORES DA REMQ PONTO A PLANO ANTES E APÓS O REGISTRO LOCAL PARA O 2º TRECHO.....	102
FIGURA 26 – FREQUÊNCIAS DOS MÉTODOS ESCOLHIDOS PARA DETERMINAR OS PARÂMETROS DE TRANSFORMAÇÃO A SEREM APLICADOS À NUVEM DE PONTOS LIDAR 3D DE PESQUISA.....	104

FIGURA 27 – VALORES DA REMQ PONTO A PLANO APÓS O REGISTRO GROSSEIRO.....	104
FIGURA 28 - VALORES DA REMQ PONTO A PLANO APÓS O MÉTODO DE REGISTRO GROSSEIRO-PARA-FINO.....	105
FIGURA 29 - VALORES DA REMQ PONTO A PLANO OBTIDOS COM O MÉTODO DE REGISTRO LOCAL PROPOSTO NESTA PESQUISA (PRETO), COM OS DADOS DE ODOMETRIA REFINADOS COM O ALGORITMO ICP (VERMELHO) E APENAS COM O ALGORITMO ICP (AZUL), PARA O 1º TRECHO.....	106
FIGURA 30 - VALORES DA REMQ PONTO A PLANO OBTIDOS COM O MÉTODO DE REGISTRO LOCAL PROPOSTO NESTA PESQUISA (PRETO), COM OS DADOS DE ODOMETRIA REFINADOS COM O ALGORITMO ICP (VERMELHO) E APENAS COM O ALGORITMO ICP (AZUL), PARA O 2º TRECHO.....	107
FIGURA 31 - VALORES DA REMQ PONTO A PLANO OBTIDOS COM O MÉTODO DE REGISTRO LOCAL PROPOSTO NESTA PESQUISA (PRETO), COM OS DADOS DE ODOMETRIA REFINADOS COM O ALGORITMO NDT (VERMELHO) E APENAS COM O ALGORITMO NDT (AZUL), PARA O 1º TRECHO.....	107
FIGURA 32 - VALORES DA REMQ PONTO A PLANO OBTIDOS COM O MÉTODO DE REGISTRO LOCAL PROPOSTO NESTA PESQUISA (PRETO), COM OS DADOS DE ODOMETRIA REFINADOS COM O ALGORITMO NDT (VERMELHO) E APENAS COM O ALGORITMO NDT (AZUL), PARA O 2º TRECHO.....	108
FIGURA 33 – IMAGEM EM DUAS PERSPECTIVAS DE UM PAR DE NUVENS DE PONTOS LIDAR 3D APÓS A APLICAÇÃO DO MÉTODO DE REGISTRO GROSSEIRO PROPOSTO NESTA PESQUISA.COM UM ESPAÇO CANÔNICO BIDIMENSIONAL GERADO.....	109
FIGURA 34 – IMAGEM VISTA DE CIMA DE UM PAR DE NUVENS DE PONTOS LIDAR 3D APÓS A APLICAÇÕES DO MÉTODO DE REGISTRO LOCAL PROPOSTO NESTA PESQUISA.....	110
FIGURA 35 - VALORES DA REMQ PONTO A PLANO APÓS O REGISTRO UTILIZANDO DADOS DE ODOMETRIA E OS ALGORITMOS ICP A) E NDT B) COMO REGISTRO FINO.....	111

FIGURA 36 – MEDIÇÕES DE TEMPO PARA OS REGISTRO LOCAIS COM O MÉTODO PROPOSTO GROSSEIRO-PARA-FINO (PRETO), COM OS DADOS DE ODOMETRIA ASSOCIADOS AO ALGORITMO ICP (VERMELHO) E COM O ALGORITMO ICP APENAS (AZUL) PARA O 1º TRECHO.....	112
FIGURA 37 - MEDIÇÕES DE TEMPO PARA OS REGISTRO LOCAIS COM O MÉTODO PROPOSTO GROSSEIRO-PARA-FINO (PRETO), COM OS DADOS DE ODOMETRIA ASSOCIADOS AO ALGORITMO ICP (VERMELHO) E COM O ALGORITMO ICP APENAS (AZUL) PARA O 2º TRECHO.....	113
FIGURA 38 - MEDIÇÕES DE TEMPO PARA OS REGISTRO LOCAIS COM O MÉTODO PROPOSTO GROSSEIRO-PARA-FINO (PRETO), COM OS DADOS DE ODOMETRIA ASSOCIADOS AO ALGORITMO NDT (VERMELHO) E COM O ALGORITMO NDT APENAS (AZUL) PARA O 1º TRECHO. EXECUÇÃO DO MÉTODO <i>SLIDING</i> (DESTAQUE EM AMARELO).	113
FIGURA 39 - MEDIÇÕES DE TEMPO PARA OS REGISTRO LOCAIS COM O MÉTODO PROPOSTO GROSSEIRO-PARA-FINO (PRETO), COM OS DADOS DE ODOMETRIA ASSOCIADOS AO ALGORITMO NDT (VERMELHO) E COM O ALGORITMO NDT APENAS (AZUL) PARA O 2º TRECHO.....	114
FIGURA 40 - TRAJETÓRIA OBTIDA COM OS DADOS DE ODOMETRIA ASSOCIADOS AO ALGORITMO ICP B) E COM O ALGORITMO ICP APENAS A).	115
FIGURA 41 - TRAJETÓRIA OBTIDA COM OS DADOS DE ODOMETRIA ASSOCIADOS AO MÉTODO NDT B) E COM O MÉTODO NDT APENAS A).	116
FIGURA 42 – TRAJETÓRIA OBTIDA ANTES E APÓS A APLICAÇÃO DOS PARÂMETROS DE REFINAMENTO GLOBAL EM DIFERENTES PERSPECTIVAS, COM DESTAQUE AO ERRO DE FECHAMENTO EM B).	116
FIGURA 43 – DESTAQUE DA FIGURA 42 COM UM ZOOM NO ERRO DE FECHAMENTO.	117
FIGURA 44 - NUVEM DE PONTOS LIDAR 3D GLOBAL OBTIDA COM O ALGORITMO ICP CONSIDERANDO OS DADOS DE ODOMETRIA COMO PARÂMETROS DE TRANSFORMAÇÃO INICIAIS.....	117

FIGURA 45 - NUVEM DE PONTOS LIDAR 3D GLOBAL OBTIDA COM O ALGORITMO NDT COM OS DADOS DE ODOMETRIA COMO PARÂMETROS DE TRANSFORMAÇÃO INICIAIS.	118
FIGURA 46 – NUVEM DE PONTOS LIDAR 3D GLOBAL OBTIDA COM O MÉTODO PROPOSTO SOB UMA PERSPECTIVA 2D. COM DESTAQUE À REGIÃO DE FECHAMENTO DO <i>LOOP</i> (VERMELHO).	119
FIGURA 47 - NUVEM DE PONTOS LIDAR 3D GLOBAL OBTIDA COM O MÉTODO PROPOSTO SOB A PERSPECTIVA 3D.	119
FIGURA 48 – REGISTRO LOCAL ENTRE A PRIMEIRA (VERMELHO) E A ÚLTIMA (PRETO) NUVEM DE PONTOS LIDAR 3D ILUSTRANDO O ERRO DE FECHAMENTO APÓS O REFINAMENTO GLOBAL MOSTRADO NA FIGURA 43 EM UMA PERSPECTIVA 3D.	120
FIGURA 49 - REGISTRO LOCAL ENTRE A PRIMEIRA (VERMELHO) E A ÚLTIMA (PRETO) NUVEM DE PONTOS LIDAR 3D ILUSTRANDO O ERRO DE FECHAMENTO DO <i>LOOP</i> APÓS O REFINAMENTO GLOBAL MOSTRADO NA FIGURA 43 EM UMA PERSPECTIVA 2D.	120

LISTA DE QUADROS

QUADRO 1 - RESUMO DOS TRABALHOS INVESTIGADOS NO ESTADO DA ARTE COM A ABORDAGEM PONTOS DE INTERESSE.....	27
QUADRO 2 - RESUMO DOS TRABALHOS INVESTIGADOS NO ESTADO DA ARTE COM A ABORDAGEM DE SUPERFÍCIES IMPLÍCITAS.....	28
QUADRO 3 - RESUMO DOS TRABALHOS INVESTIGADOS NO ESTADO DA ARTE COM A ABORDAGEM FEIÇÕES.....	31

LISTA DE TABELAS

TABELA 1 – ÂNGULOS VERTICAIS α SEGUNDO A POSIÇÃO DE CADA LASER NO PAINEL.	36
TABELA 2 - PARÂMETROS UTILIZADOS PELOS MÉTODOS CITADOS NO CAPÍTULO 3.	98
TABELA 3 – MEDIDAS DE TENDÊNCIA CENTRAL E VARIABILIDADE DOS RESULTADOS DA REMQ PONTO A PLANO ANTES E APÓS OS CENÁRIOS CONSIDERADOS NESTA PESQUISA.	111
TABELA 4 – MEDIÇÕES DE TEMPO DOS MÉTODOS DE REGISTRO GROSSEIRO E GROSSEIRO-PARA-FINO E OS ALGORITMOS USADOS COMO COMPARATIVO, ICP E NDT.....	112
TABELA 5 – ERROS ANGULARES E TRANSLACIONAIS DE FECHAMENTO ANTES E APÓS O REFINAMENTO GLOBAL.....	121

LISTA DE SIGLAS

3D	- Tridimensional
ALS	- <i>Airborne LASER Scanning</i>
BSD	- <i>Berkeley Software Distribution</i>
DoG	- <i>Difference of Gaussian</i>
FALS	- <i>Fast Approximate Least Squares</i>
EKF	- <i>Extended Kalman Filter</i>
FOG	- <i>Fiber Optic Gyro</i>
FoV	- <i>Field of View</i>
GPL	- <i>General Public License</i>
GNSS	- <i>Global Navigation Satellite System</i>
IC3PO	- <i>Iterative Closet Point Plus Plane Optimization</i>
ICP	- <i>Iterative Closest Point</i>
IMLS	- <i>Implicit Moving Least Square</i>
IMU	- <i>Inertial Measurement Unit</i>
K-4PCS	- <i>Keypoint-based 4-Points Congruent Sets</i>
LASER	- <i>Light Amplification by Stimulated Emission of Radiation</i>
LiDAR	- <i>Light Detection and Ranging</i>
MC	- <i>Matriz de Covariâncias</i>
MDT	- <i>Modelos Digitais do Terreno</i>
MEMS	- <i>Micro-Electro MEchanical Systems</i>
MLS	- <i>Mobile Laser Scanner</i>
MMF	- <i>Modelo Matemático Funcional</i>
MMQ	- <i>Método dos Mínimos Quadrados</i>
NDT	- <i>Normal Distribution Transform</i>
RANSAC	- <i>RANdon SAmple Consesus</i>
RGB-D	- <i>Red, Green, Blue e Depth</i>
SfM	- <i>Structure from Motion methods</i>
SVD	- <i>Singular Valor Decomposition</i>
TLS	- <i>Terrestrial LASER System</i>
ToF	- <i>Time of Flight</i>

SUMÁRIO

1. INTRODUÇÃO	19
1.1. CONSIDERAÇÕES INICIAIS	19
1.2. CONTEXTUALIZAÇÃO DO PROBLEMA E MOTIVAÇÃO	21
1.2. OBJETIVOS.....	24
1.3. ESTADO DA ARTE	25
1.4. ESTRUTURA DO TRABALHO	32
2. FUNDAMENTAÇÃO TEÓRICA	33
2.1. SENSOR VELODYNE	33
2.2. ALGORITMO RANSAC	36
2.3. ALGORITMO CRESCIMENTO DE REGIÃO E DETERMINAÇÃO DOS PARÂMETROS DOS PLANOS	38
2.4. ÁLGEBRA DE DUAL QUATERNIONS.....	40
2.4.1. Números Duais.....	40
2.4.2. Dual Quaternions.....	41
2.4.3. Representação da Transformação de Corpo Rígido por Dual Quaternions	43
2.4.4. Cálculo dos Parâmetros de Transformação de Pontos com Dual Quaternions	45
2.5. ALGORITMO ICP	52
2.6. MÉTODO LUM	53
3. MATERIAIS E MÉTODOS	57
3.1. MATERIAIS	57
3.1.1. Recursos de Hardware	59
3.1.2. Recursos de Software e Bibliotecas Computacionais <i>Open-Source</i>	59
3.2. MÉTODO	60
3.2.1. Registro Local.....	61
3.2.1.1. Voxelização e Remoção de Pontos Espúrios	61
3.2.1.2. Segmentação e Extração dos Planos.....	62
3.2.1.3. Estabelecimento de Correspondências entre os Planos e o Cálculo dos Parâmetros de Transformação	64
3.2.1.3.1. Método de Correspondência dos Planos.....	69

3.2.1.3.1.1. Função de Probabilidade - Inclinação.....	70
3.2.1.3.1.2. Função de Probabilidade - Distância entre os Centroides.....	71
3.2.1.3.1.3. Função de Probabilidade – Cardinalidade dos Planos	73
3.2.1.3.1.4. Matriz de Probabilidade <i>a priori</i> P.....	74
3.2.1.3.1.5. Inferência Bayesiana na Atualização da Matriz de Probabilidades ...	75
3.2.1.3.2. Determinação do Espaço Canônico Gerado pelos Planos.....	79
3.2.1.3.3. Estimativa dos Parâmetros de Transformação com Planos e a Álgebra de Dual Quaternions.....	80
3.2.1.3.4. Determinação dos Parâmetros de Transformação com um Espaço Canônico Bidimensional	86
3.2.1.3.5. Método de Aproximação <i>Sliding</i>	90
3.2.1.3.6. Registro Fino	93
3.2.2. Refinamento Global.....	94
4. EXPERIMENTOS E ANÁLISE DOS RESULTADOS.....	97
5. CONCLUSÕES E RECOMENDAÇÕES	122
6. REFERÊNCIAS	125
APÊNDICE	131

1. INTRODUÇÃO

1.1. CONSIDERAÇÕES INICIAIS

De acordo com o site *Grand View Research*¹ especializado em marketing empresarial e tendências de mercado, em um relatório publicado em março de 2020, a indústria mundial de mapeamento digital possui um valor de mercado em torno de 6,2 bilhões de dólares com uma previsão de crescimento anual de 13% pelos próximos sete anos. Essa expansão é resultado principalmente das inovações tecnológicas dos sensores utilizados nos projetos de mapeamento e do crescimento da demanda por informações tridimensionais em diversas áreas como: Arqueologia, Engenharia, Geociências, Planejamento Urbano, Robótica, entre outras.

No mapeamento 3D (tridimensional) as nuvens de pontos são obtidas por sensores de imageamento passivo, por exemplo, por imagens digitais (*SfM - Structure from Motion methods*), ou ativos, por sensores como: câmeras de distância 3D, câmeras de profundidade RGB-D (*Red, Green, Blue e Depth*), e, sensores de varredura a LASER (*Light Amplification by Stimulated Emission of Radiation*) TLS (*Terrestrial Laser System*) e LiDAR; com a finalidade de representar o ambiente digitalmente. Por definição, nuvens de pontos 3D são conjuntos de pontos referenciados a um determinado sistema de coordenadas tridimensionais (X, Y, Z) que representam digitalmente a superfície de um objeto qualquer (WEHR e LOHR, 1999). Uma característica dos sensores LiDAR é o fornecimento da intensidade do pulso LASER, que varia de acordo com a composição da superfície do objeto, observando a força do retorno do pulso (SHAN e TOTH, 2010).

De acordo com Shan e Toth (2008), embora o LASER tenha sido inventado no início dos anos 60, foi somente a partir da década de 90 com o avanço da tecnologia e do poder computacional que o georreferenciamento direto se tornou viável para fins comerciais dando mais visibilidade para os sensores LiDAR. Nos últimos anos, esses sensores se tornaram mais acessíveis e atrativos, principalmente em razão do avanço da tecnologia MEMS (*Micro-Electro MEchanical Systems*) e do aumento da capacidade

¹ Disponível em <<https://www.grandviewresearch.com/industry-analysis/digital-map-market>>. Report ID: GVR-2-68038-021-7. Acesso em 05 de Maio de 2020.

computacional de processamento das leituras dos sensores LASER, da acurácia das informações geoespaciais dos objetos e da resolução digital. Assim, aplicações podem ser encontradas na modelagem tridimensional de pessoas, animais, plantas, objetos, criação de Modelos Digitais do Terreno (MDT), Inventário Digital, investigação forense, impressão 3D, direção autônoma de veículos, mapeamento urbano, estudos ambientais, gerenciamento de desastres, análise de riscos e etc.

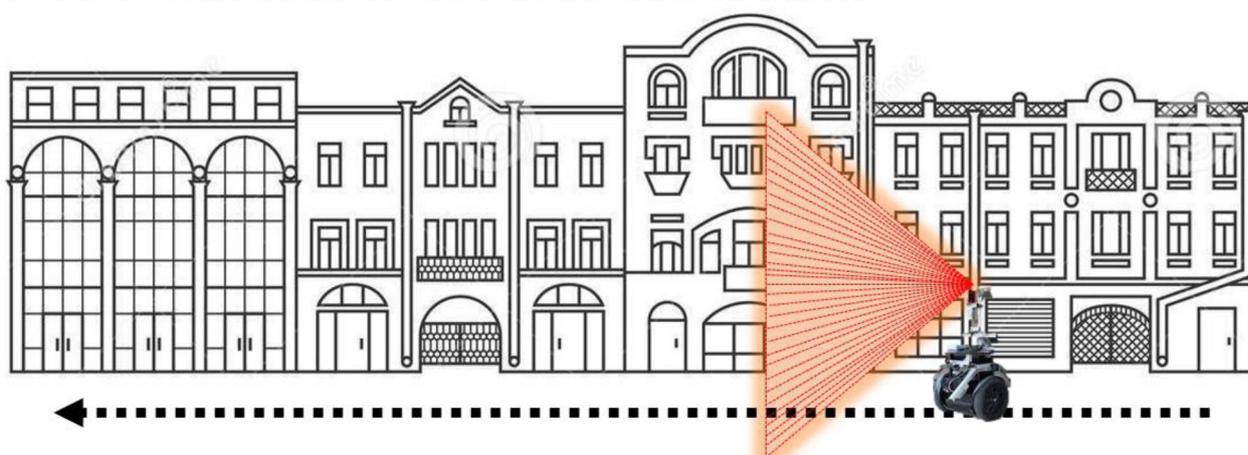
De todos os sensores LiDAR disponíveis no mercado o Velodyne tem se destacado pela precisão e custo benefício (a partir de 4 mil dólares na versão VLP-16). Com um FoV (*Field of View*) de 360° na horizontal, uma precisão na ordem de 2 centímetros e uma alta taxa de obtenção dos dados (até 20 Hertz), o sensor foi projetado para auxiliar na direção autônoma de veículos dando “sentido de percepção” para que naveguem com segurança em ambientes complexos e desconhecidos. Independente do sensor utilizado, lidar com alterações das condições ambientais e em ambientes restritos a POS (*Positioning and Orientation System*), formado pela integração de sensores de orientação IMU (*Inertial Measurement Unit*) e posicionamento GNSS (*Global Navigation Satellite System*), ainda é um desafio em projetos de mapeamento urbano (GRANT, 2013 apud LICHTI *et al*, 2008).

Dentre as vantagens deste tipo de sensor para fins de mapeamento destaca-se seu alcance (até 100 metros no modelo HDL-32), além disso, este sensor pode ser usado em mapeamentos de grande escala e, portanto, o desenvolvimento de uma solução SLAM (*Simultaneous Localization And Mapping*) pode ser aplicada com base em medições LiDAR 3D. Uma das abordagens utilizadas para a solução de SLAM emprega o registro par a par de nuvens de pontos LiDAR 3D para estabelecer as relações matemáticas entre os sistemas de coordenadas independentes referentes as posições do sensor na determinação da trajetória executada. Segundo Akca *et al*, (2010), o registro de nuvens de pontos 3D é essencial e influencia diretamente na qualidade final de um mapeamento tridimensional.

1.2. CONTEXTUALIZAÇÃO DO PROBLEMA E MOTIVAÇÃO

Considerando que em uma única revolução do sensor ativo LiDAR possam ser calculados milhares de pontos, geralmente num levantamento de ambientes urbanos o sensor deve ser reposicionado devido ao limite do seu FoV ou pela cobertura insuficiente dos objetos presentes no espaço, o que Pfeifer e Böhn (2008) chamam de *multi-station configuration*, escopo da área de Fotogrametria Terrestre. O levantamento pode ocorrer de três formas distintas: em um *Scanning Cinemático* (ver Figura 1) onde o sensor LiDAR realiza as leituras em uma plataforma em movimento; em um modo *stop and go*, onde a plataforma estaciona para realizar as leituras; e, pelo modo estático com o reposicionamento do equipamento por um operador.

FIGURA 1 – REPRESENTAÇÃO DO DESLOCAMENTO DE UM ROBÔ DE MODO CINEMÁTICO TRANSPORTANDO UM SENSOR LIDAR EM UM AMBIENTE URBANO.



FONTE: O autor² (2020)

Se um POS não for rigorosamente integrado ao sensor LiDAR, as nuvens de pontos LiDAR 3D obtidas continuarão referenciadas ao sistema de coordenadas do sensor, necessitando assim dos parâmetros de transformação (rotação e translação) entre cada par de nuvens para referenciá-las a origem do sistema de coordenadas da primeira nuvem de pontos LiDAR 3D obtida. Segundo o manual de Fotogrametria (SLAMA *et al*, 1980), o processo de estimativa dos parâmetros de transformação entre dois sistemas de coordenadas a partir de observações conjugadas é similar a orientação

² Créditos a DREAMSTIME (2020) e MICHIGAN (2012).

relativa na Fotogrametria. De acordo com Pfeifer e Böhn (2008), o processo de referenciamento de um conjunto de nuvens de pontos LiDAR 3D a um mesmo sistema de coordenadas de referência denomina-se registro. O registro de nuvens de pontos LiDAR 3D pode ser utilizado até em sistemas que usam informações externas, por exemplo, quando se deseja integrar medidas georreferenciadas de um sensor ALS (*Airborne LASER Scanning*) com nuvens de pontos 3D obtidas por um MLS (*Mobile Laser Scanner*) ou TLS para se ter uma visão das fachadas dos edifícios, o que não é possível com o levantamento aéreo.

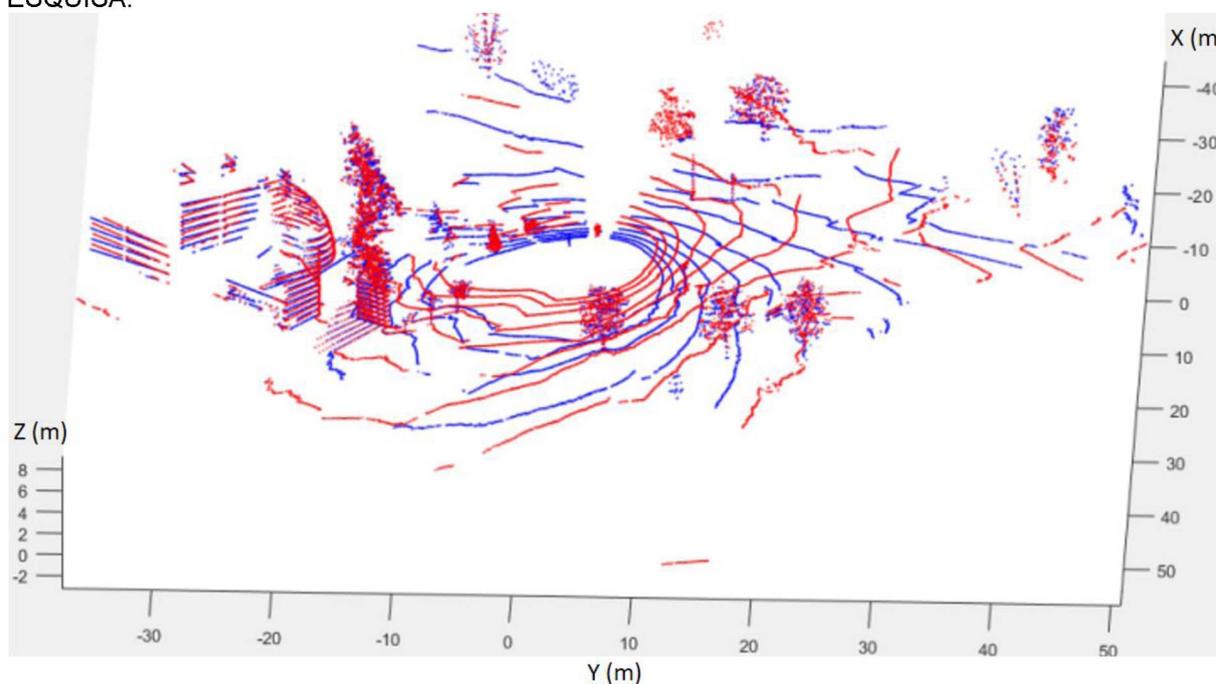
Segundo Gressin *et al*, (2013), a abordagem mais eficiente para a realização do registro de nuvens de pontos LiDAR 3D por meio do estabelecimento de correspondências entre primitivas (pontos, retas, planos e etc.) nas nuvens de pontos LiDAR 3D, principalmente se forem esparsas, volumosas e com a presença de pontos espúrios. Nessas abordagens, o registro de nuvens de pontos LiDAR 3D pode ser dividido em duas etapas: grosseiro e fino. O papel do registro grosseiro (geralmente analítico) é fornecer um bom chute inicial para os parâmetros de transformação entre os pares de nuvens de pontos LiDAR 3D, enquanto que o do registro fino (usualmente iterativo) é realizar uma aproximação final entre as nuvens de pontos 3D. Na literatura, quando se aproveita da sinergia entre eles, têm-se o que chamam de registro grosseiro-para-fino (DOS SANTOS *et al*, 2016). Neste trabalho é apresentado um método de mapeamento urbano utilizando o sensor LiDAR com o registro de nuvens de pontos LiDAR 3D no modo grosseiro-para-fino, com as principais contribuições no campo do registro grosseiro. Inclusive, conforme pode ser visto no Capítulo IV, em muitos registros realizados, o registro fino nem chegou a ser empregado, diminuindo assim o custo computacional (tempo empregado no processamento).

Geralmente em projetos de mapeamento 3D em ambientes urbanos, o sinal recebido por uma antena GNSS sofre interferências devido a obstruções dentro ou próximo de grandes estruturas e objetos em movimento como pedestres, veículos e etc. Por isso, usualmente apenas a IMU é utilizada como fonte de orientação externa, entretanto, este sensor acumula erros exponencialmente conforme opera, o que impede seu uso de forma desassistida. Por isso, faz-se necessário um método robusto para

fornecer uma estimativa inicial dos parâmetros de transformação para amenizar os erros do processo de alinhamento entre os pares de nuvens de pontos 3D.

Um fator dificultador para o processo de registro de nuvens de pontos 3D obtidas por sensores LiDAR é a impossibilidade de se determinar uma correspondência exata entre pontos presentes nas regiões de sobreposição das nuvens de pontos LiDAR 3D, com evidência aos pontos pertencentes ao terreno, como pode ser visto na Figura 2. Para contornar esse problema, as feições planares, abundantes em ambientes antrópicos, podem ser usadas para estabelecer as relações matemáticas entre os dois sistemas de coordenadas.

FIGURA 2 – RESULTADO DO REGISTRO LOCAL ENTRE AS NUVENS DE PONTOS LIDAR 3D DE REFERÊNCIA (VERMELHO) E DE PESQUISA (AZUL) COM O MÉTODO PROPOSTO NESTA PESQUISA.



FONTE: O autor (2020)

Um grande problema em relação ao uso de feições para o registro de nuvens de pontos LiDAR 3D, além do custo computacional inerente ao processo de segmentação e extração que deve ser o mais otimizado possível, é a disponibilidade dessas feições no ambiente a ser mapeado. Dessa forma, primeiramente deve ser verificado o Espaço Canônico gerado por essas observações, ou seja, analisar qual a contribuição de cada feição para a solução dos parâmetros de transformação. Além disso, um método de

correspondência automático e eficiente é indispensável para a realização dos registros locais para este tipo de abordagem baseada em feições. Portanto, todos esses fatores devem ser considerados na formulação de um método de registro local de nuvens de pontos LiDAR 3D visando a realização de mapeamento tridimensional otimizado de ambientes urbanos.

Nesta pesquisa o método de correspondência entre planos proposto é baseado nos conceitos de rotulação por relaxação probabilística (HUMMEL E ZUCKER, 1983) aplicada à correspondência de planos obtidos de nuvens de pontos LiDAR 3D de forma inédita. Basicamente, relações geométricas globais entre os planos candidatos a homólogos são caracterizadas como probabilidades na forma de uma matriz de similaridade, e em seguida, essa matriz é atualizada utilizando Inferência Bayesiana a partir de relações geométricas locais e uma configuração inicial.

Para a solução dos parâmetros de transformação do registro grosseiro, um método analítico é proposto em uma abordagem híbrida com planos e pontos. Caso os vetores normais dos planos divirjam para todas as direções do Espaço Canônico tridimensional apenas os planos são utilizados de forma inédita com a álgebra de Dual Quaternions, do contrário, pontos são utilizados para garantir o grau de liberdade que falta ao sistema de equações. Para não aumentar o tempo de processamento, os métodos propostos para o estabelecimento e a correspondência dos pontos garantem sua inclusão diretamente na solução dos parâmetros com a álgebra de Dual Quaternions sem a necessidade de iteração. A seguir serão descritos os objetivos aos quais se deseja alcançar na presente pesquisa.

1.2. OBJETIVOS

O objetivo geral deste trabalho é o desenvolvimento de um método para o mapeamento tridimensional em ambientes urbanos utilizando planos automaticamente correspondidos por um processo estocástico e a álgebra de Dual Quaternions na estimativa dos parâmetros de transformação do registro grosseiro. Para alcançar o objetivo deste trabalho, a seguir são elencados os objetivos específicos:

- 1) Extração dos planos nas nuvens de pontos LiDAR 3D obtidas com o sensor LiDAR;
- 2) Estabelecer as correspondências dos planos homólogos entre os pares de nuvens de pontos LiDAR 3D e determinar o Espaço Canônico gerado por eles;
- 3) Estimar os parâmetros de transformação entre os pares de nuvens de pontos LiDAR 3D utilizando os planos e a álgebra de Dual Quaternions para os Espaços Canônicos tridimensionais juntamente com pontos para os bidimensionais;
- 4) Refinar local e globalmente os parâmetros de transformação do registro local utilizando o algoritmo ICP e o método LUM, respectivamente;
- 5) Avaliar quantitativa e qualitativamente os resultados obtidos com o método proposto em um comparativo com outros dois algoritmos: ICP (BELS e MCKAY, 1992) e NDT (BIBER e STRASSER, 2003) utilizando ou não os dados de odometria como parâmetros de transformação iniciais;

1.3. ESTADO DA ARTE

Como dito anteriormente, com o avanço da tecnologia os sensores LiDAR se tornaram cada vez mais acurados e acessíveis para fins de mapeamento tridimensional em ambientes urbanos. Conseqüentemente também surge a necessidade de técnicas robustas para lidar com a abundância de observações geradas por sensores LiDAR em projetos de mapeamento, como por exemplo, soluções de SLAM *offline*. Segundo Grisetti *et al*, (2010), o SLAM é um processo pela qual uma plataforma/robô constrói um mapa de um ambiente desconhecido a qual se encontra, e, simultaneamente se localiza dentro do mapa atualizando-o, sem a ajuda de sistemas de posicionamento local ou global. Na literatura existem várias propostas diferentes para solução desse problema, como por exemplo, as baseadas em Filtros de Partículas (BLANCO *et al*, 2010), Filtros Extendidos de Kalman (HUANG *et al*, 2008), Intersecção de Covariâncias (SIMON e UHLMANN, 2007) e Grafos (LU e MILLIOS, 1997; GRISSETTI *et al*, 2007; KÜMMERLE *et al*, 2011).

Um processo recorrente às propostas por grafos é a realização do registro entre pares de nuvens de pontos LiDAR 3D, que pode ser dividido em 3 diferentes abordagens (GRESSIN *et al*, 2013): Baseadas em feições, de forma analítica (HORN, 1987; ARUN *et*

al, 1987) e iterativa (BESL e MCKAY, 1992; CHEN e MEDIONI, 1992, BLAIS e LEVINE, 1995; MASUDA *et al*, 1996; RUSISKIEWICZ e LEVOY, 2001; NUCHTER *et al*, 2007; BAE e LITCHI, 2008; SEGAL *et al*, 2009; GRANT *et al*, 2012; GRANT *et al*, 2019); baseadas em superfícies implícitas, onde os pontos são representados por modelos de superfície e as nuvens de pontos 3D são adicionadas diretamente no mapa global (GRUEN e AKCA, 2005; DESCHAUD, 2018); e, baseadas em pontos de interesse, dispensando os processos de segmentação ou interpolação dos pontos e focando na determinação de pontos de interesse baseado em relações geométricas dessas entidades na própria nuvem (GRESSIN *et al*, 2013; THEILER *et al*, 2014; VLAMINCK *et al*, 2019).

Em Gressin *et al*, (2013) a abordagem baseada em pontos de interesse é objeto de estudo onde para cada ponto são calculados descritores locais sobre a estrutura de sua vizinhança e definem seu método como um refinamento do registro grosseiro. Theiler *et al*, (2014) propuseram um método chamado K-4PCS (*Keypoint-based 4-Points Congruent Sets*), uma adaptação do trabalho de Aiger *et al*, (2008), para o registro entre pares de nuvens de pontos 3D adquiridas com um TLS. Basicamente, os autores incluem a extração de pontos de interesse a partir do operador DoG (*Difference of Gaussian*) e um processo de reamostragem por voxelização. Em ambos os trabalhos, há a necessidade de uma aproximação inicial entre os pares de nuvens de pontos 3D. Segundo Theiler *et al*, (2014), se não for aplicado um método de registro grosseiro robusto, não há garantia da convergência da função custo das distâncias entre os pontos para um mínimo global.

Em Vlaminck *et al*, (2019) é apresentado um método para a detecção de *loop closures* (lugares revisitados) com dados de Velodyne utilizando descritores globais e uma técnica para o registro de nuvens de pontos LiDAR 3D no modo grosseiro-para-fino utilizando o método 4PCS (AIGER *et al*, 2008) e o algoritmo ICP. Entretanto, mesmo com um processo de reamostragem das nuvens de pontos LiDAR 3D, todos os processos referentes ao registro grosseiro utilizado: seleção de uma base coplanar, determinação de conjuntos congruentes e aplicação de testes para determinar a melhor transformação de corpo rígido entre as nuvens são onerosos computacionalmente. Um resumo dos métodos apresentados no estado da arte com a abordagem baseada em pontos de interesse pode ser observado no Quadro 1.

Em Gruen e Akca (2005) é proposto um método baseado no ajuste de superfícies 3D chamado LS3D (*Least Squares 3D Surface Matching*). O método associa uma ou mais superfícies implícitas a um modelo de superfícies pré-existente, minimizando a soma dos quadrados das distâncias euclidianas entre as superfícies. Uma vantagem do algoritmo proposto pelos autores é a possibilidade de propagação dos erros por meio da ponderação da MC (Matriz de Covariâncias) para obter uma melhor estimativa dos parâmetros de transformação. Entretanto, o método busca um mínimo local ótimo e necessita de valores iniciais para convergir para a solução ótima.

QUADRO 1 - RESUMO DOS TRABALHOS INVESTIGADOS NO ESTADO DA ARTE COM A ABORDAGEM PONTOS DE INTERESSE.

Trabalho	Dados utilizados	Cálculo dos parâmetros de transformação	Vantagens	Desvantagens
Gressin et al, (2013)	ALS; TLS; MMS	ICP	Foi testado com grande diversidade de ambientes e funcionou bem em todos eles.	Alto custo computacional; Iterativo; Necessita de um alinhamento inicial (registro grosseiro).
Theiler et al, (2014)	TLS estático	K-4PCS + ICP	Não necessita de alta sobreposição entre os pares de nuvens de pontos.	Alto custo computacional; Necessita de um registro fino.
Vlaminck et al, (2019)	Velodyne LiDAR móvel	4PCS + ICP	Funcionou bem com diferentes versões do Velodyne.	Alto custo computacional; Iterativo; Necessita realizar um processo de segmentação e extração de segmentos para acelerar o 4PCS.

FONTE: O autor (2020).

Em Deschaud (2018) é apresentada uma solução de SLAM no formato *scan-to-model* a partir da reamostragem dos pontos obtidos por um sensor Velodyne e de um processo iterativo de minimização da distância entre os pontos reamostrados na nuvem de pontos LiDAR 3D de Pesquisa e os pontos projetados nas superfícies implícitas estimadas na nuvem de pontos 3D global (modelo). Essa representação pode ser definida por uma abordagem chamada IMLS (*Implicit Moving Least Square*) que transforma os pontos pertencentes ao modelo em superfícies implícitas por um processo similar a uma interpolação. Para a estimativa dos parâmetros, uma função não-linear de minimização das distâncias entre os pontos e as superfícies implícitas é definida. Assim,

o algoritmo iterativamente projeta pontos nas superfícies IMLS e calcula os parâmetros de transformação (parametrização linear da rotação) entre a nuvem e o modelo, até que o máximo de iterações seja atingido. Além do custo computacional considerável devido ao processo de reamostragem, o método é dependente somente de pontos, assim, no caso de obstrução do sensor pode levar a convergência do algoritmo a um mínimo local. Um resumo dos métodos apresentados no estado da arte com a abordagem baseada em Superfícies Implícitas pode ser observado no Quadro 2.

QUADRO 2 - RESUMO DOS TRABALHOS INVESTIGADOS NO ESTADO DA ARTE COM A ABORDAGEM DE SUPERFÍCIES IMPLÍCITAS.

Trabalho	Dados utilizados	Cálculo dos parâmetros de transformação	Vantagem	Desvantagem
Gruen e Akca (2005)	LASER Scanning; Stereo Matching	Least Squares 3D Surface Matching (LS3D)	Propagação dos erros por meio da ponderação da MC.	Necessita de um alinhamento inicial (registro grosseiro).
Deschaut (2018)	Velodyne LiDAR móvel	IMLS (<i>Implicit Moving Least Square</i>) + ICP	Não necessita de processos de segmentação de feições no conjuntos de dados.	Processo iterativo de minimização da distância entre os pontos; Custo computacional elevado em decorrência dos processos de estimativa dos vetores normais dos pontos a partir de suas vizinhanças.

FONTE: o autor (2020)

Dois trabalhos são considerados precursores das abordagens baseada em feições: Besl e Mckay (1992) e Chen e Medioni (1992), ambos iterativos e com o estabelecimento de correspondências ponto a ponto e ponto a plano, respectivamente. Besl e Mckay (1992) propuseram o ICP onde os parâmetros de transformação (rotação e translação) são estimados a partir da determinação de pontos pseudo-conjugados nas nuvens de pontos iterativamente até que o erro de alinhamento entre as nuvens de pontos 3D seja menor que um limiar pré-estabelecido ou se estabilize. Segundo os autores, uma vantagem de seu algoritmo é a convergência monotônica para um mínimo global segundo uma função objetivo da distância média quadrática entre os pontos.

Já Chen e Medioni (1992) propuseram utilizar a distância ponto-a-plano em vez da distância ponto-a-ponto. Essa medida é tomada entre os pontos da nuvem de Pesquisa

em relação à superfície definida localmente por planos tangentes da nuvem de pontos de Referência. Para encontrar o ponto de origem, os autores usaram uma técnica de pesquisa baseada no método de Newton-Raphson em um sistema de coordenadas ortogonal. Segundo os autores, para o algoritmo ICP, as correspondências ponto-a-plano podem levar a um modelo de estimativa mais robusta e com um custo computacional menor. Entretanto, tanto os algoritmos apresentados em Besl e Mckay (1992) quanto Chen e Medioni (1992) são dependentes de parâmetros de transformação iniciais para um resultado ótimo.

Em Rusinkiewicz e Levoy (2001) são investigadas variações do ICP com a utilização de diferentes primitivas e segundo seus estágios: seleção e correspondência de pontos, sistema de ponderação das observações, remoção de pontos espúrios, o MMF (Modelo Matemático Funcional) da função erro e o seu método de minimização. Os autores propõem soluções para todas as etapas, principalmente no processo de reamostragem dos pontos. Segundo Rusinkiewicz e Levoy (2001), em conjuntos de dados esparsos, volumosos e com a presença de pontos espúrios, as abordagens mais indicadas são aquelas baseadas em planos.

Outro problema recorrente no registro de nuvens de pontos 3D é a representação da rotação, sendo na maioria das vezes parametrizada por ângulos de Euler, o que pode levar a problemas, por exemplo, com o *Gimbal Lock* (perda de um grau de liberdade do Espaço Canônico quando dois eixos se alinham³). Assim, uma solução numericamente mais estável é a representação pela álgebra de Quaternions para a rotação, ou Dual Quaternions para estimar a translação no mesmo modelo. Em Walker *et al*, (1991) Dual Quaternions são usados para estimar os parâmetros de transformação pela minimização de uma função custo associada a soma dos erros da posição e orientação juntas. Em Wang *et al*, (2014) o parâmetro de escala é introduzido diretamente no cálculo dos parâmetros de transformação dentro da Álgebra de Dual Quaternions. Entretanto, ambos os trabalhos usam pontos conjugados para a determinação dos parâmetros, ao qual não é indicado para nuvens de pontos LiDAR 3D obtidas pelo Velodyne, devido à geometria das linhas de varredura dos sensores LiDAR, onde não existe uma correspondência direta entre essas primitivas.

³ Para mais detalhes ver KENWRIGHT (2020).

Pavan e dos Santos (2017) utilizaram a otimização por grafos e a representação da matriz de Rotação a partir de Quaternions baseado em uma abordagem plano a plano para a determinação dos parâmetros de rotação e translação entre pares de nuvens de pontos 3D. O método é feito em duas etapas. Primeiro, os parâmetros de rotação são calculados por SVD (*Singular Value Decomposition*) e globalmente refinados com uma estratégia proposta baseada nas propriedades de rotação dos Quaternions. Em seguida é utilizado o método de LUM (LU e MILLIOS, 1997) adaptado para o caso 3D no refinamento global apenas dos parâmetros de translação. Entretanto, no referido trabalho um método subótimo de correspondências dos planos é utilizado e a ocorrência de um Espaço Canônico incompleto não é investigada.

Grant *et al*, (2019) apresentam uma solução do SLAM com propostas para a detecção de planos especificamente pensada para o sensor Velodyne, e, com a inclusão de pontos segundo o Espaço Canônico gerado. Primeiramente, os planos são extraídos por meio de um detector similar ao proposto por Bormann *et al*, (2011), em seguida, é calculada uma matriz de restrições para determinar quais as direções no Espaço Canônico necessitam de observações com as primitivas ponto. Assim, planos e pontos são usados para estimar os parâmetros de transformação em um modelo de otimização não-linear. Para a correspondência dos planos, em Grant *et al*, (2019) são utilizados apenas dois critérios para estabelecer a correspondência: inclinação entre as normais e proporção entre os valores dos parâmetros de distância do plano a origem para os dois planos candidatos; ambos comparados a um limiar. Diante da complexidade e pluralidade de feições em ambientes urbanos, apenas dois critérios são insuficientes para garantir uma correspondência biunívoca ótima entre os planos, principalmente quando o algoritmo de detecção dos planos não corrobora.

Em Grant *et al*, (2019), os pontos adicionados ao cálculo dos parâmetros de transformação são pseudo-conjugados, por isso, há necessidade de iterações do algoritmo para a estimativa de parâmetros de transformação. Embora os autores tenham limitado o número máximo de pontos para diminuir o custo computacional, o processo de estabelecimento de pseudo-conjugados (estabelecimento do ponto mais próximo na nuvem subsequente como o correspondente) com dados obtidos com o Velodyne não é recomendado devido à forma como o sensor funciona, de acordo com Grant *et al*, (2013),

onde o formato das linhas de varredura em planos horizontais podem mudar totalmente de uma nuvem de pontos LiDAR 3D para outra. Um resumo dos métodos apresentados no estado da arte com a abordagem baseada em Feições pode ser observado no Quadro 3.

QUADRO 3 - RESUMO DOS TRABALHOS INVESTIGADOS NO ESTADO DA ARTE COM A ABORDAGEM FEIÇÕES.

Trabalho	Dados utilizados	Primitivas	Cálculo dos parâmetros de transformação	Vantagem	Desvantagem
Besl e Mckay (1992); Chen e Medioni (1992)	Imagens de profundidade	Ponto a Ponto; Ponto a Plano	Método de Horn <i>et al</i> , (1987); Método de Newton	Não necessita de processos de segmentação e extração de feições; É eficiente quando os parâmetros iniciais são próximos aos valores ótimos.	Alto custo computacional; Baixa robustez a presença de pontos espúrios e necessita de alta taxa de sobreposições entre os conjuntos de pontos.
Wang <i>et al</i>, (2014)	Simulados	Ponto a Ponto	Analítico (Dual Quaternions)	Método direto de obtenção dos parâmetros de transformação.	Necessita de um método de estabelecimento de correspondências entre pontos.
Pavan <i>et al</i>, (2017)	TLS estático	Plano a Plano	Método de Horn <i>et al</i> , (1987) + ICP	Converge rápido por utilizar o MMQ linear e SVD baseado em Horn (1987).	Não investiga a possibilidade de um Espaço Canônico 3D não ser gerado pelos planos; Não aplica a dados obtidos com o sensor Velodyne; Método de correspondência entre planos subótimo.
Grant <i>et al</i>, (2019)	Velodyne LiDAR móvel	Planos e Pontos	IC3PO (<i>Iterative Closet Point Plus Plane Optimization</i>)	Algoritmo de extração de planos adequado aos dados; Adiciona apenas pontos na direção ausente do Espaço Canônico definido pelos planos.	Iterativo; Método de correspondência entre planos subótimo; Atribuição de pontos pseudo-conjugados.

FONTE: o autor (2020)

As principais contribuições do presente trabalho são:

- a) Proposta de um método fundamentado nos algoritmos de rotulação por relaxação para a correspondência entre planos presentes nas nuvens de pontos LiDAR 3D a partir de uma Matriz de Probabilidades *a priori* obtida pela associação

de funções de probabilidades que consideram características geométricas dos planos, e, a sua posterior atualização por Inferência Bayesiana a partir de relações locais dos planos correspondidos.

b) Método analítico de determinação dos parâmetros de transformação entre pares de nuvens de pontos LiDAR 3D diretamente com os parâmetros dos planos e a álgebra de Dual Quaternions.

c) Se os planos correspondidos gerarem um Espaço Canônico bidimensional, é proposto o uso de *clusters* (superfícies quase-planares descartadas pelo algoritmo crescimento de região) para criar uma zona de busca por pontos virtuais de interesse que, de forma direta e com a mesma álgebra de Dual Quaternions, determinam os parâmetros de transformação ótimos.

d) Caso não se detectem *clusters* com a direção do Espaço Canônico não gerado pelos planos, é proposto um método de aproximação chamado de *Sliding*, em que um plano de referência acima do terreno é deslizado sobre seu correspondente até um mínimo global das distâncias ponto a ponto das suas extremidades.

1.4. ESTRUTURA DO TRABALHO

Este trabalho está dividido em seis capítulos. O primeiro refere-se às considerações iniciais do objeto de estudo, uma contextualização do problema abordado, objetivos e uma discussão sobre o estado da arte do assunto. No segundo capítulo é realizado um levantamento dos principais conceitos relacionados a temática abordada. No terceiro capítulo são descritos os materiais utilizados e o método proposto neste trabalho. No quarto capítulo são apresentados os ensaios experimentais e a discussão dos resultados. Por fim, no quinto capítulo são apresentadas as conclusões e as recomendações para trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo tem como objetivo apresentar os fundamentos teóricos necessários para compreensão do problema discutido no trabalho. Primeiramente é apresentada uma descrição do sensor utilizado nesta pesquisa para a geração das nuvens de pontos LiDAR 3D. Em seguida, uma breve descrição dos principais métodos e algoritmos disponíveis na literatura utilizados nesta pesquisa: RANSAC, Crescimento de Região, Dual Quaternions, ICP e LUM.

2.1. SENSOR VELODYNE

O sensor LiDAR Velodyne HDL-32E, mostrado na Figura 3, foi criado em 2011 e é considerado uma versão mais compacta e acessível do Velodyne HDL-64E. Originalmente a versão HDL-64E foi projetada para a competição DARPA (*Defense Advanced Research Project Agency's*)⁴ “Grand Challenge II” em 2007. Cinco dos seis finalistas da competição utilizaram essa tecnologia (SCHWARZ, 2010). Mais informações sobre o funcionamento do sistema de emissão e recepção LASER presente em sensores LiDAR em geral, ver Shan e Toth (2010).

FIGURA 3 – SENSOR VELODYNE HDL-32

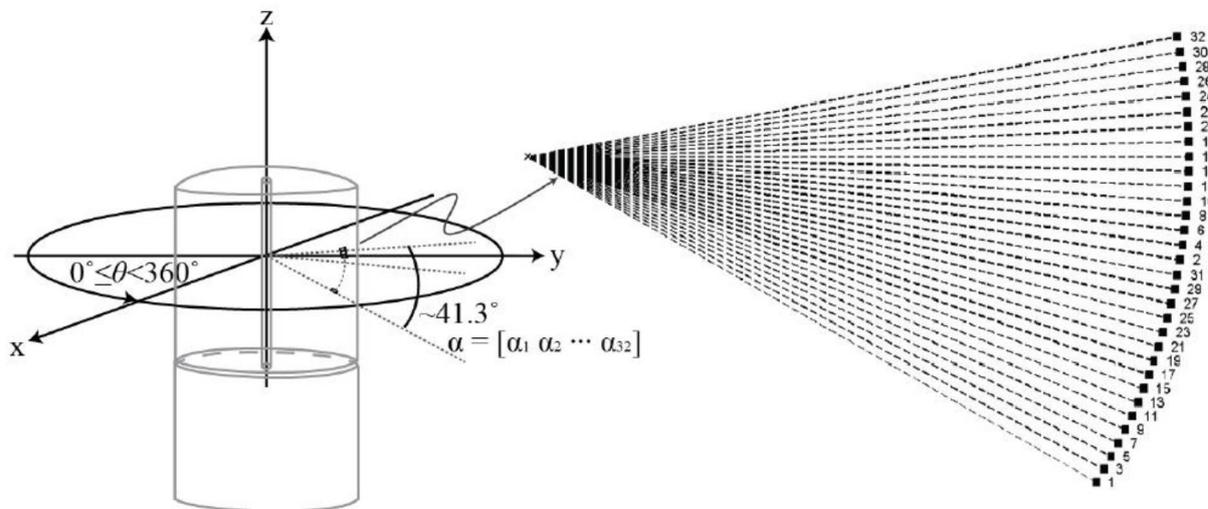


FONTE: Adaptado de VELODYNE (2012)

⁴ DARPA Grand Challenge é uma competição entre veículos autônomos patrocinada pelo Departamento de Defesa dos Estados Unidos de Universidades e empresas nacionais. Disponível em: <<https://www.darpa.mil/about-us/timeline/darpa-urban-challenge>>. Acesso em 07 de Maio de 2020.

A versão HDL-32E é composta por 32 feixes de LASER posicionados lado a lado radialmente e fixados em um painel como mostra a Figura 4. Em pleno funcionamento, esse painel gira em torno de seu eixo vertical continuamente cerca de 10 vezes por segundo (pode ser atribuído entre 7 e 14 para esta versão). Seu FoV é de aproximadamente 41.34° (α) e 360° (θ) nas direções vertical e horizontal, respectivamente. Sua frequência de emissão do pulso LASER é de aproximadamente 700.000 pulsos por segundo, com um alcance de até 100 metros e uma acurácia de aproximadamente 2 cm para a medida de distância do LASER (VELODYNE, 2012).

FIGURA 4 – DESCRIÇÃO DO SISTEMA DE REVOLUÇÃO APRESENTADO PELO SENSOR VELODYNE HDL-32 COM DESTAQUE AOS 32 LASERS E SUAS POSIÇÕES E ORIENTAÇÕES.



FONTE: Adaptado de em CHAN E LITCH (2015)

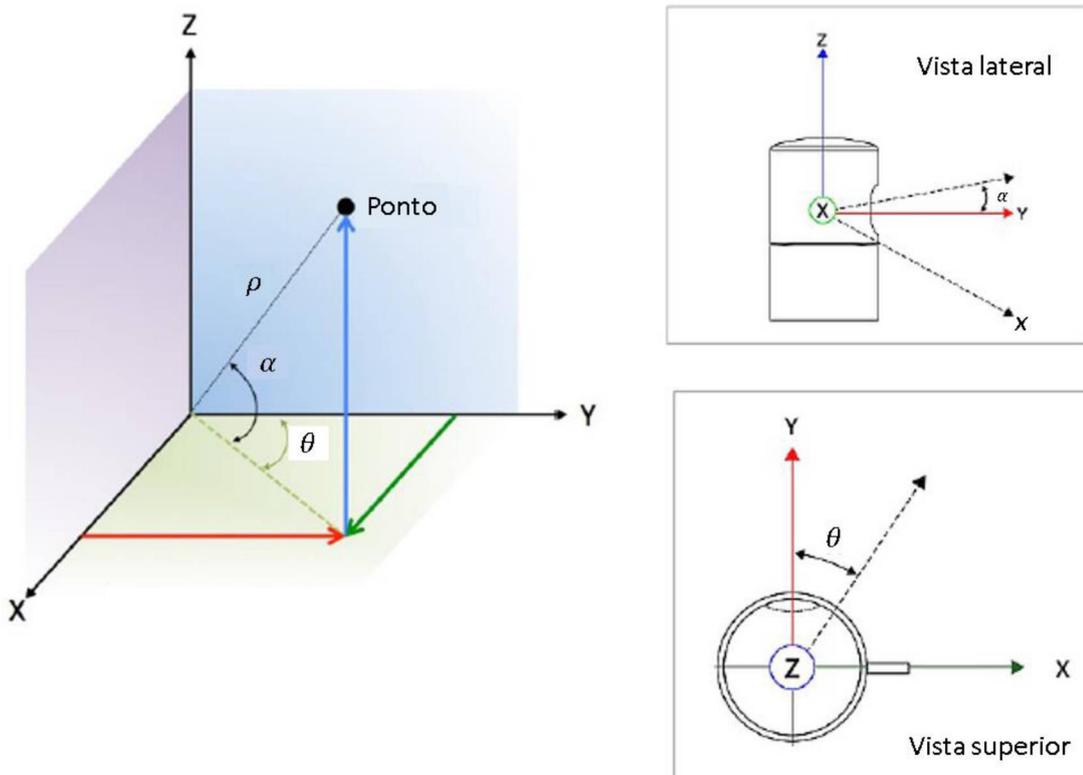
Na Figura 4, a diferença angular entre os valores de α são de $1,3^\circ$ dentro de um intervalo de $+10,67^\circ$ a $-30,67^\circ$ em relação a vertical (VELODYNE, 2012).

Para o cálculo das distâncias (ρ) de cada pulso LASER dos 32 em funcionamento, o pulso LASER é gerado e uma parte desse pulso é usada para comunicar o início da contagem de tempo no canal de recepção enquanto a outra parte é responsável por interagir com a superfície física de um alvo, ou parte dele, seja detectado pelo receptor. Antes de ser amplificada, a energia radiante do pulso é convertida em corrente elétrica e a contagem de tempo é finalizada. Assim, o sensor Velodyne usa o princípio ToF (*Time of Flight*) para o cálculo dos valores de ρ . A Equação (1) é usada para o cálculo do alcance (ρ) de cada ponto LiDAR 3D (SHAN e TOTH, 2010).

$$\rho = \frac{\Delta t}{2} c \quad (1)$$

sendo c a velocidade da luz ($\approx 3 \times 10^8$ m/s) e Δt o tempo medido entre a emissão e a recepção do pulso.

FIGURA 5 – ILUSTRAÇÃO DO SISTEMA DE COORDENADAS DO SENSOR VELODYNE HDL-32.



FONTE: Adaptado de VELODYNE (2012)

De acordo com o manual de usuários (VELODYNE, 2012) e como pode ser visto na Figura 5, para referenciar um ponto (i) ao sistema de coordenadas do sensor, calculado a partir de uma revolução do sensor na posição (k) pelo LASER (j), é usada a seguinte equação:

$$\begin{pmatrix} X_{ijk} \\ Y_{ijk} \\ Z_{ijk} \end{pmatrix} = \begin{pmatrix} \rho_{ijk} \cos(\alpha_j) \text{sen}(\theta_{ijk}) \\ \rho_{ijk} \cos(\alpha_j) \text{cos}(\theta_{ijk}) \\ \rho_{ijk} \text{sen}(\alpha_j) \end{pmatrix} \quad (2)$$

sendo ρ e θ o alcance e o ângulo horizontal, respectivamente, e α um valor constante para o ângulo vertical do LASER j representado por um índice, como mostra a Tabela 1 a seguir.

TABELA 1 – ÂNGULOS VERTICAIS α SEGUNDO A POSIÇÃO DE CADA LASER NO PAINEL.

LASER id	Ângulo vertical (°)	LASER id	Ângulo vertical (°)
1	-30.67	17	-20.00
2	-9.33	18	1.33
3	-29.33	19	-18.67
4	-8.00	20	2.67
5	-28.00	21	-17.33
6	-6.67	22	4.00
7	-26.67	23	-16.00
8	-5.33	24	5.33
9	-25.33	25	-14.67
10	-4.00	26	6.67
11	-24.00	27	-13.33
12	-2.67	28	8.00
13	-22.67	29	-12.00
14	-1.33	30	9.33
15	-21.33	31	-10.67
16	0.00	32	10.67

FONTE: Velodyne (2012)

O resultado é um conjunto de pontos LiDAR 3D referenciados ao sistema de coordenadas do sensor Velodyne juntamente com o valor da intensidade de retorno do pulso e o índice do LASER utilizado. A seguir é apresentado o algoritmo utilizado nesta pesquisa para a segmentação de superfícies planas presentes nas nuvens de pontos 3D.

2.2. ALGORITMO RANSAC

O algoritmo RANSAC (*RANdon SAmpleS Consensus*) idealizado por Fischler e Bolles, (1981) é baseado na noção de conjuntos mínimos, que consiste no menor número de amostras necessárias para de maneira unívoca definir um modelo. Basicamente, para a segmentação de superfícies planares em nuvens de pontos 3D, o algoritmo se divide em (FISHER E BOLLES, 1981):

- 1 - Dada uma nuvem de pontos LiDAR 3D, é selecionado aleatoriamente um subconjunto de pontos com número mínimo necessário para calcular os parâmetros do plano.
- 2 - É verificado se os pontos do conjunto selecionado randomicamente pertencem à superfície definida pelo critério de proximidade formando um conjunto chamado de consenso.
- 3 - Se o número de pontos do conjunto consenso (também chamados de *inliers*) exceder um limiar pré-determinado, então os parâmetros do plano são recalculados usando todos os *inliers*.
- 4 - Caso contrário, as etapas 1, 2 e 3 são repetidas.
- 5 - Depois de algum número k pré-determinado de repetições das etapas anteriores, o conjunto consenso com maior número de *inliers* é utilizado para recalculiar os parâmetros do plano.

O número máximo de repetições k é determinado estocasticamente. Seja γ a probabilidade de que qualquer ponto do conjunto de dados ser um *inlier*, então a probabilidade de escolher s boas amostras (*inliers*) de pontos é γ^s . Assim, para k repetições, a probabilidade de insucesso é o complemento $(1 - \gamma^s)^k$. Portanto, para garantir uma probabilidade p de que pelo menos uma seleção aleatória de pontos é um conjunto sem pontos espúrios, faz-se necessário pelo menos k repetições (FISCHLER e BOLLES, 1981). Logo, a seguinte relação é obtida:

$$1 - p = (1 - \gamma^s)^k \quad (3)$$

Para um conjunto de correspondências com um respectivo percentual γ de *inliers*, o número de tentativas k necessárias para suficientemente oferecer uma probabilidade máxima p na escolha de um conjunto sem pontos espúrios consistindo de s correspondências entre pontos é dada por (FISCHLER e BOLLES, 1981):

$$k = \frac{\log(1 - p)}{\log(1 - \gamma^s)} \quad (4)$$

A probabilidade p é determinada dentro do intervalo $0,95 \leq p \leq 0,99$ (para mais detalhes ver FISHER E BOLLES, 1981).

2.3. ALGORITMO CRESCIMENTO DE REGIÃO E DETERMINAÇÃO DOS PARÂMETROS DOS PLANOS

O algoritmo Crescimento de Região (RABBANI *et al*, 2006) utiliza os vetores normais calculados para cada ponto segundo sua vizinhança bem como seus resíduos para agrupar pontos segmentados em uma mesma superfície plana. A partir do processo de segmentação, pressupõe-se que os pontos pertencentes a um mesmo segmento representem alguma feição planar, ou seja, estão conectados localmente. Portanto, esses pontos devem apresentar vetores normais com inclinações próximas entre si uma vez que representam a mesma superfície plana. Para avaliar essa condição, um limiar (τ) é comparado aos ângulos de pontos considerados como “sementes” com seus vizinhos da seguinte forma: $\|\vec{n}_p \cdot \vec{n}_s\| > \cos(\tau)$, sendo \vec{n}_p e \vec{n}_s os vetores normais do ponto semente e do ponto candidato, respectivamente.

A partir da seleção aleatória de uma superfície, cada ponto desta superfície é comparado com o seu vizinho e se este se ajusta ao plano definido, é agregado ao plano. Assim, os pontos vizinhos são incorporados à superfície obedecendo a certos critérios e, a cada vez que um ponto é adicionado, a equação do plano é atualizada. A decisão para aceitar um ponto no plano pode ser baseada em um ou mais critérios, dentre os quais se podem citar (VOSSELMAN *et al*, 2004):

- Proximidade: somente pontos que estão a uma certa distância da superfície teste são adicionados a este plano.
- Curvatura local: um ponto candidato é aceito em um segmento se a sua distância ortogonal ao plano formado por todos os pontos existentes no segmento está dentro de certo limiar.
- Superfície suavizada: para aplicar este critério, uma superfície normal local para cada ponto da nuvem é considerada. O ponto candidato é aceito se o ângulo entre a superfície normal local do ponto e a normal da superfície estiver abaixo de um limiar pré-estabelecido.

A partir do processo de extração dos planos pelo algoritmo Crescimento de Região, os parâmetros dos planos podem ser estimados utilizando SVD por meio da equação geral do plano $n_x X + n_y Y + n_z Z + d = 0$, sendo (n_x, n_y, n_z) o vetor normal e d

a distância dos planos a origem (parâmetros do plano) e (X, Y, Z) as coordenadas 3D de um ponto pertencente ao plano. Considerando a minimização da soma dos quadrados dos resíduos de cada ponto em relação ao modelo, os parâmetros dos planos são determinados pelo Método dos Mínimos Quadrados (para mais detalhes ver GEMAEL 1994) utilizando a equação a seguir (PATHAK *et al*, 2010):

$$\sum_{i=1}^n \frac{r_i^2}{\sigma_i^2} = \min \quad (5)$$

sendo $r_i = n_x X_i + n_y Y_i + n_z Z_i + d$ o resíduo do plano em relação ao ponto $P_i = [X_i \ Y_i \ Z_i]^T$ com $i = 1, 2, \dots, n$; n o número de pontos atribuídos ao plano e σ_i^2 o valor do peso para cada ponto determinado segundo o valor da variância da distância do ponto P_i à origem. O somatório na Equação (5) pode ser reescrito por multiplicação de matrizes, como segue:

$$[n_x \ n_y \ n_z \ d] H \begin{bmatrix} n_x \\ n_y \\ n_z \\ d \end{bmatrix} = \min \quad (6)$$

sendo $H = \sum_{i=1}^n [P_i \ 1]^T [P_i \ 1]$ quadrada e simétrica, o que implica que essa matriz apresenta autovalores reais, $\lambda_1, \lambda_2, \lambda_3$ e λ_4 ($\lambda_1 > \lambda_2 > \lambda_3 > \lambda_4$) com os respectivos autovetores associados v_1, v_2, v_3 e v_4 , unitários e ortogonais entre si. Desta forma, os parâmetros do plano são obtidos por meio das componentes do autovetor v_4 , que corresponde ao menor autovalor λ_4 da matriz H . Posteriormente, esse vetor é normalizado, como mostra a Equação (7):

$$\begin{bmatrix} n_x \\ n_y \\ n_z \\ d \end{bmatrix} = \frac{1}{\sqrt{1 - v_{4W}^2}} \begin{bmatrix} v_{4X} \\ v_{4Y} \\ v_{4Z} \\ v_{4W} \end{bmatrix} \quad (7)$$

A partir dos autovalores a curvatura do plano pode ser calculada pela Equação (8) (RUSU, 2010) :

$$c = \frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3} \quad (8)$$

2.4. ÁLGEBRA DE DUAL QUATERNIONS

Quaternions foram idealizados por Hamilton W. R. em 1843 (HAMILTON, 1866) e em Clifford (1873) é proposta uma generalização chamada *biquaternions*. Em Mcaulay (1898) é definido a dupla unidade ε com $\varepsilon^2 = 0$ resultando na álgebra de Dual Quaternions. Posteriormente, Kotelnikov (1895) e Study (1901) aplicaram o Dual Quaternions no estudo da mecânica unificando a translação e a rotação em um mesmo modelo algébrico. Mais informações sobre Quaternions ver Galo e Tozzi (2001).

2.4.1. Números Duais

Definição (CLIFFORD, 1882): Seja $r, d \in R$, qualquer dual número z_d pode ser escrito como:

$$z_d = r + \varepsilon d \quad (9)$$

sendo $\varepsilon^2 = 0$ com $\varepsilon \neq 0$, um elemento base chamada dupla unidade, r a parte real (não-dual) e d a parte dual. Como pode ser visto na Equação (9), dual números se assemelham aos números complexos. O conjugado de um numero dual é análogo ao conjugado dos números complexos, isto é:

$$z'_d = r - \varepsilon d \quad (10)$$

A adição de dois números duais z_{d1} e z_{d2} é dada por (BERGEN, 2009):

$$z_{d1} + z_{d2} = (r_1 + \varepsilon d_1) + (r_2 + \varepsilon d_2) = (r_1 + r_2) + \varepsilon(d_1 + d_2) \quad (11)$$

A multiplicação de dois números duais z_{d1} e z_{d2} é dada por (BERGEN, 2009):

$$z_{d1}z_{d2} = (r_1 + \varepsilon d_1)(r_2 + \varepsilon d_2) = r_1r_2 + \varepsilon d_1r_2 + r_1\varepsilon d_2 + \varepsilon^2 d_1d_2 = r_1r_2 + \varepsilon(r_2d_1 + r_1d_2) \quad (12)$$

Os números duais puros, ou seja, os números duais com $r = 0$, não tem um inverso multiplicativo (CLIFFORD, 1873). Esta é uma diferença fundamental de números complexos, porque cada número complexo diferente de zero tem o inverso. O inverso multiplicativo de um número dual com $r \neq 0$ é aquele que assegura a expressão (BERGEN, 2009):

$$z_d^{-1}z_d = z_d z_d^{-1} = 1 \quad (13)$$

Como, $z_d' z_d = z_d' z_d = r^2$ então dividindo ambos os lados por r^2 temos $\frac{z_d' z_d}{r^2} = \frac{z_d z_d'}{r^2} = 1$ logo (BERGEN, 2009):

$$z_d^{-1} = \frac{z_d'}{r^2} \quad (14)$$

2.4.2. Dual Quaternions

Definição (STUDY 1901): Seja $q, \hat{q} \in H$, qualquer Dual Quaternion q_d pode ser escrito como a soma de dois Quaternions (CLIFFORD, 1882):

$$q_d = q + \varepsilon \hat{q} \quad (15)$$

sendo $\hat{q} = \hat{q}_0 + \hat{q}_1 i + \hat{q}_2 j + \hat{q}_3 k$ e $q = q_0 + q_1 i + q_2 j + q_3 k$. Os Dual Quaternions também podem ser escritos como (CLIFFORD, 1882):

$$\begin{aligned} q_d = q + \varepsilon \hat{q} &= (q_0 + q_1 i + q_2 j + q_3 k) + \varepsilon(\hat{q}_0 + \hat{q}_1 i + \hat{q}_2 j + \hat{q}_3 k) = \\ &= (q_0 + \varepsilon \hat{q}_0) + (q_1 + \varepsilon \hat{q}_1) i + (q_2 + \varepsilon \hat{q}_2) j + (q_3 + \varepsilon \hat{q}_3) k = \\ &= q_{d0} + q_{d1} i + q_{d2} j + q_{d3} k \end{aligned} \quad (16)$$

sendo q_{d0}, q_{d1}, q_{d2} e q_{d3} dual números.

Dados dois Quaternions:

$$\begin{aligned} p_d = p + \varepsilon \hat{p} &= (p_0 + \varepsilon \hat{p}_0) + (p_1 + \varepsilon \hat{p}_1) i + (p_2 + \varepsilon \hat{p}_2) j + (p_3 + \varepsilon \hat{p}_3) k \\ q_d = q + \varepsilon \hat{q} &= (q_0 + \varepsilon \hat{q}_0) + (q_1 + \varepsilon \hat{q}_1) i + (q_2 + \varepsilon \hat{q}_2) j + (q_3 + \varepsilon \hat{q}_3) k \end{aligned} \quad (17)$$

são iguais, ou seja, $p_d = q_d$, se, e somente se $p_0 = q_0$, $p_1 = q_1$, $p_2 = q_2$, $p_3 = q_3$, $\hat{p}_0 = \hat{q}_0$, $\hat{p}_1 = \hat{q}_1$, $\hat{p}_2 = \hat{q}_2$ e $\hat{p}_3 = \hat{q}_3$.

A adição de dois Dual Quaternions q_d e p_d é dada por (CLIFFORD, 1882):

$$p_d + q_d = (p + \varepsilon\hat{p}) + (q + \varepsilon\hat{q}) = (p + q) + \varepsilon(\hat{p} + \hat{q}) \quad (18)$$

O produto de dois Dual Quaternions q_d e p_d é definido como (CLIFFORD, 1882):

$$p_d q_d = (p + \varepsilon\hat{p})(q + \varepsilon\hat{q}) = pq + \varepsilon\hat{p}q + p\varepsilon\hat{q} + \varepsilon^2\hat{p}\hat{q} = pq + \varepsilon(\hat{p}q + p\hat{q}) \quad (19)$$

A multiplicação de Dual Quaternions é associativa, distributiva, mas não comutativa. O conjugado de um Dual Quaternion é definido utilizando o conjugado de Quaternion clássico (CLIFFORD, 1882):

$$q'_d = \hat{q} + \varepsilon\hat{q} \quad (20)$$

No entanto, o conjugado de um número dual pode ser aplicado a um Quaternion duplamente conjugado (CLIFFORD, 1882):

$$q''_d = \hat{q} - \varepsilon\hat{q} \quad (21)$$

A norma de um Dual Quaternion é um dual escalar e é definido como (CLIFFORD, 1882):

$$\|q_d\| = \sqrt{(q_0 + \varepsilon\hat{q}_0)^2 + (q_1 + \varepsilon\hat{q}_1)^2 + (q_2 + \varepsilon\hat{q}_2)^2 + (q_3 + \varepsilon\hat{q}_3)^2} = \sqrt{q_d q'_d} \quad (22)$$

onde a raiz é assumido positiva. Um Dual Quaternion é chamado unitário se sua norma é 1 ($\|q_d\| = 1$).

Nota-se que um Dual Quaternion q_d é unitário, se, e somente se (CLIFFORD, 1882):

$$\|q\| = 1 \text{ e } \langle q, \hat{q} \rangle = 0 \quad (23)$$

Com efeito, se $\|q\| = 1$ e $\langle q, \hat{q} \rangle = 0$ então:

$$\begin{aligned} \|q\| &= \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1 \text{ e } \langle q, \hat{q} \rangle = q_0\hat{q}_0 + q_1\hat{q}_1 + q_2\hat{q}_2 + q_3\hat{q}_3 = 0 \\ \|q_d\| &= \sqrt{(q_0 + \varepsilon\hat{q}_0)^2 + (q_1 + \varepsilon\hat{q}_1)^2 + (q_2 + \varepsilon\hat{q}_2)^2 + (q_3 + \varepsilon\hat{q}_3)^2} = \\ &\sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2 + 2\varepsilon(q_0\hat{q}_0 + q_1\hat{q}_1 + q_2\hat{q}_2 + q_3\hat{q}_3)} = \sqrt{1 + 2\varepsilon(0)} = 1 \end{aligned} \quad (24)$$

Por outro lado, se $\|q_d\| = 1$, ou seja:

$$\|q_d\|^2 = 1 = q_0^2 + q_1^2 + q_2^2 + q_3^2 + 2\varepsilon(q_0\hat{q}_0 + q_1\hat{q}_1 + q_2\hat{q}_2 + q_3\hat{q}_3) \quad (25)$$

então $2\varepsilon(q_0\hat{q}_0 + q_1\hat{q}_1 + q_2\hat{q}_2 + q_3\hat{q}_3) = 0$ e $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$, logo $\|q\| = 1$ e $\langle q, \hat{q} \rangle = 0$.

O inverso multiplicativo de um Dual Quaternion com \hat{q} nulo ($\hat{q}_0 = \hat{q}_1 = \hat{q}_2 = \hat{q}_3 \neq 0$) é aquele que assegura a expressão (CLIFFORD, 1882):

$$q_d^{-1}q_d = q_dq_d^{-1} = 1 \quad (26)$$

Como, $\acute{q}_dq_d = \acute{q}_dq_d = q\acute{q} = \|q\|^2$ então dividindo ambos os lados por $\|q\|^2$ tem-se $\frac{\acute{q}_dq_d}{\|q\|^2} = \frac{q_d\acute{q}_d}{\|q\|^2} = 1$ logo (CLIFFORD, 1882):

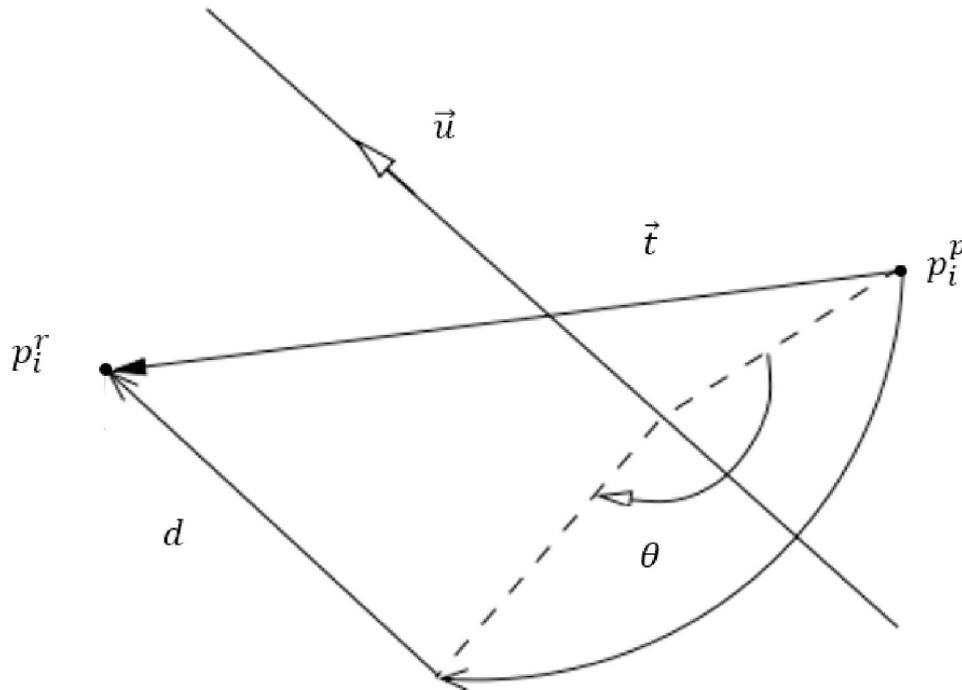
$$q_d^{-1} = \frac{\acute{q}_d}{\|q\|^2} \quad (27)$$

2.4.3. Representação da Transformação de Corpo Rígido por Dual Quatérnions

Um método utilizado para representar transformações de corpo rígido no espaço é baseado no uso de Dual Quaternions. Segundo Walker *et al*, (1991) e a partir do Teorema de Chasles (CHASLES, 1830), na transformação de um sistema de coordenadas para outro com Dual Quaternions, primeiro o sistema de coordenadas

original é transladado sob a direção de \vec{u} (eixo *screw*) com magnitude d , e, rotacionado por um ângulo θ com relação a uma linha paralela a \vec{u} que passa por um ponto p_i^r , como mostra a Figura 6:

FIGURA 6 - TRANSFORMAÇÃO DE CORPO RÍGIDO ENTRE PONTOS COM A ÀLGEBRA DE DUAL QUATERNIONS.



FONTE: Baseado em GALO e TOZZI (2001)

Na Figura 6, p_i^p representa um ponto qualquer na nuvem LiDAR 3D de Pesquisa e p_i^r um ponto qualquer na nuvem LiDAR 3D de Referência, \vec{t} um vetor de translação, \vec{u} o eixo *screw* representado por um vetor unitário, θ o ângulo de rotação e d o deslocamento obtido por: $(\vec{t} \cdot \vec{u}) \triangleq \langle \vec{t}, \vec{u} \rangle$. Seja $q = q_0 + q = \cos \frac{\theta}{2} + u \text{sen} \frac{\theta}{2}$, a translação e a rotação são dadas pelo operador (WALKER *et al*, 1991) :

$$\begin{aligned} \rho_{qd}(p_d) &= q_d p_d q'_d = q_d (1 + \varepsilon p) \left(\dot{q} + \frac{1}{2} \varepsilon \dot{q} \vec{t} \right) = \left(q + \frac{1}{2} \varepsilon \vec{t} q \right) \left(\dot{q} + \varepsilon \left(p \dot{q} + \frac{1}{2} \dot{q} \vec{t} \right) \right) \\ &= \\ q \dot{q} + \varepsilon \left(q p \dot{q} + \frac{1}{2} q \dot{q} \vec{t} + \frac{1}{2} \dot{q} q \vec{t} \right) &= 1 + \varepsilon \left(q p \dot{q} + \frac{1}{2} \vec{t} + \frac{1}{2} \vec{t} \right) = 1 + \varepsilon (q p \dot{q} + \vec{t}) \end{aligned} \quad (28)$$

sendo $q_d = q + \frac{1}{2}\varepsilon\vec{t}q$ e $p_d = 1 + \varepsilon p = 1 + \varepsilon(p_1i + p_2j + p_3k)$

onde p_d, q_d são os Dual Quaternions unitários, \vec{t} um Quaternion puro e \vec{u} o eixo de rotação.

2.4.4. Cálculo dos Parâmetros de Transformação de Pontos com Dual Quaternions

É possível reescrever a transformação de corpo rígido através de matrizes e Quaternions (YAN-BIN, 2020):

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2(q_0^2 + q_1^2) - 1 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 0 & 2(q_1q_2 + q_0q_3) & 2(q_0^2 + q_2^2) - 1 & 2(q_2q_3 - q_0q_1) \\ 0 & 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & 2(q_0^2 + q_3^2) - 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & M \end{bmatrix} \quad (29)$$

O vetor de translação pode ser escrito em termos dos componentes de Dual Quaternions como (WALKER *et al*, 1991):

$$\hat{q} = \frac{1}{2}\vec{t}q \quad (30)$$

Portanto, o produto do Quaternion puro $\vec{t} = t_1i + t_2j + t_3k$ e do Quaternion unitário $q = q_0 + q_1i + q_2j + q_3k$ pode ser reescrito através da multiplicação de uma matriz 4×4 por um vetor coluna 4×1 , ou seja:

$$\hat{Q} = \frac{1}{2}[R_q]T = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & q_3 & -q_2 \\ q_2 & -q_3 & q_0 & q_1 \\ q_3 & q_2 & -q_1 & q_0 \end{bmatrix} \begin{bmatrix} 0 \\ t_1 \\ t_2 \\ t_3 \end{bmatrix} = \begin{bmatrix} \hat{q}_0 \\ \hat{q}_1 \\ \hat{q}_2 \\ \hat{q}_3 \end{bmatrix} \quad (31)$$

sendo $T = [0 \ t_1 \ t_2 \ t_3]^T$ e $\hat{Q} = [\hat{q}_0 \ \hat{q}_1 \ \hat{q}_2 \ \hat{q}_3]^T$.

Para determinar os parâmetros de transformação entre pontos em diferentes sistemas de coordenadas com a álgebra de Dual Quaternions faz-se necessário um

modelo de minimização do erro para avaliar a contribuição de cada observação no MMF. A seguir, uma descrição do cálculo utilizando o MMQ.

A transformação de corpo rígido entre os pontos p_i^r e p_i^p pode ser escrita como:

$$p_i^r = Mp_i^p + \vec{t} \quad (32)$$

sendo M uma matriz de rotação. A fim de determinar os parâmetros da transformação de corpo rígido, a seguinte equação deve ser minimizada:

$$E = \sum_{i=1}^n \|p_i^r - Mp_i^p - \vec{t}\|^2 = \min \quad (33)$$

sendo n o número de pontos correspondentes estabelecidos em ambas as nuvens de pontos LiDAR 3D. Reescrevendo o somatório da Equação (33):

$$\begin{aligned} E &= \sum_{i=1}^n \|p_i^r - Mp_i^p - \vec{t}\|^2 = \sum_{i=1}^n (p_i^r - Mp_i^p - \vec{t}) \cdot (p_i^r - Mp_i^p - \vec{t}) = \\ &= \sum_{i=1}^n \left(\|p_i^r\|^2 - 2(Mp_i^p) \cdot p_i^r - 2\vec{t} \cdot p_i^r + \|Mp_i^p\|^2 + 2(Mp_i^p) \cdot \vec{t} + \|\vec{t}\|^2 \right) = \\ &= \sum_{i=1}^n \|p_i^r\|^2 - 2 \sum_{i=1}^n \left((Mp_i^p) \cdot p_i^r + \vec{t} \cdot p_i^r - (Mp_i^p) \cdot \vec{t} \right) + \sum_{i=1}^n \|p_i^p\|^2 + n\|\vec{t}\|^2 \end{aligned} \quad (34)$$

Tornando $\sum_{i=1}^n (\|p_i^r\|^2 + \|p_i^p\|^2) = c$ constante, é possível expressar a Equação (34) por meio da equação:

$$E = c - 2 \sum_{i=1}^n \left((Mp_i^p) \cdot p_i^r \right) - 2 \sum_{i=1}^n (\vec{t} \cdot p_i^r) + 2 \sum_{i=1}^n \left(\vec{t} \cdot (Mp_i^p) \right) + n\|\vec{t}\|^2 = \min \quad (35)$$

Assim, buscando um Quaternion $q \in H$ que minimiza o somatório, as rotações podem ser representadas por Quaternions unitários como mostra as simplificações a seguir (YAN-BIN, 2020):

$$\begin{aligned}
E &= c - 2 \sum_{i=1}^n \langle \rho_q(p_i^p), p_i^r \rangle - 2 \sum_{i=1}^n \langle \vec{t}, p_i^r \rangle + 2 \sum_{i=1}^n \langle \vec{t}, \rho_q(p_i^p) \rangle + n \langle \vec{t}, \vec{t} \rangle = \\
&= c - 2 \sum_{i=1}^n \langle (qp_i^p \hat{q}), p_i^r \rangle - 2 \sum_{i=1}^n \langle (\vec{t}q \hat{q}), p_i^r \rangle + 2 \sum_{i=1}^n \langle (\vec{t}q \hat{q}), (qp_i^p \hat{q}) \rangle + n \langle (\vec{t}q \hat{q}), \vec{t} \rangle = \\
&= c - 2 \sum_{i=1}^n \langle (qp_i^p), (p_i^r q) \rangle - 2 \sum_{i=1}^n \langle (\vec{t}q), (p_i^r q) \rangle + 2 \sum_{i=1}^n \langle (\vec{t}q), (qp_i^p) \rangle + n \langle (\vec{t}q), (\vec{t}q) \rangle \quad (36) \\
&= \\
&= c - 2 \sum_{i=1}^n \langle (qp_i^p), (p_i^r q) \rangle - 4 \sum_{i=1}^n \langle \hat{q}, (p_i^r q) \rangle + 4 \sum_{i=1}^n \langle \hat{q}, (qp_i^p) \rangle + 4n \langle \hat{q}, \hat{q} \rangle = \\
&= c - 2 \sum_{i=1}^n \langle (qp_i^p), (p_i^r q) \rangle + 4 \sum_{i=1}^n \langle \hat{q}, (-p_i^r q + qp_i^p) \rangle + 4n \langle \hat{q}, \hat{q} \rangle
\end{aligned}$$

Reescrevendo o produto à direita qp_i^p (Equação (37)) e o produto à esquerda $p_i^r q$ (Equação (38)) através da multiplicação de uma matriz 4×4 por um vetor coluna 4×1 , ou seja:

$$[R_{p_i^p}] Q = \begin{bmatrix} 0 & -p_x^p & -p_y^p & -p_z^p \\ p_x^p & 0 & p_z^p & -p_y^p \\ p_y^p & -p_z^p & 0 & p_x^p \\ p_z^p & p_y^p & -p_x^p & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (37)$$

$$[L_{p_i^r}] Q = \begin{bmatrix} 0 & -p_x^r & -p_y^r & -p_z^r \\ p_x^r & 0 & -p_z^r & p_y^r \\ p_y^r & p_z^r & 0 & -p_x^r \\ p_z^r & -p_y^r & p_x^r & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (38)$$

O produto interno $\langle (qp_i^p), (p_i^r q) \rangle$ pode ser reescrito como produtos de matriz. Assim, para $1 \leq i \leq n$, tem-se:

$$([R_{p_i^p}] Q)^T ([L_{p_i^r}] Q) = Q^T [R_{p_i^p}]^T [L_{p_i^r}] Q \quad (39)$$

Já o produto interno $\langle \hat{q}, (p_i^r q - qp_i^p) \rangle$, pode ser reescrito como um produto de matriz, como mostra a equação a seguir:

$$\hat{Q}^T \left([L_{p_i^r}] Q - [R_{p_i^p}] Q \right) = \hat{Q}^T \left(-[L_{p_i^r}] + [R_{p_i^p}] \right) Q \quad (40)$$

Logo, o somatório da Equação (36) é equivalente ao produto das matrizes:

$$\begin{aligned} E &= c - 2 \sum_{i=1}^n \left(Q^T [R_{p_i^p}]^T [L_{p_i^r}] Q \right) + 4 \sum_{i=1}^n \left(\hat{Q}^T \left(-[L_{p_i^r}] + [R_{p_i^p}] \right) Q \right) + 4n \hat{Q}^T \hat{Q} = \\ &c - 2Q^T \left(\sum_{i=1}^n \left([R_{p_i^p}]^T [L_{p_i^r}] \right) \right) Q + 4\hat{Q}^T \left(\sum_{i=1}^n \left(-[L_{p_i^r}] + [R_{p_i^p}] \right) \right) Q + 4n\hat{Q}^T \hat{Q} = \\ &c + 2Q^T W_1 Q + 4\hat{Q}^T W_2 Q + 4\hat{Q}^T W_3 \hat{Q} \end{aligned} \quad (41)$$

sendo, $W_1 = -\sum_{i=1}^n \left([R_{p_i^p}]^T [L_{p_i^r}] \right)$, $W_2 = \sum_{i=1}^n \left(-[L_{p_i^r}] + [R_{p_i^p}] \right)$ e $W_3 = n$.

Portanto, para encontrar Q e \hat{Q} o somatório deve ser minimizado com as seguintes restrições (WALKER *et al*, 1991):

$$Q^T Q = 1 \text{ e } \hat{Q}^T \hat{Q} = 0 \quad (42)$$

Para isso, o método dos Multiplicadores de Lagrange pode ser utilizado, fornecendo uma solução para determinar os mínimos do somatório sujeita as condições descritas na Equação (42), como mostra a equação a seguir:

$$\tilde{E} = c + 2Q^T W_1 Q + 4\hat{Q}^T W_2 Q + 4\hat{Q}^T W_3 \hat{Q} + \lambda_1 (Q^T Q - 1) + \lambda_2 \hat{Q}^T \hat{Q} \quad (43)$$

sendo λ_1 e λ_2 os Multiplicadores de Lagrange. Calculando as derivadas parciais em relação aos parâmetros do Dual Quaternions, obtém-se:

$$\frac{\partial \tilde{E}}{\partial Q} = 2(W_1 + W_1^T)Q + 4W_2^T \hat{Q} + 2\lambda_1 Q + \lambda_2 \hat{Q} = 0 \quad (44)$$

$$\frac{\partial \tilde{E}}{\partial \hat{Q}} = 4W_2Q + 8W_3\hat{Q} + \lambda_2Q = 0$$

Como $\begin{bmatrix} R_{p_i^p} \end{bmatrix}^T \begin{bmatrix} L_{p_i^r} \end{bmatrix}$ é simétrica então $W_1 + W_1^T = 2W_1$. Sendo $W_3 = n$, assim as duas equações acima são equivalentes ao sistema (YAN-BIN, 2020):

$$\begin{cases} 4W_1Q + 4W_2^T\hat{Q} + 2\lambda_1Q + \lambda_2\hat{Q} = 0 \\ 4W_2Q + 8n\hat{Q} + \lambda_2Q = 0 \\ Q^TQ = 1 \\ \hat{Q}^TQ = 0 \end{cases} \quad (45)$$

Para resolver esse sistema de equações, a segunda equação pode ser pré-multiplicada (pela esquerda) por Q^T , ou seja:

$$4Q^TW_2Q + 8nQ^T\hat{Q} + \lambda_2Q^TQ = 4Q^TW_2Q + 8\lambda_2Q^TQ = 0 \quad (46)$$

sendo,

$$\begin{aligned} -\begin{bmatrix} L_{p_i^r} \end{bmatrix} + \begin{bmatrix} R_{p_i^p} \end{bmatrix} &= -\begin{bmatrix} 0 & -p_x^r & -p_y^r & -p_z^r \\ p_x^r & 0 & -p_z^r & p_y^r \\ p_y^r & p_z^r & 0 & -p_x^r \\ p_z^r & -p_y^r & p_x^r & 0 \end{bmatrix} + \begin{bmatrix} 0 & -p_x^p & -p_y^p & -p_z^p \\ p_x^p & 0 & p_z^p & -p_y^p \\ p_y^p & -p_z^p & 0 & p_x^p \\ p_z^p & p_y^p & -p_x^p & 0 \end{bmatrix} = \\ & \begin{bmatrix} 0 & p_x^r - p_x^p & p_y^r - p_y^p & p_z^r - p_z^p \\ -p_x^r + p_x^p & 0 & p_z^r + p_z^p & -p_y^r - p_y^p \\ -p_y^r + p_y^p & -p_z^r - p_z^p & 0 & p_x^r + p_x^p \\ -p_z^r + p_z^p & p_y^r + p_y^p & -p_x^r - p_x^p & 0 \end{bmatrix} \end{aligned} \quad (47)$$

então,

$$Q^T \left(-\begin{bmatrix} L_{p_i^r} \end{bmatrix} + \begin{bmatrix} R_{p_i^p} \end{bmatrix} \right) Q = \quad (48)$$

$$[q_0 \quad q_1 \quad q_2 \quad q_3] \begin{bmatrix} 0 & p_x^r - p_x^p & p_y^r - p_y^p & p_z^r - p_z^p \\ -p_x^r + p_x^p & 0 & p_z^r + p_z^p & -p_y^r - p_y^p \\ -p_y^r + p_y^p & -p_z^r - p_z^p & 0 & p_x^r + p_x^p \\ -p_z^r + p_z^p & p_y^r + p_y^p & -p_x^r - p_x^p & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} =$$

logo,

$$Q^T W_2 Q = Q^T \left(\sum_{i=1}^n (-[L_{p_i^r}] + [R_{p_i^p}]) \right) Q = 0 \quad (49)$$

Substituindo na segunda equação do sistema representado na Equação (45) e tornando $\lambda_2 = 0$, chega-se a:

$$4W_2 Q + 8n\dot{Q} = 0 \quad (50)$$

ou seja,

$$\dot{Q} = -\frac{1}{2n} W_2 Q \quad (51)$$

Substituído o resultado mostrado na Equação (51) na primeira equação do sistema representado na Equação (45):

$$4W_1 Q - \frac{2}{n} W_2^T W_2 Q + 2\lambda_1 Q = 0 \quad (52)$$

ou seja,

$$\lambda_1 Q = -2W_1 Q + \frac{1}{n} W_2^T W_2 Q = A Q \quad (53)$$

sendo,

$$-2W_1 + \frac{1}{n}W_2^T W_2 = A \quad (54)$$

Assim, Q é um autovetor da matriz A e λ_1 é o autovalor correspondente. Reconsiderando o sistema mostrado na Equação (45), pré-multiplicando a primeira equação por $\frac{1}{2}Q^T$ e a segunda equação por $\frac{1}{2}\hat{Q}^T$, logo tem-se o sistema:

$$\begin{cases} 2Q^T W_1 Q = -2Q^T W_2^T \hat{Q} - \lambda_1 Q^T Q - \lambda_2 Q^T \hat{Q} = -2Q^T W_2^T \hat{Q} - \lambda_1 \\ 4\hat{Q}^T W_3 \hat{Q} = -2\hat{Q}^T W_2 Q - \frac{1}{2}\lambda_2 \hat{Q}^T Q = -2\hat{Q}^T W_2 Q \end{cases} \quad (55)$$

Substituído este sistema em E na Equação (35) chega-se a:

$$\begin{aligned} E &= c + 2Q^T W_1 Q + 4\hat{Q}^T W_2 Q + 4\hat{Q}^T W_3 \hat{Q} \\ &= c + 2Q^T W_1 Q + 4\hat{Q}^T W_2 Q - 2\hat{Q}^T W_2 Q = \\ c + 2Q^T W_1 Q + 2\hat{Q}^T W_2 Q &= c - 2Q^T W_2^T \hat{Q} - \lambda_1 + 2\hat{Q}^T W_2 Q = c - \lambda_1 \end{aligned} \quad (56)$$

Deste modo, o erro é minimizado se for selecionado o autovetor Q correspondente ao maior autovalor λ_1 . Depois de Q , agora é possível encontrar t substituindo na equação:

$$\hat{Q} = -\frac{1}{2n}W_2 Q \quad (57)$$

na equação,

$$\hat{Q} = \frac{1}{2}[R_q]^T \quad (58)$$

ou seja:

$$T = 2[R_q]^T \hat{Q} = \frac{-1}{n}[R_q]^T W_2 Q \quad (59)$$

Por fim, utilizando as Equações (54) e (59) a rotação e a translação são obtidas, respectivamente, utilizando pontos conjugados nos dois sistemas de coordenadas 3D.

2.5. ALGORITMO ICP

Besl e McKay (1992) propuseram o algoritmo ICP a partir do método de Horn (1987), ao qual utiliza Quaternions para representar as rotações entre os sistemas de coordenadas por meio de pares de pontos correspondentes. Neste método, a translação é determinada a partir dos deslocamentos após a rotação e da utilização dos centroides. Como o estabelecimento das correspondências se dá por pontos pseudo-conjugados, não é possível aplicar o método proposto por Horn diretamente, assim Besl e McKay (1992) propuseram aplicá-lo iterativamente. Os parâmetros de transformação são obtidos pelo algoritmo ICP através de uma função erro que minimiza o somatório da distância entre os pontos pertencentes a um sistema de coordenadas de Referência (R) e de Pesquisa (P). Basicamente, o algoritmo funciona da seguinte forma (BESL e MCKAY, 1992):

- 1) São definidos os valores de rotação e translação iniciais. No caso desses valores não serem definidos, os valores adotados devem ser nulos;
- 2) Para cada ponto da nuvem de pontos de Pesquisa (P) são localizados os vizinhos mais próximos na nuvem de pontos de Referência (R). Nessa etapa são estabelecidos os pares de pontos pseudo-conjugados. Para esse cálculo que localiza o vizinho mais próximo, deve ser considerada a distância métrica $dist$ entre um ponto individual $p \in P$ e a nuvem R denotada por (BESL e MCKAY, 1992):

$$dist(p, R) = \min_{q \in R} \|q - p\| \quad (60)$$

sendo, p representado pelos n pontos $\{p_i^p\}$ e R pelos m pontos $\{p_i^r\}$. Assim, é determinado o ponto mais próximo de p em R .

- 3) A partir do estabelecimento dos pontos pseudo-conjugados, os parâmetros de transformação são determinados após a parametrização da rotação e solucionados pela minimização da função energia mostrada na Equação (33). Na sequência, esses parâmetros de transformação entre as duas nuvens de pontos LiDAR 3D são aplicados à nuvem de pontos 3D de Pesquisa;

4) Por fim, é calculada a REMQ ponto a ponto entre as duas nuvens e comparado com o critério de convergência. Se o critério não for satisfeito, o processo é repetido novamente a partir do passo 2 entre a nuvem de pontos de Pesquisa e de Referência após a aplicação dos parâmetros calculados no passo 3. Caso contrário, o algoritmo encerra sua execução.

A principal vantagem do algoritmo ICP consiste no desempenho do algoritmo quando há um alinhamento inicial e alta sobreposição entre os pares de nuvens de pontos.

Para informação sobre o algoritmo NDT, utilizando neste trabalho para fins comparativos, ver Biber e Strasser (2003).

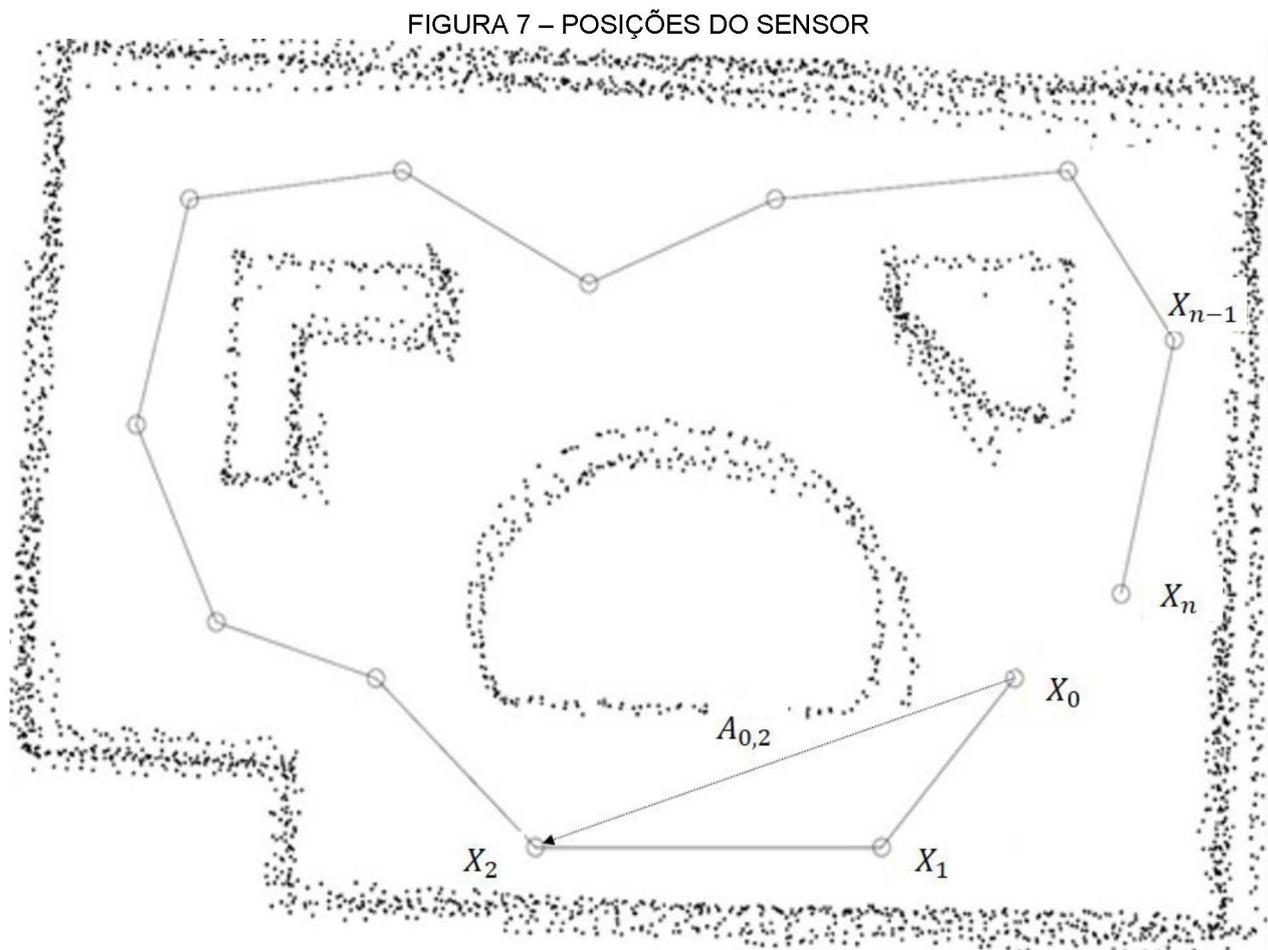
2.6. MÉTODO LUM

Lu e Milios (1997) propuseram uma formulação intuitiva para o problema de refinamento global dos parâmetros de transformação entre pares de nuvens de pontos 3D, conhecido na literatura como método LUM. Nessa formulação considera-se a realização de um percurso com diferentes posições do sensor, sendo a inicial denominada X_0 , como observa-se na Figura 7.

A partir da aquisição de dados de diferentes perspectivas ($X_0, X_1, X_2, \dots, X_{n-1} \dots X_n$) é possível definir uma série de relações entre as posições por meio de junções (arestas). Na estrutura de grafos do método LUM, os vértices representam os parâmetros de transformação entre pares de nuvens, enquanto que as arestas representam as relações entre estações do sensor consecutivas ($A_{0,1}, A_{0,2}, A_{0,n}, \dots A_{m,n}$). Sendo definidas essas relações, a principal tarefa do processo de refinamento global é estimar as posições dos sensores em um sistema de coordenadas globalmente consistente. A mensuração realizada entre cada estação do sensor ($A_{i,j}$), pode ser definida matematicamente, conforme a equação a seguir (LU e MILIOS, 1997).

$$F: A_{i,j} = X_i - X_j \quad (61)$$

sendo F o modelo funcional, $A_{i,j}$ o novo conjunto dos parâmetros de translação ajustados, X_i a Matriz Homogênea dos parâmetros de translação da pose X_i e X_j a Matriz Homogênea dos parâmetros de translação da pose X_j .



FONTE: Adaptado de LU E MILIOS (1997)

Devido aos erros presentes no processo, a observação $A_{i,j}$ é modelada de acordo com a equação a seguir (LU e MILIOS, 1997).

$$\bar{A}_{i,j} = A_{i,j} + \Delta A_{i,j} \quad (62)$$

sendo $\Delta A_{i,j}$ o erro gaussiano.

Para determinar a posição ideal do sensor, utiliza-se a estimação por máxima verossimilhança. Maximizar a probabilidade $P(A_{i,j}|\bar{A}_{i,j})$ é o equivalente a minimizar a distância de Mahalanobis, conforme mostra a equação a seguir (LU e MILIOS, 1997).

$$W(x) = \sum_{i,j} (A_{i,j} - \bar{A}_{i,j})^T (A_{i,j} - \bar{A}_{i,j}) \quad (63)$$

Na Equação 63 não está sendo considerado o termo de ponderação. A abordagem de Lu e Milios (1997) considera que cada vértice ($X_0, X_1, X_2, X_{n-1} \dots X_n$) da rede estará conectado com o sucessivo através das arestas ($A_{0,1}, A_{0,2}, A_{0,n}, \dots A_{m,n}$). Logo, a Equação (63) poderá ser escrita conforme mostra a equação a seguir (LU e MILIOS, 1997).

$$W(x) = \sum_{0 \leq i \leq j \leq n} (X_i - X_j - \bar{A}_{i,j})^T (X_i - X_j - \bar{A}_{i,j}) \quad (64)$$

Considerando uma matriz de incidência H , que contém as informações de cada transformação entre as diferentes estações do sensor, a relação entre os vértices é definida conforme a equação a seguir (LU e MILIOS, 1997).

$$A = HX \quad (65)$$

sendo A a relação matemática entre os vértices do grafo, H a matriz incidência e X a concatenação de X_0 a X_n .

Substituindo a Equação (65) na Equação (64), a distância de Mahalanobis pode ser escrita conforme Equação (66).

$$W = (HX - \bar{A})^T (HX - \bar{A}) \quad (66)$$

sendo \bar{A} a concatenação de todas as observações.

A solução (X) que minimiza a Equação (66) é apresentada na equação a seguir (LU e MILIOS, 1997).

$$X = (H^T H)^T (\bar{A} - HX) \quad (67)$$

O método de LUM é proposto para a resolução do problema da propagação de erros ao longo da trajetória do sensor utilizando os valores das translações obtidas na etapa do registro local entre os pares de nuvens de pontos LiDAR 3D. Em Borrmam *et al*, (2008) é proposta uma extensão do trabalho de Lu e Millios (1997) para lidar com a rotação a partir da sua linearização. Para redistribuir globalmente os erros do registro local ao longo da trajetória é calculada uma MC para cada posição no estabelecimento de correspondências entre primitivas pertencentes as nuvens de pontos LiDAR 3D. Assim, os parâmetros de transformação globalmente refinados são calculados pela minimização da distância de Mahalanobis dos pontos pseudo-conjugados.

A seguir serão apresentados os materiais e métodos empregados neste trabalho.

3. MATERIAIS E MÉTODOS

No capítulo anterior foi apresentado o sensor LiDAR utilizado nesta pesquisa, bem como algoritmos para a segmentação e extração de planos presentes nas nuvens de pontos LiDAR 3D obtidas com o sensor, além da álgebra de Dual Quaternions para a estimativa de parâmetros de transformação por pontos entre os pares de nuvens de pontos LiDAR 3D e o refinamento local e global pelo algoritmo ICP e método LUM, respectivamente. Neste capítulo serão abordados os materiais e o método utilizados para a realização da presente pesquisa.

3.1. MATERIAIS

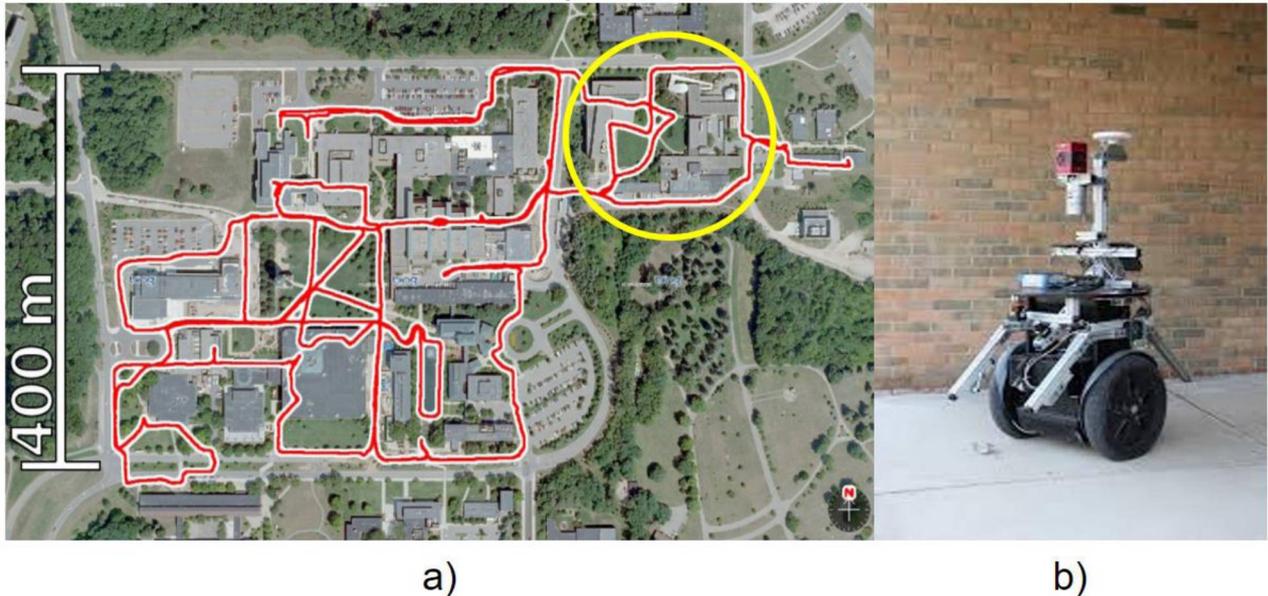
Os recursos bibliográficos e os dados empregados na realização do trabalho foram:

- i) Material bibliográfico disponível pela biblioteca de Tecnologia e Ciências Exatas da Universidade Federal do Paraná.
- ii) Conjunto de dados obtidos pelos pesquisadores Nicholas Carlevaris-Bianco, Arash K. Ushani e Ryan M. Eustice do Campus Norte da Universidade de Michigan (MICHIGAN, 2012) entre janeiro de 2012 e abril de 2013.

O conjunto de dados consiste em imagens omnidirecionais, LiDAR 3D, dados de posicionamento GNSS e propriedades proprioceptivas (sensores para odometria) obtidos por um robô Segway (ver Figura 8b). Segundo Michigan (2012), este conjunto de dados é composto por 27 sessões contendo 34,9 horas de processamento cobrindo uma trajetória de 147,4 km. As sessões exploram repetidamente o Campus da Universidade de Michigan em diferentes trajetórias e em diferentes momentos do dia nas quatro estações do ano. Isso permite que o conjunto de dados contenha várias situações complexas em um projeto de mapeamento tridimensional urbano, incluindo: obstáculos em movimento (por exemplo, pedestres, ciclistas e veículos), mudança de iluminação, ponto de vista variável, sazonalidade e mudanças climáticas, por exemplo, queda de folhas e neve, e, mudanças estruturais de longo prazo causadas por projetos de construção. A seguir, na Figura 8 é apresentada uma imagem contendo o croqui da área

levantada Figura 8a e do robô Segway Figura 8b utilizado para executar os levantamentos. Cabe ressaltar que o robô é controlado manualmente pelo usuário para definir a trajetória e desviar de obstáculos.

FIGURA 8 – (a) IMAGEM AÉREA DO CAMPUS NORTE DA UNIVERSIDADE DE MICHIGAM; (b) IMAGEM DO ROBÔ SEGWAY USADO PARA A OBTENÇÃO DOS DADOS.



FONTE: MICHIGAN (2012)

Para a realização deste trabalho foi utilizado apenas um recorte (círculo amarelo na Figura 8a de um dos conjuntos de dados disponibilizados. Esse conjunto está disponível no repositório datado em 17-03-2012, com aproximadamente 5,8 km de extensão sendo o trecho recortado de aproximadamente 580 metros.

No conjuntos de dados obtidos especificamente pelo sensor Velodyne, estão disponíveis todas as revoluções obtidas pelo sensor durante o trajeto do dia 17-03-2012, chamadas de *velodyne_hits*. Entretanto, devido a capacidade de processamento disponível e a superabundância de pontos, foram usadas apenas as leituras associadas aos dados de odometria visual disponíveis em um arquivo chamado de *vel_sync*.

Os dados de odometria visual foram estimados usando um EKF (*Extended Kalman Filter*) que usa um processo diferencial de integração das medições dos encoders das rodas do robô Segway e de um giroscópio óptico (FOG - *Fiber Optic Gyro*) que retorna observações sobre alterações na direção que a plataforma sofre. Assim, as medidas são

atualizadas a partir das entradas da IMU que fornece as taxas angulares de rotação, inclinação e translação da plataforma.

3.1.1. Recursos de Hardware

i) 01 sensor Velodyne HDL-32E: O HDL-32E possui 32 lasers acoplados em uma plataforma giratória que gira em torno de seu eixo vertical fornecendo uma visão de 360° do campo de visão azimutal (VELODYNE, 2012).

ii) Para a obtenção dos dados de odometria como critério comparativo, um Microstrain 3DM-GX3-45 IMU contendo acelerômetros, giroscópios e magnetômetros tridimensionais; e, um receptor GNSS integrado; Um giroscópio óptico KVH DSP-3000 *single-axis* FOG usado para fornecer as medidas do azimute.

iii) 01 laptop DELL *INSPIRION* N4050; Processador Intel Core i7; Sistema operacional windows 10© Microsoft Corporation devidamente licenciado.

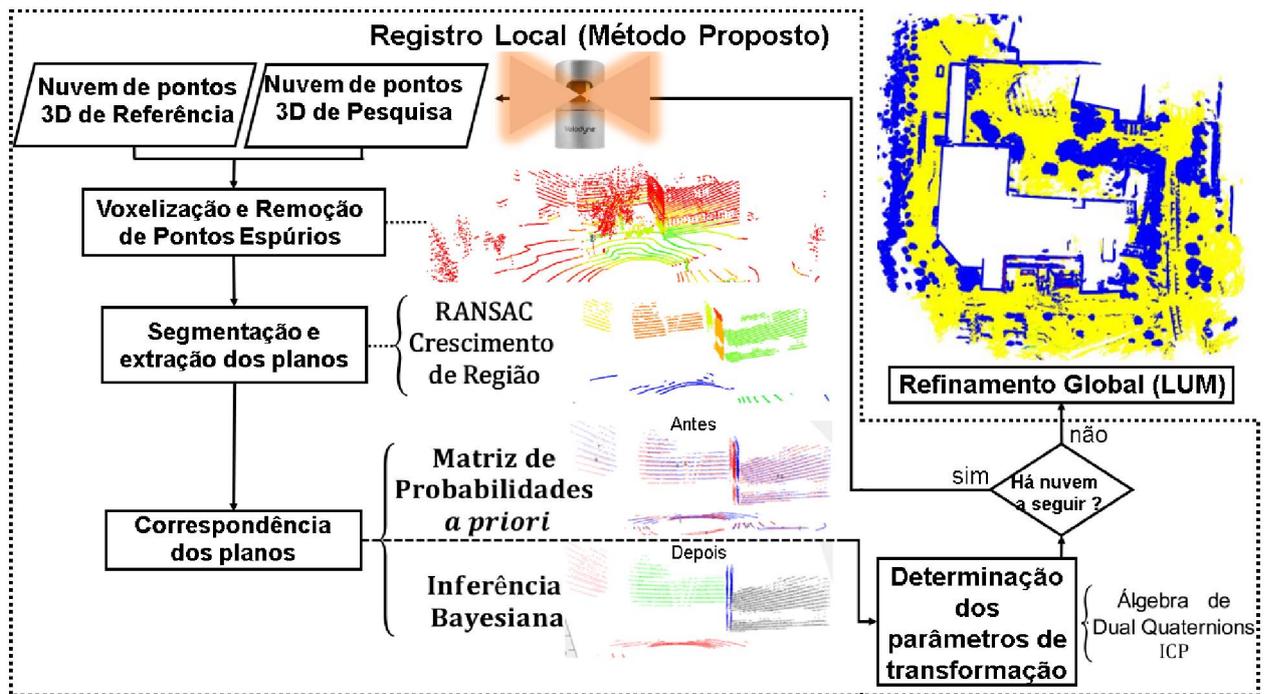
3.1.2. Recursos de Software e Bibliotecas Computacionais *Open-Source*

- a) Linux Ubuntu 14.04 LTS. Open Source, licença GPL (*General Public License*);
- b) Linguagem C++ em Ambiente CodeBlock, open-source licença GPL;
- c) Linguagem python para a extração dos arquivos “.csv” contendo as nuvens de pontos LiDAR 3D;
- d) Biblioteca Armadillo. Biblioteca de álgebra linear para a linguagem C ++. Fornece sintaxe e funcionalidades deliberadamente semelhantes ao Matlab (mais detalhes ver SANDERSON E CURTIN, 2016);
- e) Biblioteca PCL (*Point Cloud Library*). Biblioteca desenvolvida em C++ (RUSU, 2011), *open-source*, distribuído sob licença BSD (*Berkeley Software Distribution*). Esta biblioteca inclui algoritmos para processamento de nuvens de pontos n-dimensionais. Versão 1.7.2.;

3.2. MÉTODO

O propósito central dessa tese é o desenvolvimento de um método para mapeamento tridimensional em ambientes urbanos utilizando nuvens de pontos LiDAR 3D obtidas com o sensor Velodyne. A Figura 9 ilustra um fluxograma dos processos empregados para tal.

FIGURA 9 - FLUXOGRAMA DO MÉTODO PROPOSTO PARA MAPEAMENTO TRIDIMENSIONAL EM AMBIENTES URBANOS.



FONTE: O autor (2020)

Como pode ser visto na Figura 9, o método proposto pode ser dividido em 2 etapas principais:

- 1) Registro Local: Primeiramente é aplicado os processos de voxelização e remoção de pontos espúrios às nuvens de pontos LiDAR 3D de Pesquisa e Referência. Em seguida, a posterior segmentação e extração dos planos por meio dos algoritmos RANSAC e crescimento de região, respectivamente. Por fim, o registro local dos pares de nuvens de pontos LiDAR 3D com os métodos propostos para a correspondência automática e determinação dos parâmetros de transformação utilizando planos.

2) Refinamento Global dos parâmetros de transformação entre os pares de nuvens de pontos LiDAR 3D previamente registradas utilizando o método LUM.

A seguir será apresentada a descrição das etapas supracitadas e de seus processos inerentes.

3.2.1. Registro Local

Esta etapa tem por objetivo obter os parâmetros de transformação entre os pares de nuvens de pontos LiDAR 3D de Pesquisa e de Referência. O registro local, neste trabalho, é determinado pela melhor escolha dentre os resultados obtidos pelos registros grosseiro, grosseiro-para-fino e fino, dependendo de uma análise do valor da REMQ ponto a plano após a aplicação às nuvens de pontos LiDAR 3D de Pesquisa. Esta etapa pode ser dividida nos processos mostrados na Figura 9 e que são descritos a seguir .

3.2.1.1. Voxelização e Remoção de Pontos Espúrios

As nuvens de pontos LiDAR 3D obtidas pelo sensor LiDAR Velodyne e disponíveis no banco de dados hospedado pelo site: <http://robots.engin.umich.edu/nclt/>, estão originalmente em formato binário. Assim, primeiramente foi aplicado o *script velodyne_sync* (linguagem python), também disponível no endereço citado, para transformar as nuvens de pontos LiDAR 3D em um formato “.csv” (*comma separated values*), compatível com os formatos de entrada pelos algoritmos de leitura de dados implementados na PCL (RUSU, 2011). O resultado desse processo são arquivos contendo as coordenadas tridimensionais de cada revolução do Velodyne, a intensidade e o índice do LASER que gerou cada leitura: ($X, Y, Z, intensidade, laser\ id$).

Cada nuvem de pontos LiDAR 3D está referenciada ao sistema de coordenadas do sensor, assim, a princípio os centroides das nuvens de pontos LiDAR 3D de Pesquisa e de Referência são calculados, como mostra a Equação (68), e, em seguida, as nuvens são todas referenciadas aos seus centroides como a origem do sistema de coordenadas.

Este processo é necessário para o estabelecimento da estrutura *kd-tree* (FRIEDMAN *et al*, 1977) e para um melhor condicionamento do sistema de equações.

$$C_p = \frac{\sum_{i=1}^m p_i^p}{m} \text{ e } C_r = \frac{\sum_{i=1}^n p_i^r}{n} \quad (68)$$

sendo p_i^p e p_i^r os pontos pertencentes a nuvem de pontos LiDAR 3D de Pesquisa e Referência, respectivamente, e m, n suas cardinalidades.

Devido a alta frequência de obtenção dos pontos pelo sensor LiDAR utilizado nesta pesquisa, é realizado um tratamento inicial à nuvem de pontos LiDAR 3D com um processo de voxelização e remoção de pontos espúrios. Primeiramente é realizado um processo de reamostragem por voxelização com o algoritmo *Voxel_grid* (RUSU, 2011) com a definição do tamanho da grade (*setGridSize*). Assim, é calculado o centroide dos pontos dentro de cada grade, com o uso da Equação (68), e o ponto médio é usado para compor a nuvem de pontos LiDAR 3D reamostrada. Posteriormente, é realizada a remoção de pontos espúrios pelo algoritmo *Statistical Outlier Removal* (RUSU, 2011) com o intuito de filtrar pontos indesejados, resultados de leituras errôneas do sensor LASER e, assim, diminuindo o custo computacional do algoritmo de segmentação. Como parâmetros, foram definidos:

- i) Estimativa da Distância Média (*setMeanK*): Define o número de pontos a serem usados na estimativa da distância média.
- ii) Limite do Multiplicador de Desvio Padrão (*setStddevMulThresh*): Todos os pontos fora do intervalo: $(\mu \mp \sigma * \text{setStddevMulThresh})$ serão considerados pontos espúrios, sendo μ a média estimada e σ o desvio padrão.

3.2.1.2. Segmentação e Extração dos Planos

Para a extração dos planos é necessário realizar uma segmentação com a finalidade de remover pontos que não pertençam a regiões planares. Assim, nesta pesquisa será utilizado o algoritmo RANSAC, na qual para cada nuvem de pontos LiDAR 3D obtida, os seguintes passos são realizados (RUSU, 2011):

- 1) São selecionados os ξ vizinhos mais próximos de cada ponto pertencente a nuvem de pontos LiDAR 3D.
- 2) É estabelecida uma estrutura *kd-tree* para agilizar as buscas por vizinhos mais próximos a partir da quantidade definida no passo 1.
- 3) O algoritmo RANSAC é aplicado utilizando a estrutura *kd-tree*, com a definição do número ζ máximo de iterações (número de pontos escolhidos aleatoriamente) e com uma tolerância correspondente a distância máxima ζ que o ponto pode ter do plano calculado para ser considerado *inlier*.

O algoritmo RANSAC retorna para cada nuvem de pontos LiDAR 3D um arquivo com as seguintes informações: $(X, Y, Z, intensidade, n_x, n_y, n_z)$, sendo n_x, n_y e n_z componentes do vetor normal. Este vetor normal é definido a partir da vizinhança de cada ponto. Assim, a partir dessas observações o algoritmo Crescimento de Regiões (RABBANI *et al*, 2006) é aplicado com o objetivo de extrair os planos com as menores curvaturas possíveis. Como nem sempre os planos extraídos formarão um Espaço Canônico Tridimensional, nesta pesquisa o algoritmo de crescimento de região é aplicado com parâmetros menos rígidos para gerar dois resultados: I - Planos de referência com um valor de curvatura inferior a um determinado limiar τ (ver Equação (8)); II - O complemento desse conjunto, aqui chamado de *clusters*.

Os parâmetros de configuração do algoritmo de crescimento de região foram definidos da seguinte forma:

- i) Cardinalidade do conjunto (*setMinClusterSize* e *setMaxClusterSize*): Define o número mínimo e máximo de pontos que um determinado *cluster* candidato deve conter.
- ii) Quantidade de vizinhos (*setNumberOfNeighbours*): Define a quantidade de pontos vizinhos para cada semente que será analisada.
- iii) Tolerância angular (*setSmoothnessThreshold*): Critério que leva em consideração o vetor normal associado a cada ponto para avaliar se o ponto deve ser adicionado ao *cluster* investigado.
- iv) Tolerância residual da curvatura (*setCurvatureThreshold*): Caso a distância perpendicular do ponto ao plano seja superior a este parâmetro, o mesmo não é adicionado ao *cluster* investigado.

Como forma de refinar os resultados do algoritmo, também foram aplicados critérios de distância ponto a plano e cardinalidade dos planos. Assim, é calculada a distância ponto a plano entre todos os pontos e os planos definidos por eles, e, se o ponto estiver a uma distância menor que um limiar pré-estabelecido (*setMinDist*), é considerado *inlier*, do contrário, descartado. Então, é avaliada a cardinalidade dos planos remanescentes, ou seja, para não ser desconsiderado deve apresentar um valor mínimo (*setMinPoints*) de pontos após o processo de filtragem previamente realizado. Caso o plano seja rejeitado em algum desses critérios, passará a integrar o conjunto dos *clusters*.

Como resultado um conjunto de pontos é obtido com as coordenadas tridimensionais X , Y e Z , e, com as informações de intensidade substituídas por um índice identificando a que plano cada ponto pertence.

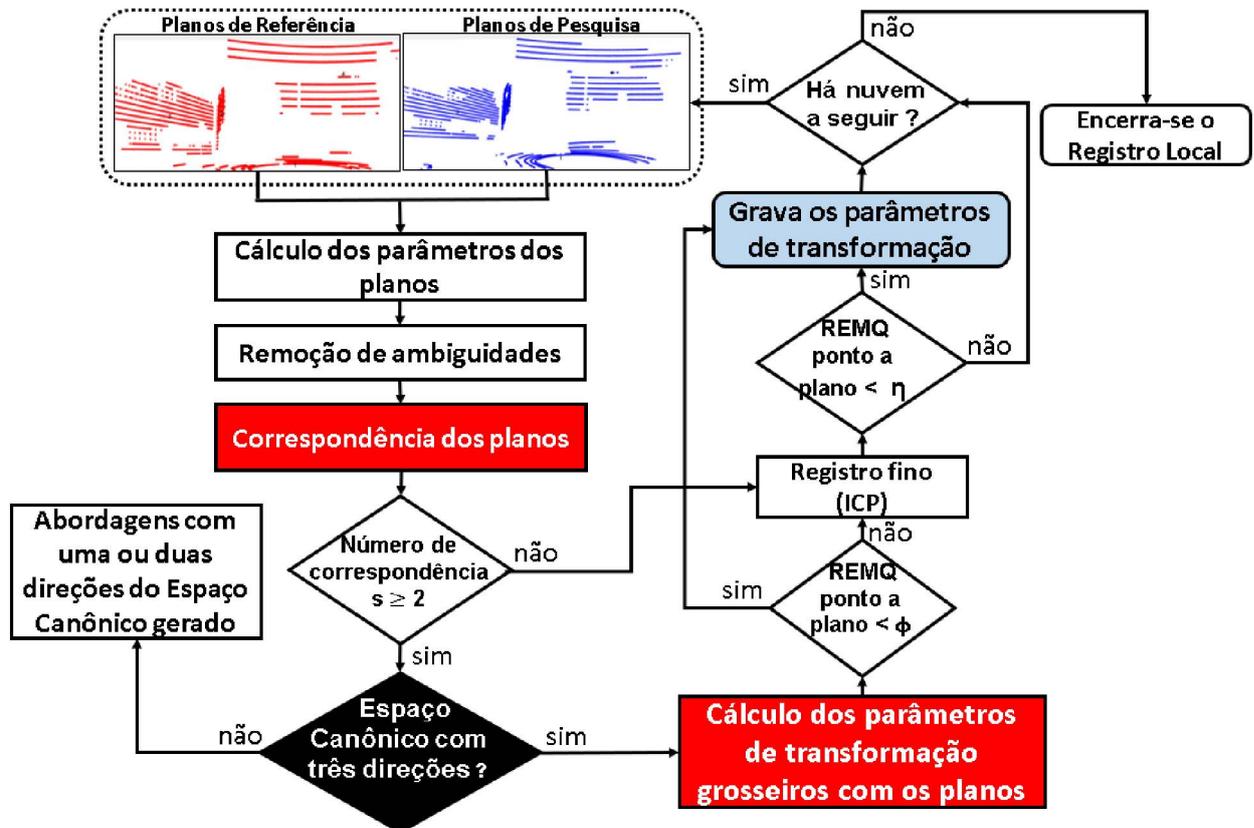
3.2.1.3. Estabelecimento de Correspondências entre os Planos e o Cálculo dos Parâmetros de Transformação

Primeiramente, a partir do processo de extração dos planos são calculados os parâmetros desses planos presentes em ambas as nuvens de pontos LiDAR 3D de Referência e de Pesquisa como mostra a Seção 2.3. Em seguida, as ambiguidades causadas pelo processo de segmentação são removidas, onde os parâmetros dos planos divididos/separados sejam recalculados a partir de seus pontos reunidos. Posteriormente, é realizada a correspondência entre os planos presentes nos pares de nuvens de pontos LiDAR 3D. Se não houver alguma correspondência entre os planos, os parâmetros de transformação entre as nuvens de pontos LiDAR 3D são obtidos diretamente pelo registro fino com o algoritmo ICP. Caso o número de correspondências seja maior ou igual a dois, então é avaliado o Espaço Canônico gerado pelos vetores normais dos planos entre os pares de nuvens de pontos LiDAR 3D. Dependendo do Espaço Canônico gerado, não é possível estimar todos os parâmetros de transformação utilizando somente os planos. Por isso, nessa pesquisa é apresentado um método híbrido que adiciona pontos para a estimativa dos parâmetros de transformação ótimos.

Caso os vetores normais dos planos correspondidos gerem um Espaço Canônico tridimensional, os parâmetros de transformação entre os pares de nuvens de pontos LiDAR 3D são estimados utilizando uma solução inédita com a álgebra de Dual

Quaternions. Na Figura 10 é mostrado um fluxograma, onde as caixas em vermelho referem-se as principais contribuições deste trabalho. Esse destaque também ocorre para os casos mostrados nas Figura 11 e 12.

FIGURA 10 - FLUXOGRAMA DO MÉTODO DE REGISTRO LOCAL PROPOSTO PARA O CASO DE UM ESPAÇO CANÔNICO TRIDIMENSIONAL TER SIDO GERADO PELOS PLANOS OBSERVADOS.



FONTE: O autor (2020)

Para aumentar a qualidade da definição dos parâmetros dos planos, é feita uma filtragem a partir do cálculo da distância de cada ponto ao plano. Pontos que apresentam uma distância maior que determinado limiar (*setMaxDist*) são descartados e os parâmetros dos planos são recalculados.

O algoritmo usado neste trabalho para a segmentação e extração dos planos pode apresentar resultados inconsistentes onde planos aparecem divididos, ou seja, ambiguidades entre dois ou mais planos que representam a mesma superfície. Lidar com esse problema é de suma importância para assegurar uma correspondência ótima entre os planos. Assim, cada plano é comparado com todos os outros da mesma nuvem de

pontos LiDAR 3D e aqueles cujo centroide estiver a uma distância ortogonal de seu plano vizinho menor que um limiar ($setMaxDist$), e, a uma distância entre seus centroides menor que ($setMaxDistCent$), são considerados ambíguos e, por consequência, unificados com o posterior recálculo dos seus parâmetros.

Como pode ser visto na Figura 10, se a REMQ ponto a plano encontrada for menor que um limiar ϕ pré-estabelecido, o processo de refinamento dos parâmetros de transformação é desprezado, reduzindo o custo computacional. Do contrário, é feito o refinamento com o algoritmo ICP, obtendo os parâmetros de transformação grosseiro-para-fino da etapa de registro local proposta nesta pesquisa.

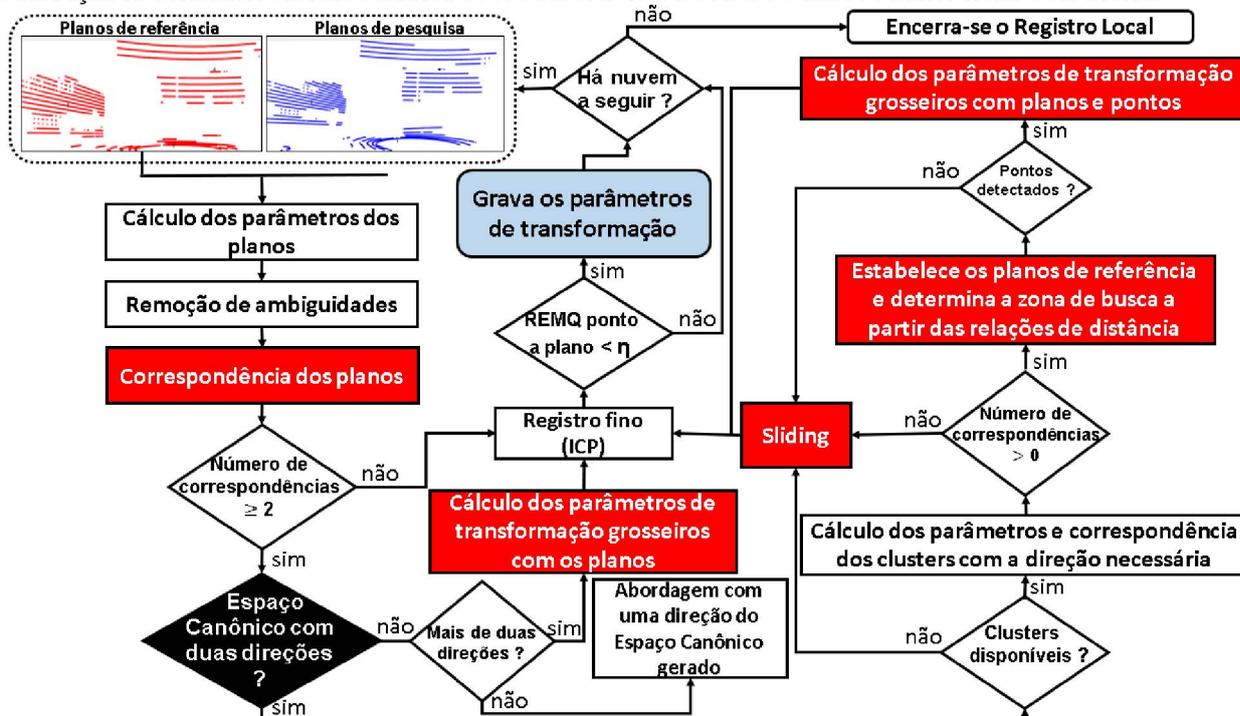
Como pode ser visto na Figura 10, para avaliar os resultados do registro antes da aplicação dos parâmetros de transformação à nuvem de pontos LiDAR 3D de Pesquisa, o valor da REMQ ponto a plano é comparado a um limiar η pré-estabelecido. Caso o erro seja menor que o limiar, os parâmetros de transformação são salvos, a nuvem de pontos LiDAR 3D de Pesquisa passa a ser considerada de Referência e a nuvem subsequente é adicionada como sendo a de Pesquisa. Do contrário a nuvem de pontos LiDAR 3D na sequência é atribuída como de Pesquisa e é feita uma nova tentativa para a realização do registro.

Com a definição do Espaço Canônico bidimensional, há uma impossibilidade da estimativa de todos os componentes da translação. A Figura 11 mostra um fluxograma contendo os processos utilizados nesta pesquisa para este caso. Como pode ser visto na Figura 11, nesta pesquisa é proposto o uso dos *clusters* para a diminuição do espaço de busca por correspondências entre pontos nas nuvens de pontos LiDAR 3D de Pesquisa e Referência.

Para auxiliar no processo, é feito o cálculo dos parâmetros dos *clusters* (vetor normal e distância à origem) bem como do centroide, deixando apenas aqueles com a direção deficiente no Espaço Canônico definido, e, a posterior correspondência desses *clusters* (algoritmo de correspondência de planos adaptado) de modo a garantir uma atribuição ótima entre os pontos. Em seguida, a partir de relações geométricas para com os planos de referência, são definidas duas regiões de interesse em cada *cluster* na nuvem de pontos LiDAR 3D de Pesquisa e a intersecção dessas duas regiões é usada para definir um ponto virtual pseudo-conjugado ao centroide do *cluster* correspondente

na nuvem LiDAR 3D de Referência. A partir da correspondência dos pontos, um cálculo direto utilizando a álgebra de Dual Quaternions é realizado para complementar o vetor de translação.

FIGURA 11 - FLUXOGRAMA DO MÉTODO DE REGISTRO LOCAL PROPOSTO PARA O CASO DE UM ESPAÇO CANÔNICO BIDIMENSIONAL TER SIDO GERADO PELOS PLANOS OBSERVADOS.



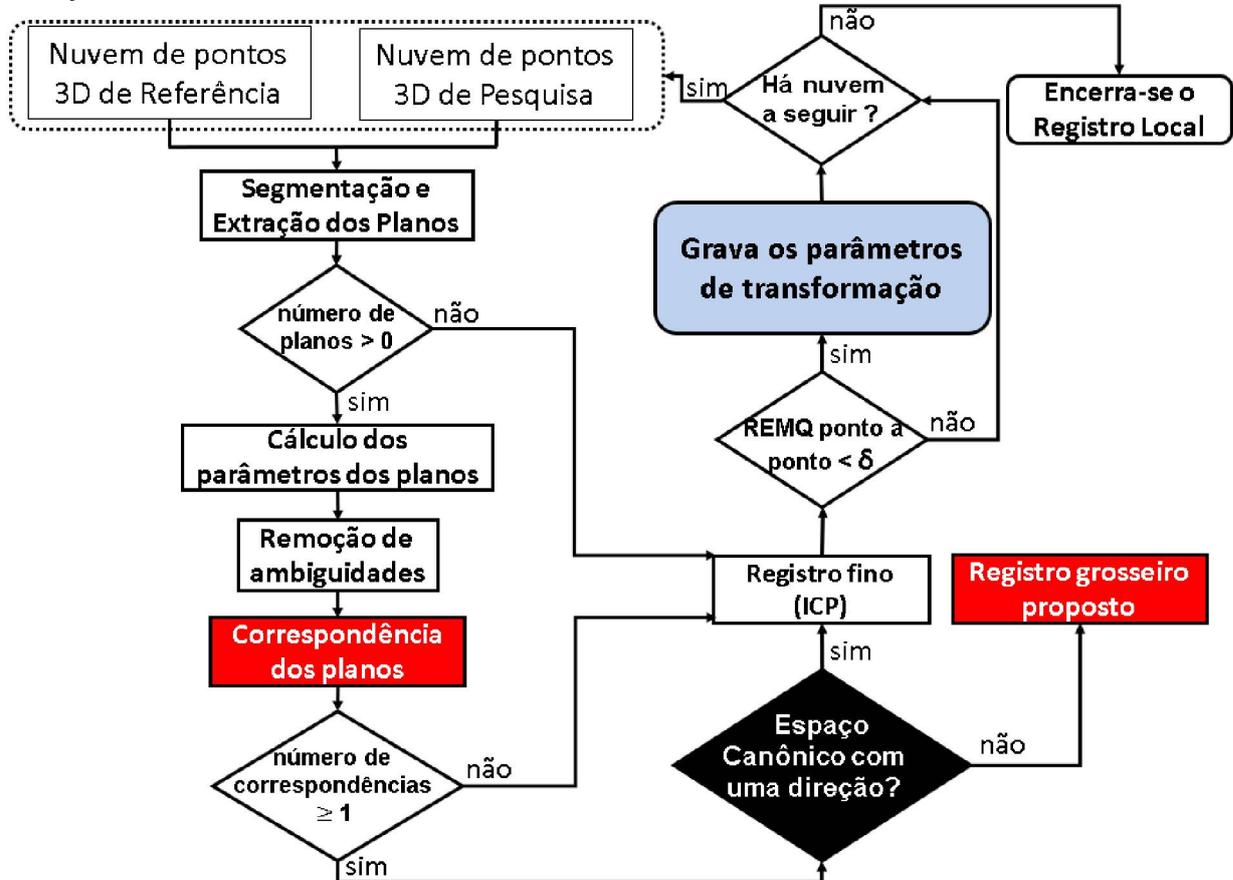
FONTE: O autor (2020)

Diante da ausência de *clusters* ou de sua inutilidade, nesta pesquisa é proposto um método de aproximação chamado de *Sliding*. Nesse método, os parâmetros de transformação parciais (parâmetros de translação obtidos com o espaço canônico disponível) são aplicados à nuvem de pontos LiDAR 3D de Pesquisa e um plano chave escolhido acima do terreno é deslizado sobre seu correspondente na nuvem de pontos LiDAR 3D de Referência visando a maior sobreposição possível entre eles. Por fim, usando o método de *clusters* e o de *Sliding* os parâmetros de transformação do registro grosseiro são posteriormente refinados usando o algoritmo ICP.

Na Figura 12 é mostrado um fluxograma contendo os processos utilizados nesta pesquisa quando nenhuma ou apenas uma direção do Espaço Canônico gerado pelos planos observados for detectada. Neste caso, não há graus de liberdade suficientes para a solução do sistema de equações, assim, os parâmetros de transformação entre as

nuvens de pontos LiDAR 3D de Referência e de Pesquisa são obtidos com o algoritmo ICP (registro fino).

FIGURA 12 – FLUXOGRAMA DO MÉTODO DE REGISTRO LOCAL PROPOSTO PARA O CASO DE UM ESPAÇO CANÔNICO UNIDIMENSIONAL OU INEXISTENTE SENDO GERADO PELOS PLANOS.



Como pode ser visto na Figura 12, um limiar δ foi estabelecido para avaliar os valores da REMQ ponto a ponto após a aplicação dos parâmetros de transformação às nuvens de pontos LiDAR 3D de Pesquisa. Caso o critério seja atendido os parâmetros são armazenados, do contrário, a nuvem de Pesquisa subsequente é adicionada e uma nova tentativa de registro é realizada.

3.2.1.3.1. Método de Correspondência dos Planos

O método de correspondência de planos proposto nesta pesquisa aplica a Inferência Bayesiana com o objetivo de atualizar a Matriz de Probabilidades *a priori* em uma Estimativa Máxima *a posteriori*. A Matriz de Probabilidades caracteriza relações geométricas entre os pares de planos como: direções dos vetores normais, distâncias entre os centroides e cardinalidade; e estima uma probabilidade *a priori* da correspondência entre cada par de planos candidatos. Em um segundo momento, cada probabilidade é atualizada em uma Estimativa Máxima *a posteriori* por Inferência Bayesiana considerando a configuração *a priori* e características geométricas locais dos planos com relação a seus vizinhos em cada nuvem de pontos LiDAR 3D.

Em um ambiente urbano, além da complexidade e pluralidade das feições disponíveis, o algoritmo de segmentação possui um certo limite de atuação, o que pode levar a atribuições incorretas de planos como correspondentes se um método subótimo for utilizado. Uma correspondência incorreta entre planos leva a uma estimativa insatisfatória dos parâmetros de transformação e conseqüentemente a um registro local incorreto. Portanto, a etapa de atualização da Matriz de Probabilidades é fundamental para eliminar falsos positivos na etapa de correspondências entre planos.

Cada linha da Matriz de Probabilidades mostrada na Equação (69) representa um plano na nuvem LiDAR 3D de Referência e cada coluna um plano na nuvem LiDAR 3D de Pesquisa. Em cada entrada, é determinada uma probabilidade (variando entre 0 e 1) dos planos serem correspondentes. Para a determinação dessas probabilidades pressupõe-se que os planos sendo correspondidos estão presentes em nuvens de pontos LiDAR 3D subsequentes e, portanto, estão próximas entre si. Isso implica que relações angulares e de distância entre pares subsequentes são proporcionalmente afetadas conforme o intervalo de obtenção pelo sensor. Para quantificar essa similaridade dos planos, 3 funções de probabilidades são propostas: Inclinação (Inp); Distância entre os centroides (Dtp) e Cardinalidade (Np).

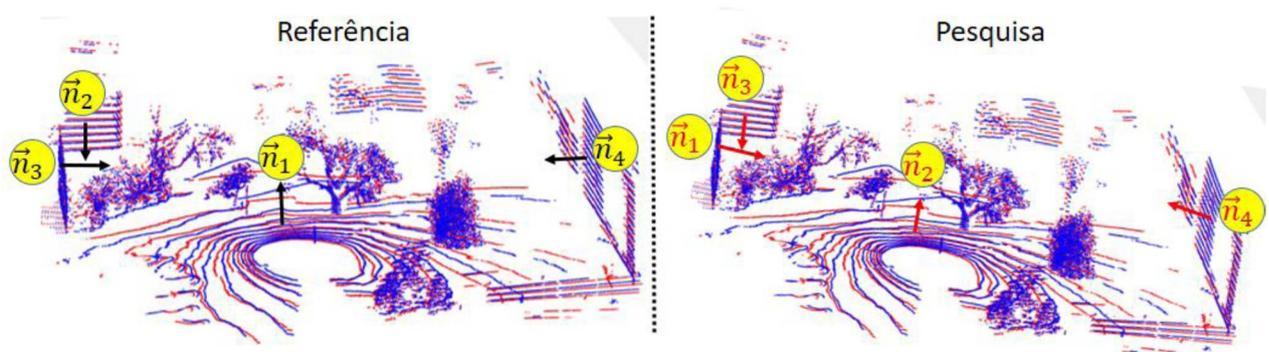
$$P = \begin{bmatrix} p(i, j) & \cdots & p(i, m) \\ \vdots & \ddots & \vdots \\ p(n, j) & \cdots & p(n, m) \end{bmatrix} \quad (69)$$

Na Equação (69) m e n representam o número de planos na nuvem de pontos LiDAR 3D de Pesquisa e de Referência, respectivamente. A seguir, uma descrição de cada função de Probabilidade que foi utilizada para compor a Matriz de Probabilidades a priori P .

3.2.1.3.1.1. Função de Probabilidade - Inclinação

Cada plano extraído na nuvem de pontos LiDAR 3D de Referência é comparado com todos os planos da nuvem de pontos LiDAR 3D de Pesquisa por meio dos seus vetores normais usando o produto escalar, o que resulta em uma probabilidade associada. Devido ao alcance moderado do sensor utilizado (~100 m) e o ambiente estudado (ambientes urbanos) o número de planos extraídos geralmente não chega a uma dezena, então o custo computacional desse processo é baixo. Na Figura 13, é mostrada uma imagem de duas nuvens de pontos LiDAR 3D vizinhas (exageradamente transladadas entre si para fins de visualização) com destaque aos vetores normais dos planos detectados pelo algoritmo Crescimento de Região.

FIGURA 13 - PAR DE NUVENS DE PONTOS LiDAR 3D COM DESTAQUE AOS VETORES NORMAIS DOS PLANOS DETECTADOS.



FONTE: O autor (2020)

Considerando dois planos quaisquer e seus vetores normais, sabe-se que quanto mais paralelos eles forem maior será a projeção escalar de um vetor no outro resultando no valor um (IEZZI, 2006), como mostra a Equação (70). Portanto, para cada vetor normal

do conjunto de planos de referência é estimado o produto escalar com todos os vetores normais do conjunto de planos da nuvem de pontos LiDAR 3D de Pesquisa por meio da Equação (70).

$$In(i, j) = \langle \vec{n}_i, \vec{n}_j \rangle \quad (70)$$

Como a Matriz de Probabilidade *a priori* P deve possuir valores dentro do intervalo entre 0 e 1, foi utilizado um sistema de ponderação para os valores das funções de probabilidades, como mostra a Equação (71). O valor adotado para o peso na função de inclinação dos vetores normais foi de 60% para os planos acima do terreno e 30% pertencentes ao terreno. A escolha desses valores é empírica e leva em conta o modo sequencial de obtenção das nuvens de pontos LiDAR 3D com um curto espaço temporal entre as varreduras, assim, os planos homólogos tendem a apresentar valores das componentes direcionais de seus vetores normais próximos entre si. Quanto aos planos do terreno, há uma homogeneidade de suas direções, assim, há pouca variabilidade de seus valores.

$$Inp(i, j) = \omega_1 * In(i, j) \quad (71)$$

sendo,

$$\begin{cases} \omega_1 = 0,6 & \text{planos acima do terreno;} \\ \omega_1 = 0,3 & \text{planos pertencentes ao terreno;} \end{cases}$$

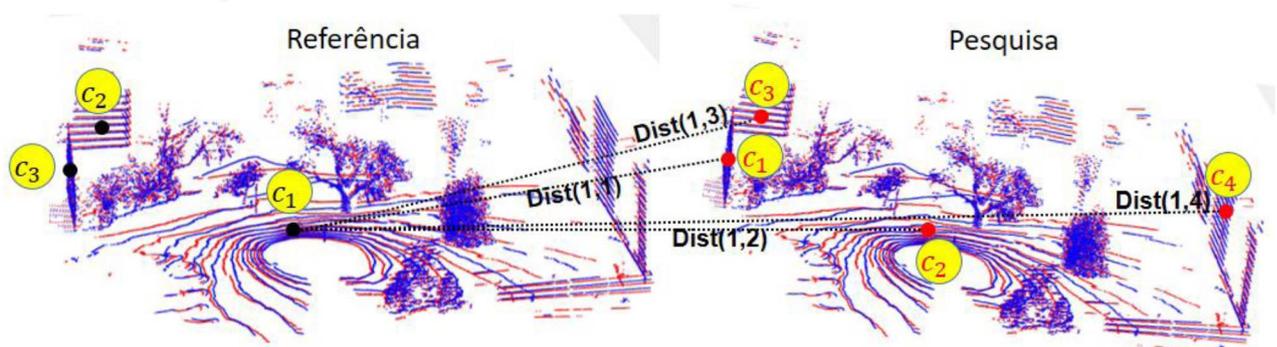
3.2.1.3.1.2. Função de Probabilidade - Distância entre os Centroides

A função de probabilidade Dtp considera a distância euclidiana dos centroides dos planos candidatos. Para cada plano na nuvem de pontos LiDAR 3D de Referência é determinado seu centroide e estimada a distância para todos os centroides pertencentes à nuvem de pontos LiDAR 3D de Pesquisa, como mostra a Equação (72).

$$Dist(i,j) = \sqrt{(c_{x_i} - c_{x_j})^2 + (c_{y_i} - c_{y_j})^2 + (c_{z_i} - c_{z_j})^2} \quad (72)$$

Para fins de ilustração, a Figura 14 representa as distâncias euclidianas entre o centroide do plano de referência c_1 com todos os centroides dos planos detectados na nuvem de pontos LiDAR 3D de Pesquisa c_j .

FIGURA 14 - PAR DE NUVENS DE PONTOS LIDAR 3D E OS CENTROIDES DOS PLANOS EXTRAÍDOS, COM DESTAQUE AOS VETORES DE DISTÂNCIA ENTRE c_1 E OS CENTROIDES c_j DOS PLANOS NA NUVEM DE PESQUISA.



FONTE: O autor (2020)

Pressupõe-se que planos próximos nas nuvens de pontos LiDAR 3D subsequentes tendem a serem correspondentes, afinal o sensor LiDAR utilizado nesta pesquisa foi transportado em uma plataforma em baixa velocidade. Assim, quanto menor a distância entre eles, maior a probabilidade associada. Mas, para quantificar essa probabilidade é necessário relacionar todas as distâncias encontradas entre o centroide do plano de Referência e os centroides dos planos de Pesquisa. A Equação (73) apresenta o cálculo da média das distâncias euclidianas entre os centroides de um plano (i) de Referência com os de Pesquisa (j).

$$Dtm(i,j) = \frac{\sum_{j=1}^m Dist(i,j)}{m} \quad (73)$$

A contribuição dessa função para a Matriz P será dada pela Equação (74) descrita a seguir. Essa relação visa atribuir probabilidades maiores para planos pertencentes a

nuvens de pontos LiDAR 3D subsequentes com menores distâncias entre seus centroides.

$$Dt(i, j) = \left(\frac{Dtm(i, j) - Dist(i, j)}{Dtm(i, j)} \right) \quad (74)$$

A Equação (75) apresenta o cálculo das probabilidades ponderadas da função de probabilidade das distâncias entre os centroides dos planos.

$$Dtp(i, j) = \omega_2 * Dt(i, j) \quad (75)$$

sendo,

$$\begin{cases} \omega_1 = 0,3 & - \text{planos acima do terreno;} \\ \omega_1 = 0,6 & - \text{planos pertencentes ao terreno;} \end{cases}$$

Percebe-se uma inversão dos valores dos pesos com relação a função de probabilidade anterior, sendo que neste caso, valoriza-se mais a proximidade dos planos pertencentes ao terreno. Esses valores também são empíricos e foram escolhidos com base na geometria das linhas de varredura pertencentes ao terreno aos quais apresentam excentricidades menores de acordo com a distância do sensor ao objeto. Portanto, conforme a proximidade dos centroides e a distância média para os outros centroides, maior será o valor retornado da função de probabilidade Dtp .

Nesta pesquisa foi adotado um parâmetro de distância mínima de ($setMinDistCor$) metros, assim, pares de planos com uma distância maior que este valor recebem uma probabilidade ponderada nula. Isso se faz necessário para manter o valor da média das distâncias o menor possível e conseqüentemente aumentar a discretização das probabilidades entre si.

3.2.1.3.1.3. Função de Probabilidade – Cardinalidade dos Planos

Outro critério considerado nesta pesquisa é a relação do número de pontos entre os planos das nuvens LiDAR 3D de Pesquisa e de Referência. Assim, conforme o

resultado da razão entre o número de pontos dos planos for próximo de um, maior a contribuição para a Matriz de Probabilidades P naquela posição. A Equação (76) representa essa relação.

$$N(i, j) = \frac{m}{n} \quad (76)$$

Para a ponderação dessa função de probabilidade, foi utilizada a Equação (77).

$$Np(i, j) = \omega_3 * N(i, j) \quad (77)$$

sendo, $\omega_3 = 0,1$ para todos os pares (valor determinado empiricamente).

Esse critério é baseado na geometria dos planos correspondentes, provavelmente similares entre si nas nuvens de pontos LiDAR 3D subsequentes, e faz-se necessário para complementar as funções anteriores (inclinação e distância) e, assim, penalizar a atribuição de falsos positivos na Matriz de Probabilidades.

3.2.1.3.1.4. Matriz de Probabilidade *a priori* P

A partir da concatenação das três funções de probabilidades apresentadas chega-se a Matriz P proposta nesta pesquisa:

$$P(i, j) = Inp(i, j) + Dtp(i, j) + Np(i, j) \quad (78)$$

Essa Matriz de Probabilidades *a priori* construída a partir das funções descritas acima pode ser usada para avaliar todo um Espaço Amostral $\Omega = (n * m) + m!$ de configurações possíveis para a correspondência de dois conjuntos de planos.

Para determinar a configuração de planos para uma correspondência ótima, nesta pesquisa é proposta uma atualização da Matriz P com Inferência Bayesiana que considera uma configuração *a priori* e funções de probabilidades a partir de relações locais angulares e de distância entre os pontos dos planos mais próximos à origem. Assim, é determinada uma medida condicional da incerteza $P((i, j)/x_k)$, sendo x_k uma

configuração inicial ótima da correspondência dos planos, associada com a ocorrência deste evento x_k , dada uma nova informação disponível. Esta informação é obtida comparando distâncias e ângulos de componentes planares com seus vizinhos em cada nuvem, dada a configuração *a priori*.

3.2.1.3.1.5. Inferência Bayesiana na Atualização da Matriz de Probabilidades

Nesta pesquisa é assumido uma correspondência biunívoca entre planos visando determinar uma configuração ótima dentro de um espaço amostral Ω discreto de configurações possíveis. A Matriz de Probabilidades P contém probabilidades *a priori* da correspondência dentre todas as possíveis combinações de pares de planos, mas não considera relações locais e pode levar a falsos positivos na correspondência. Para melhorar essa estimativa, é proposta a utilização de Inferências Bayesianas a partir de funções de probabilidades que levam em conta relações geométricas locais, ou seja, entre os planos da mesma nuvem de pontos LiDAR 3D e de uma configuração *a priori* com as maiores probabilidades *a priori* determinadas na matriz P . Considerando o caso que se tem 4 planos numerados aleatoriamente para as duas nuvens da Figura 13 a configuração ótima dentro do Espaço Amostral Ω de configurações possíveis para este par de nuvens de pontos LiDAR 3D é dada por:

$$x^* = \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 1 \\ 4 & 4 \end{bmatrix} \quad (79)$$

Entretanto, uma configuração subótima, pode levar a incorreta associação dos pares de planos e ao cálculo de parâmetros de transformação inconsistentes. Para lidar com isso, nesta pesquisa são propostas duas funções de probabilidades que quantificam relações angulares e de distâncias entre os planos de acordo com a configuração *a priori* considerada ótima e atualiza a matriz P usando Inferência Bayesiana. A partir de uma configuração subótima mostrada na Equação (80) para o mesmo par mostrado na Figura 13,

$$x_k = \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 3 \\ 4 & 4 \end{bmatrix} \quad (80)$$

percebe-se que os planos 2 e 4 da nuvem LiDAR 3D de Referência possuem vetores normais perpendiculares entre si, já seus correspondentes 1 e 4, segundo a configuração apresentada na Equação (80), aparentam ser aproximadamente paralelos. Partindo do pressuposto que as relações angulares e de distâncias entre os planos de uma mesma nuvem de pontos LiDAR 3D não se alteram, essas mesmas relações devem ser mantidas entre os planos na nuvem de pontos LiDAR 3D de Pesquisa de acordo com uma determinada configuração.

Uma nova Matriz de Probabilidade *a posteriori* $P1((i,j) \setminus x_k^*)$ contendo medidas condicionais de incertezas é calculada a partir da matriz P e de uma nova informação obtida pelas funções de probabilidades segundo a configuração ótima *a priori* x_k^* . O uso apenas da configuração ótima *a priori* visa reduzir o custo computacional neste processo, uma vez que cada plano na nuvem de pontos LiDAR 3D de Referência é avaliado com todos os planos na nuvem de pontos LiDAR 3D de pesquisa com base apenas na configuração. Os critérios considerados para o cálculo dessa nova medida de verossimilhança na forma de probabilidade podem ser divididos em duas partes: semelhança direcional e semelhança posicional.

As linhas (i) da matriz P representam os planos pertencentes a nuvem LiDAR 3D de Referência, assim, para cada plano é calculado o ângulo entre seu vetor normal e todos os outros também pertencentes aos planos das nuvens de pontos LiDAR 3D de Referência dada a configuração *a priori* x_k^* , como mostra a Equação (81). O mesmo processo é realizado para os planos (j) pertencentes a nuvem de pontos LiDAR 3D de Pesquisa (Equação (81)).

$$\begin{aligned} Ar((i,i) / x_k^*) &= \text{acos}(\langle \vec{n}_i, \vec{n}_{i=1\dots n} \rangle) \\ Ap((j,j) / x_k^*) &= \text{acos}(\langle \vec{n}_j, \vec{n}_{j=1\dots m} \rangle) \end{aligned} \quad (81)$$

O cálculo dessa medida de verossimilhança baseia-se no fato de que planos corretamente associados mantém as relações angulares de seus vetores normais iguais entre si em cada nuvem de pontos LiDAR 3D. Assim, uma matriz resultante das diferenças é calculada:

$$Arp((i,j)/ x_k^*) = |Ar((i,i)/ x_k^*) - Ap((j,j)/ x_k^*)| \quad (82)$$

Cada coluna dessa matriz de diferenças angulares contribuirá para o cálculo da probabilidade associada a um plano de referência com base em uma relação com a soma de todas os valores da coluna.

$$Ainb((i,j)/ x_k^*) = \frac{Arp((i,j)/ x_k^*)}{\sum_{j=1}^m Arp((i,j)/ x_k^*)} - 1 \quad (83)$$

Portanto, conforme a Equação (83), valores baixos de $Arp((i,j)/ x_k^*)$ em relação as colunas retornam probabilidades maiores para o par (i,j) .

Como os centroides dos planos são diretamente ligados à distribuição espacial dos pontos e devido ao fato deste conjunto de dados ter como característica a obtenção durante o trânsito normal das pessoas, planos homólogos podem estar sujeitos a obstruções apresentando cardinalidades diferentes. Portanto, é determinado um ponto virtual P_o , que representa o ponto, pertencente ao plano, mais próximo da origem. O cálculo deste ponto para um plano qualquer na nuvem de pontos LiDAR 3D de Referência e Pesquisa, respectivamente, é apresentado na Equação (84) a seguir.

$$P_o = d * [n_x, n_y, n_z] \quad (84)$$

sendo d a distância entre o ponto P_o e a origem do sistema de coordenadas cartesianas tridimensional. Assim, dada uma configuração ótima *a priori* x_k^* , a distância entre os pontos P_o dos planos em cada nuvem de pontos LiDAR 3D devem ser imutáveis entre si. O cálculo da distância é ilustrado na Equação (85).

$$\begin{aligned}
Dtr(P_o(i,i)/ x_k^*) &= \sqrt{\left(P_{ox_i} - P_{ox_{i=1\dots n}}\right)^2 + \left(P_{oy_i} - P_{oy_{i=1\dots n}}\right)^2 + \left(P_{oz_i} - P_{oz_{i=1\dots n}}\right)^2} \\
Dtp(P_o(j,j)/ x_k^*) &= \sqrt{\left(P_{ox_j} - P_{ox_{j=1\dots m}}\right)^2 + \left(P_{oy_j} - P_{oy_{j=1\dots m}}\right)^2 + \left(P_{oz_j} - P_{oz_{j=1\dots m}}\right)^2}
\end{aligned} \tag{85}$$

O cálculo da medida de verossimilhança para esta função de probabilidade também baseia-se no fato de que planos corretamente associados mantêm as relações de distância entre seus pontos P_o iguais entre si em cada nuvem de pontos LiDAR 3D. Assim, uma matriz resultante das diferenças é calculada da seguinte forma:

$$Dtrp(P_o(i,j)/ x_k^*) = |Dtr(P_o(i,i)/ x_k^*) - Dtp(P_o(j,j)/ x_k^*)| \tag{86}$$

Cada coluna dessa matriz de diferenças posicionais, assim como na função de probabilidade anterior, contribuirá para o cálculo de uma probabilidade associada a um par de planos com base em todas os seus valores de P_o .

$$Dinb((i,j)/ x_k^*) = \frac{Dtrp(P_o(i,j)/ x_k^*)}{\sum_{j=1}^m Dtrp(P_o(i,j)/ x_k^*)} - 1 \tag{87}$$

A partir das duas funções de probabilidades obtidas por semelhanças direcionais e posicionais, chega-se a uma nova medida de verossimilhança conforme a configuração ótima *a priori* x_k^* :

$$Pinb((i,j)/ x_k^*) = \omega_4 * Ainb((i,j)/ x_k^*) + (1 - \omega_4) * Dinb((i,j)/ x_k^*) \tag{88}$$

sendo $\omega_4 = 0.5$ o peso (valor determinado empiricamente) relacionado a essas funções.

Para determinar a probabilidade de uma configuração qualquer, basta multiplicar as probabilidades individuais de seus componentes (associação de pares de nuvens de pontos LiDAR 3D) dentre as linhas da matriz P disponíveis.

$$P(x_k^*) = \prod_{i=1}^n P(i, j) \mid (i, j) \in x_k^* \text{ e } j = 1 \dots m \quad (89)$$

Assim, a Matriz de Probabilidades *a posteriori* $P1$ é calculada conforme mostra a Equação (90) a seguir:

$$P1((i, j) / x_k^*) = \frac{Pinb((i, j) / x_k^*) * (\prod_{i=1}^n P(i, j) \mid (i, j) \notin x_k^*)}{P(x_k^*)} \quad (90)$$

sendo,

- i) $P1((i, j) / x_k^*)$ a medida condicional da incerteza associada a configuração ótima *a priori* x_k^* , chamada de probabilidade *a posteriori*.
- ii) $Pinb((i, j) / x_k^*)$ a medida de verossimilhança representada pela probabilidade obtida com as funções de semelhanças descritas anteriormente.
- iii) $\prod_{i=1}^n P(i, j) \mid (i, j) \notin x_k^*$ a probabilidade da configuração x_k^* sem considerar a probabilidade *a priori* da correspondência (i, j) na configuração x_k^* .
- iv) $P(x_k^*)$ a probabilidade da configuração x_k^* , chamada de verossimilhança marginal.

A configuração que apresentar as maiores probabilidades será usada na correspondência dos planos. Um limiar (*setMinProb*) de certeza sobre as probabilidades foi aplicado, assim, se a maior probabilidade *a posteriori* calculada for menor que este limiar ($P1(i, j) < setMinProb$), essa correspondência não é adicionada a configuração final x^* .

3.2.1.3.2. Determinação do Espaço Canônico Gerado pelos Planos

Diante da gama de possibilidades para as formas de objetos que podem aparecer num levantamento com um sensor LiDAR 3D em ambientes urbanos, uma das maiores adversidades em abordagens baseadas em feições é a dimensionalidade gerada pelas observações, ou seja, o Espaço Canônico gerado. Caso todos os vetores normais tenham a mesma direção, por exemplo, o sistema apresentará problemas de posto e não

será possível determinar uma solução única dos parâmetros de transformação. Para se determinar quando há um Espaço Canônico tridimensional sendo gerado, nesta pesquisa, é proposta uma abordagem direta a partir do produto escalar entre os vetores normais dos planos.

Primeiramente, o limiar V é estabelecido indicando que o vetor normal de um plano qualquer se propaga em uma das três direções consideradas neste trabalho: X , Y e Z . Assim, uma análise é feita dos valores das componentes dos vetores normais dos planos e se detectado valores na componente vertical n_z superiores a V é definida a dimensionalidade Z ao sistema. Se o critério for atendido para uma outra direção apenas, então o Espaço Canônico é bidimensional e uma solução alternativa é proposta nesta pesquisa.

Caso sejam detectados valores superiores a V nas três componentes, deve ser verificado se formam um Espaço Canônico entre si. Assim, para cada vetor normal que se propaga na direção do eixo X , é feito o produto escalar com todos os vetores que se propagam na direção do eixo Y e caso algum valor encontrado seja menor que V , então é definido um Espaço Canônico com três dimensões para este sistema. Na prática, isso significa que pelo menos dois vetores apresentam certa perpendicularidade, o que garante um condicionamento adequado do sistema de equações, necessário para uma solução fechada (além da trivial) dos parâmetros de transformação entre os pares de nuvens de pontos LiDAR 3D. Por fim, se nenhum produto escalar atender ao critério, o Espaço Canônico é considerado bidimensional.

3.2.1.3.3. Estimativa dos Parâmetros de Transformação com Planos e a Álgebra de Dual Quaternions para um Espaço Canônico 3D

Para a estimativa dos parâmetros de rotação, considerando a primeira restrição apresentada na Equação (42, apenas a parte da Equação (41: $E = 2Q^T W_1 Q$ é utilizada. No caso, as coordenadas dos pontos p_i^p e p_i^r são substituídas pelos coeficientes dos vetores normais \vec{n}_i^p e \vec{n}_i^r , respectivamente. Portanto, a partir da Equação (91 é realizada a Decomposição por Valores Singulares e Q é determinado como o autovetor da matriz W_1 associado ao menor autovalor correspondente.

$$Q^T W_1 Q = Q^T \left(\sum_{i=1}^n \left([L_{\vec{n}_i^p}]^T [R_{\vec{n}_i^r}] \right) \right) Q = 0 \quad (91)$$

sendo n o número de planos correspondentes.

Em relação a translação, a partir das Equações (92) e (93),

$$\vec{n}_i^p \cdot p_i^p + d_i^p = 0 \quad (92)$$

$$\vec{n}_i^r \cdot p_i^r + d_i^r = 0 \quad (93)$$

sendo,

a) “ · ” o produto escalar (produto interno padrão do Espaço Euclidiano) entre dois vetores reais.

b) \vec{n}_i^p e d_i^p o vetor normal e a distância para a origem de um plano qualquer na nuvem de pontos 3D de Pesquisa.

c) p_i^p um ponto qualquer pertencente ao plano de Pesquisa.

d) \vec{n}_i^r e d_i^r o vetor normal e a distância para a origem de um plano qualquer na nuvem de pontos 3D de Referência.

e) p_i^r um ponto qualquer pertencente ao plano de Referência.

Assim, é possível realizar uma simplificação da Equação (32), como mostra a Equação (94) a seguir:

$$\vec{n}_i^r \cdot (R p_i^p + \vec{t}) + d_i^r = 0 \quad (94)$$

assim,

$$\vec{n}_i^r \cdot R p_i^p + \vec{n}_i^r \cdot \vec{t} + d_i^r = 0 \quad (95)$$

o que implica em

$$R^T \vec{n}_i^r \cdot p_i^p + \vec{n}_i^r \cdot \vec{t} + d_i^r = 0 \quad (96)$$

como $\vec{n}_i^r = R \vec{n}_i^p$, então:

$$R^T R \vec{n}_i^p \cdot p_i^p + \vec{n}_i^r \cdot \vec{t} + d_i^r = 0 \quad (97)$$

portanto,

$$-d_i^p + \vec{n}_i^r \cdot \vec{t} + d_i^r = 0 \quad (98)$$

A partir da Equação (98), para determinar a translação \vec{t} , nesta pesquisa, é proposta a minimização do seguinte MMF:

$$E = \sum_{i=1}^n (-d_i^p + \vec{n}_i^r \cdot \vec{t} + d_i^r)^2 = \min \quad (99)$$

O somatório acima pode ser reescrito da seguinte forma:

$$\begin{aligned} E &= \sum_{i=1}^n (\vec{n}_i^r \cdot \vec{t} + d_i^r - d_i^p)^2 = \\ &= \sum_{i=1}^n \left((\vec{n}_i^r \cdot \vec{t})^2 + 2(\vec{n}_i^r \cdot \vec{t})(d_i^r - d_i^p) + (d_i^r - d_i^p)^2 \right) = \\ &= \sum_{i=1}^n (\vec{n}_i^r \cdot \vec{t})^2 + 2 \sum_{i=1}^n (\vec{n}_i^r \cdot \vec{t})(d_i^r - d_i^p) + \sum_{i=1}^n (d_i^r - d_i^p)^2 \end{aligned} \quad (100)$$

Como todos os vetores normais \vec{n}_i^r sendo $1 \leq i \leq n$ (n o número de planos correspondentes para o par de nuvens de pontos LiDAR 3D) são unitários, então:

$$E = \sum_{i=1}^n (\vec{n}_i^r \cdot \vec{t})^2 + 2 \sum_{i=1}^n (\vec{n}_i^r \cdot \vec{t})(d_i^r - d_i^p) + \sum_{i=1}^n (d_i^r - d_i^p)^2 = \min \quad (101)$$

Considerando $\sum_{i=1}^n (d_i^r - d_i^p)^2 = c$ uma constante, chega-se a expressão:

$$E = c + \sum_{i=1}^n (\vec{n}_i^r \cdot \vec{t})^2 + 2 \sum_{i=1}^n (\vec{n}_i^r \cdot \vec{t})(d_i^r - d_i^p) \quad (102)$$

A partir da Equação (102), os produtos escalares podem ser desenvolvidos e o termo E pode ser reescrito da seguinte forma:

$$\begin{aligned} E &= c + \sum_{i=1}^n ((\vec{t}, \vec{n}_i^r))^2 + 2 \sum_{i=1}^n (d_i^r - d_i^p) \langle \vec{t}, \vec{n}_i^r \rangle = \\ &= c + \sum_{i=1}^n (\langle (\vec{t}q\bar{q}), \vec{n}_i^r \rangle)^2 + 2 \sum_{i=1}^n (d_i^r - d_i^p) \langle (\vec{t}q\bar{q}), \vec{n}_i^r \rangle = \\ &= c + \sum_{i=1}^n (\langle (\vec{t}q), (\vec{n}_i^r q) \rangle)^2 + 2 \sum_{i=1}^n (d_i^r - d_i^p) \langle (\vec{t}q), (\vec{n}_i^r q) \rangle = \\ &= c + 4 \sum_{i=1}^n (\langle \hat{q}, (\vec{n}_i^r q) \rangle)^2 + 4 \sum_{i=1}^n (d_i^r - d_i^p) \langle \hat{q}, (\vec{n}_i^r q) \rangle = \end{aligned} \quad (103)$$

para $1 \leq i \leq n$. Introduzindo os conceitos de Quaternions e reescrendo o produto à esquerda $\vec{n}_i^r q$, assim como mostra a Equação (38), por meio da multiplicação de uma matriz 4×4 por um vetor coluna 4×1 , ou seja:

$$[L_{\vec{n}_i^r}]Q = \begin{bmatrix} 0 & -n_{x_i} & -n_{y_i} & -n_{z_i} \\ n_{x_i} & 0 & -n_{z_i} & n_{y_i} \\ n_{y_i} & n_{z_i} & 0 & -n_{x_i} \\ n_{z_i} & -n_{y_i} & n_{x_i} & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (104)$$

O produto interno $\langle \hat{q}, (\vec{n}_i^r q) \rangle$ pode ser escrito como produtos de matrizes:

$$\hat{Q}^T [L_{\vec{n}_i^r}]Q = Q^T [L_{\vec{n}_i^r}]^T \hat{Q} \quad (105)$$

Reescrevendo $(d_i^r - d_i^p) \langle \hat{q}, (\vec{n}_i^r q) \rangle$ como produtos de matrizes:

$$(d_i^r - d_i^p)\hat{Q}^T \left([L_{\vec{n}_i^r}]Q \right) = \hat{Q}^T \left((d_i^r - d_i^p)[L_{\vec{n}_i^r}] \right) Q \quad (106)$$

Logo, o somatório acima equivale ao produto das matrizes:

$$\begin{aligned} E &= c + 4 \sum_{i=1}^n \left(\hat{Q}^T [L_{\vec{n}_i^r}]Q \right)^2 + 4 \sum_{i=1}^n \left(\hat{Q}^T \left((d_i^r - d_i^p)[L_{\vec{n}_i^r}] \right) Q \right) = \\ &= c + 4\hat{Q}^T \left(\sum_{i=1}^n \left([L_{\vec{n}_i^r}]Q Q^T [L_{\vec{n}_i^r}]^T \right) \right) \hat{Q} + 4\hat{Q}^T \left(\sum_{i=1}^n \left((d_i^r - d_i^p)[L_{\vec{n}_i^r}] \right) \right) Q = \\ &= c + 4\hat{Q}^T W_4 \hat{Q} + 4\hat{Q}^T W_5 Q \end{aligned} \quad (107)$$

sendo $W_4 = \sum_{i=1}^n \left([L_{\vec{n}_i^r}]Q Q^T [L_{\vec{n}_i^r}]^T \right)$ e $W_5 = \sum_{i=1}^n \left((d_i^r - d_i^p)[L_{\vec{n}_i^r}] \right)$.

Assim, para encontrar \hat{Q} o somatório deve ser minimizado com a seguinte restrição:

$$\hat{Q}^T Q = 0 \quad (108)$$

ou

$$4\hat{Q}^T Q = 0 \quad (109)$$

Para isso, o método dos Multiplicadores de Lagrange pode ser utilizado fornecendo uma solução para os mínimos do somatório sujeita a condição acima.

$$\tilde{E} = c + 2\hat{Q}^T W_4 \hat{Q} + 4\hat{Q}^T W_5 Q + 4\lambda \hat{Q}^T Q \quad (110)$$

onde λ é o multiplicador de Lagrange. As derivadas parciais são:

$$\frac{\partial \tilde{E}}{\partial \hat{Q}} = 4(W_4 + W_4^T)\hat{Q} + 4W_5 Q + 4\lambda Q = 0 \quad (111)$$

Assim, as duas equações acima são equivalentes ao sistema:

$$\begin{cases} 4(W_4 + W_4^T)\hat{Q} + 4W_5Q + 4\lambda Q = 0 \\ 4\hat{Q}^T Q = 0 \end{cases} \quad (112)$$

como W_4 é simétrica, então para resolver essas equações, tem-se:

$$\begin{cases} (2W_4)\hat{Q} + Q\lambda = -W_5Q \\ Q^T\hat{Q} = 0 \end{cases} \quad (113)$$

na forma matricial

$$\begin{cases} [2W_4 & Q] \begin{bmatrix} \hat{Q} \\ \lambda \end{bmatrix} = -W_5Q \\ [Q^T & 0] \begin{bmatrix} \hat{Q} \\ \lambda \end{bmatrix} = 0 \end{cases} \quad (114)$$

ou seja,

$$\begin{bmatrix} 2W_4 & Q \\ Q^T & 0 \end{bmatrix} \begin{bmatrix} \hat{Q} \\ \lambda \end{bmatrix} = \begin{bmatrix} -W_5Q \\ 0 \end{bmatrix} \quad (115)$$

Portanto, basta a solução da Equação (116) a seguir para a determinação da parte dual \hat{Q} :

$$\begin{bmatrix} \hat{Q} \\ \lambda \end{bmatrix} = \left(\begin{bmatrix} 2W_4 & Q \\ Q^T & 0 \end{bmatrix} \right)^{-1} \begin{bmatrix} -W_5Q \\ 0 \end{bmatrix} \quad (116)$$

Para determinar a inversa da matriz apresentada na Equação (116), nesta pesquisa foi usado o algoritmo de decomposição LU. A partir da determinação do vetor \hat{Q} , e da Equação (117):

$$\hat{Q} = \frac{1}{2}[R_q]T \quad (117)$$

a translação T é determinada pela Equação (118):

$$T = 2[R_q]^T \hat{Q} \quad (118)$$

Para demonstrar que a matriz $\begin{bmatrix} 2W_4 & Q \\ Q^T & 0 \end{bmatrix}$ é invertível, seja o sistema homogêneo,

$$\begin{bmatrix} 2W_4 & Q \\ Q^T & 0 \end{bmatrix} \begin{bmatrix} X \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (119)$$

ou seja,

$$\begin{cases} 2W_4 X + Qy = 0 \\ Q^T X = 0 \end{cases} \quad (120)$$

Supondo por contrassenso que X é não nula, então Q^T é nula o que é impossível. Logo $X = 0$. Assim, pela Equação (119) tem-se que:

$$Qy = 0 \quad (121)$$

Portanto $y = 0$. Como o sistema homogêneo mostrado na Equação (120) só tem solução trivial, então $\begin{bmatrix} 2W_4 & Q \\ Q^T & 0 \end{bmatrix}$ é invertível.

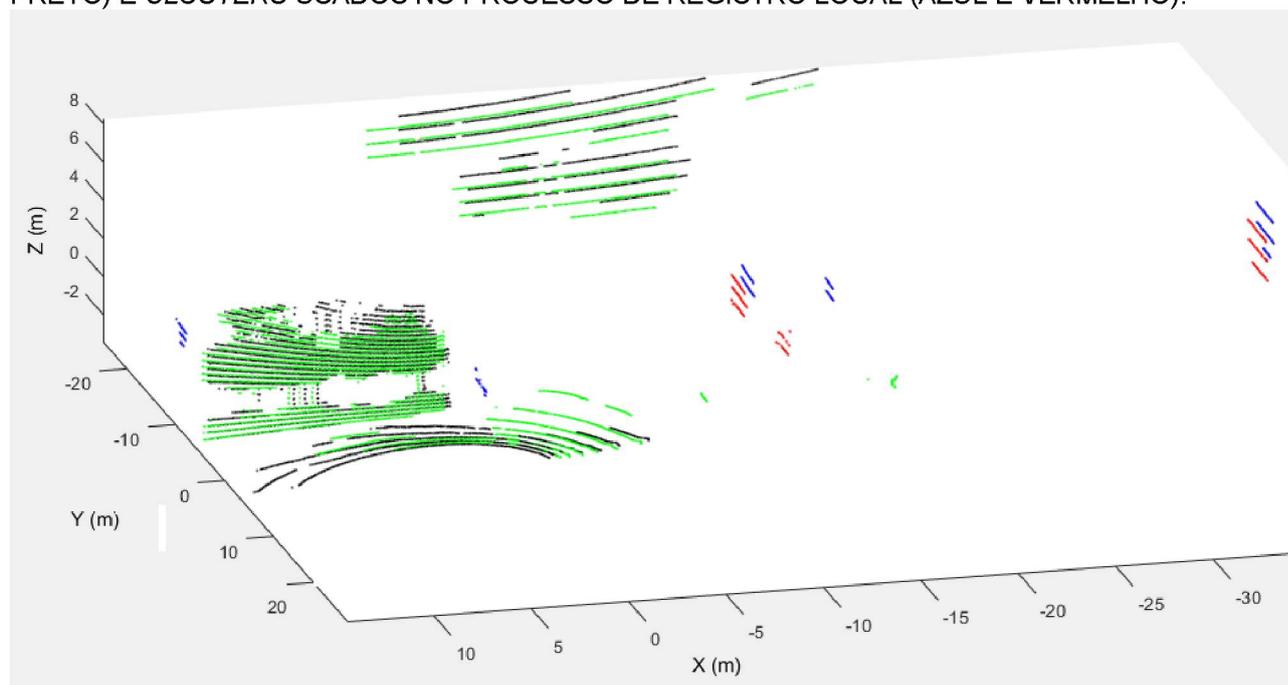
3.2.1.3.4. Determinação dos Parâmetros de Transformação com um Espaço Canônico Bidimensional

Quando um par de nuvens de pontos LiDAR 3D possui planos correspondidos que geram um Espaço Canônico em apenas duas direções, não é possível determinar todos os parâmetros de translação devido ao mal condicionamento do sistema de equações. Assim, para contornar esse problema, nesta pesquisa é proposto o uso da primitiva ponto para estimar a translação na direção ausente.

Com o objetivo de determinar correspondências entre pontos nas nuvens LiDAR 3D, nesta pesquisa é proposto o uso de *clusters*. Essas superfícies foram descartadas pelo algoritmo Crescimento de Região devido a sua baixa cardinalidade ou elevado grau de curvatura e podem ser encontradas facilmente, por exemplo, em laterais de veículos,

faces de lixeiras, partes de paredes parcialmente obstruída e etc. Portanto, os *clusters* são usados para definir regiões de busca para o estabelecimento de correspondências entre pontos a partir de relações de distância entre o centroide dos *clusters* e os planos que foram detectados pelo algoritmo Crescimento de Região. A Figura 15 mostra em vermelho e azul os *clusters* extraídos das nuvens de pontos LiDAR 3D de Referência e Pesquisa, respectivamente, e, em verde e preto, os planos extraídos das nuvens de pontos LiDAR 3D de Referência e Pesquisa, nesta ordem.

FIGURA 15 – PLANOS DETECTADOS PELO ALGORITMO DE CRESCIMENTO DE REGIÃO (VERDE E PRETO) E *CLUSTERS* USADOS NO PROCESSO DE REGISTRO LOCAL (AZUL E VERMELHO).



FONTE: O autor (2020)

Para a correta associação de pontos neste tipo de conjunto de dados é imprescindível que haja uma correta associação entre os *clusters*. Assim, o algoritmo de correspondência abordado na Seção (3.2.1.3.1) é aplicado, resultando em uma configuração de *clusters* correspondentes. Entretanto, para aumentar as probabilidades de uma detecção ótima, alguns parâmetros do algoritmo de correspondência foram adaptados, principalmente em relação as componentes direcionais devido as curvaturas mais elevadas dos *clusters*. Assim, os pesos ω_1 e ω_2 representados nas Equações (71) e (75) foram ambos igualados a 0,4.

Caso pelo menos uma correspondência entre *clusters* seja confirmada, o algoritmo continua com a determinação dos planos, chamados de planos chave, que serão usados no processo de busca pelos pontos pseudo-conjugados. Esses dois planos chave, um pertencente e outro não ao terreno, podem ser usados para estabelecer uma relação de distância com o centroide do *cluster* na nuvem de pontos LiDAR 3D de Referência. Os planos chave são escolhidos com base em sua cardinalidade, ou seja, os planos com o maior número de pontos. Dessa forma, são calculadas as distâncias ortogonais ponto a plano entre o centroide do *cluster* e os plano chave na nuvem de pontos LiDAR 3D de Referência, por meio da Equação (122).

$$\begin{aligned} d(C_{cl_r}, pl_{r_t}) &= \vec{n}_{r_t} * C_{cl_r} - d_{r_t} \\ d(C_{cl_r}, pl_{r_{at}}) &= \vec{n}_{r_{at}} * C_{cl_r} - d_{r_{at}} \end{aligned} \quad (122)$$

sendo,

- i) C_{cl_r} o centroide de um *cluster* qualquer presente na nuvem de pontos LiDAR 3D de Referência.
- ii) pl_{r_t} o plano chave, pertencente ao terreno, presente na nuvem de pontos LiDAR 3D de Referência e representado pelos parâmetros \vec{n}_{r_t} e d_{r_t} .
- iii) $pl_{r_{at}}$ o plano chave, acima do terreno, presente na nuvem de pontos LiDAR 3D de Referência e representado pelos parâmetros $\vec{n}_{r_{at}}$ e $d_{r_{at}}$.

A partir dessas medidas de distâncias calculadas na nuvem de pontos LiDAR 3D de Referência entre os *clusters* e os planos chave, é possível determinar um conjunto de pontos, pertencentes aos *clusters* na nuvem de pontos LiDAR 3D de Pesquisa, com as medidas equivalentes (dentro de um determinado intervalo) para os planos chave na respectiva nuvem. Assim, para determinar esses pontos, é feito o cálculo das diferenças entre o valor encontrado na Equação (122) e as distâncias de todos os pontos pertencentes ao *cluster* correspondente com os planos chave na nuvem de pontos LiDAR 3D de Pesquisa. Aqueles candidatos que apresentarem o valor da diferença menor que um limiar pré-estabelecido e serão usados no cálculo do ponto virtual, pseudo-conjugado, ao centroide do *cluster* da nuvem de pontos LiDAR 3D de Referência. A Equação (123)

ilustra o cálculo das diferenças das distâncias, realizado nesse processo, para os planos chave acima e pertencentes ao terreno.

$$\begin{aligned} Dft(P_{cl_p}) &= (\vec{n}_{p_t} * P_{cl_p} - d_{p_t}) - d(C_{cl_r}, pl_{r_t}) \\ Dfat(P_{cl_p}) &= (\vec{n}_{p_{at}} * P_{cl_p} - d_{p_{at}}) - d(C_{cl_r}, pl_{r_{at}}) \end{aligned} \quad (123)$$

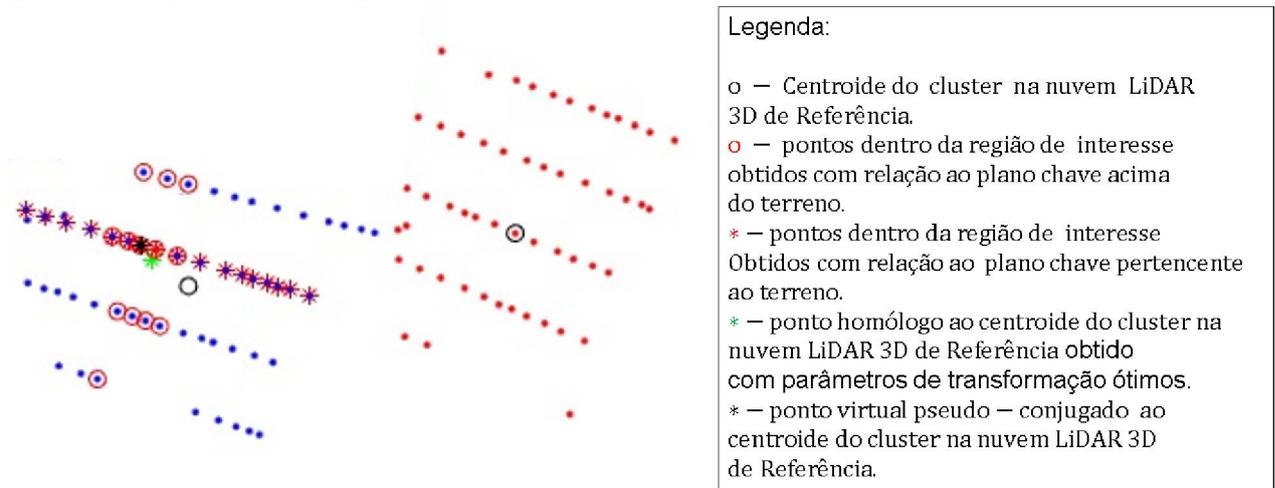
sendo,

- i) $Dft(P_{cl_p})$ o valor da diferença encontrada entre as distâncias do centroide de um *cluster* qualquer e o plano chave, pertencente ao terreno, na nuvem de pontos LiDAR 3D de Referência e de um ponto qualquer P_{cl_p} pertencente ao *cluster* e o plano chave correspondente, pertencente ao terreno, na nuvem de pontos LiDAR 3D de Pesquisa;
- ii) \vec{n}_{p_t} e d_{p_t} parâmetros dos planos chave pertencente ao terreno presentes na nuvem de pontos LiDAR 3D de Pesquisa;
- iii) $\vec{n}_{p_{at}}$ e $d_{p_{at}}$ parâmetros dos planos chave acima do terreno presentes na nuvem de pontos LiDAR 3D de Pesquisa;

A partir do conjunto de pontos pertencentes ao intervalo da região de interesse, uma simples média aritmética desses pontos pode ser usada como uma estimativa da posição do ponto pseudo-conjugado. A Figura 16 a seguir mostra um exemplo da definição da região de interesse em um *cluster*, pertencente a nuvem de pontos LiDAR 3D de Pesquisa (azul) a partir do centroide do *cluster* correspondente (vermelho) na nuvem de pontos LiDAR 3D de Referência.

Como pode ser visto na Figura 16, os pontos pseudo-conjugado (preto) e o obtido com os parâmetros de transformação ótimos estão próximos entre si (verde). A partir da obtenção dos pontos, os mesmos podem ser usados para a estimativa de todos os componentes do vetor de translação usando a Equação (59) uma vez que o Quaternion Q foi determinado usando os vetores normais dos planos e $W_2 = \sum_{i=1}^n \left(-[L_{p_i^r}] + [R_{p_i^p}] \right)$ pode ser calculado usando os pontos virtuais.

FIGURA 16 – CLUSTERS CORRESPONDENTES NA NUVEM DE PONTOS LIDAR 3D DE REFERÊNCIA (VERMELHO) E PESQUISA (AZUL).



FONTE: O autor (2020)

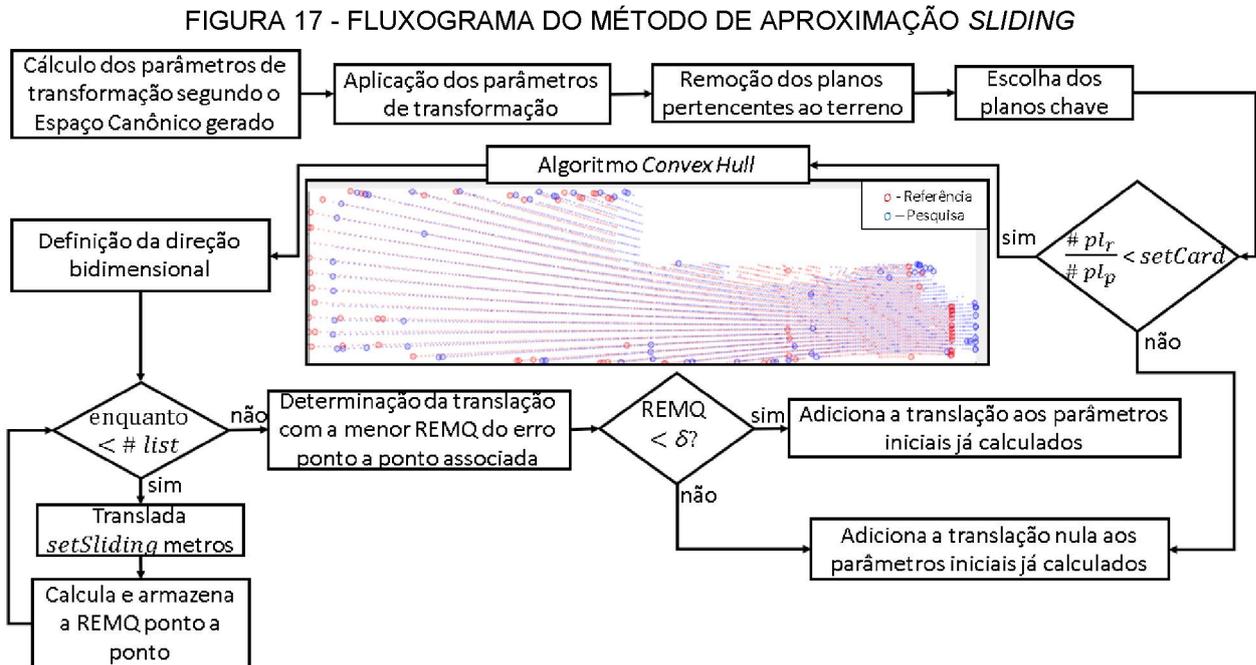
A seguir uma descrição do método para lidar com a ausência de clusters no pares de nuvens de pontos LiDAR 3D.

3.2.1.3.5. Método de Aproximação *Sliding*

Diante de observações que geram um Espaço Canônico bidimensional e a ausência de *clusters*, o método de registro grosseiro proposto nesta pesquisa explora a distribuição espacial dos planos extraídos acima do terreno para calcular os parâmetros de transformação. Essa estimativa é obtida a partir dos parâmetros de transformação determinados de acordo com os planos disponíveis e de um processo iterativo chamado de *Sliding*. Após aplicar os parâmetros obtidos com os planos apenas nas duas direções disponíveis, o *Sliding* consiste em estimar um polígono convexo (algoritmo *Convex Hull* – RUSU, 2011) dos planos chave da nuvem de pontos LiDAR 3D de Referência e Pesquisa, acima do terreno, e, deslizar o plano de Pesquisa sobre o de Referência segundo uma direção bidimensional para determinar uma sobreposição máxima desses contornos. O Fluxograma na Figura 17 ilustra esse processo.

Como pode ser visto na Figura 17, a aproximação *Sliding* inicia com o cálculo dos parâmetros de transformação segundo as direções do Espaço Canônico gerado e, em

seguida, com a aplicação desses parâmetros às nuvens de pontos LiDAR 3D de Pesquisa, conforme mostra a Figura 18b.

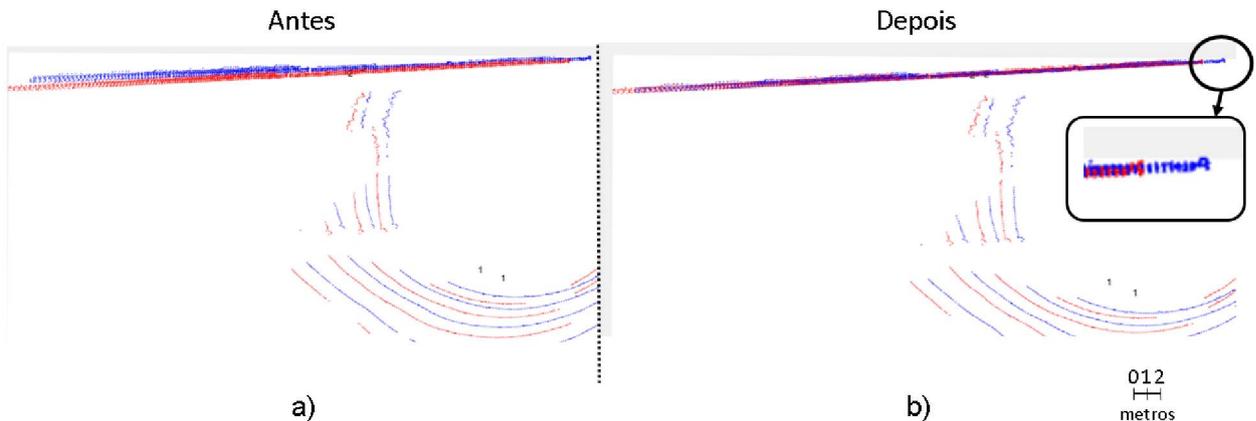


FONTE: O autor (2020)

Como nem todos os parâmetros de translação ótimos foram definidos, os planos não se sobrepõem completamente, como mostra o detalhe (ampliado) na Figura 18b. O processo de *Sliding* ocorre a partir de dois planos chave correspondentes pertencentes às nuvens de pontos LiDAR 3D de Referência e de Pesquisa, presentes acima do terreno. Assim, os planos do terreno são removidos e o plano chave é escolhido como aquele de maior cardinalidade. Para dar mais confiança a esse processo, os dois planos chave correspondentes em ambas as nuvens devem ter uma determinada proporção (*setCard*) de pontos.

Para diminuir o custo computacional da aproximação, foi aplicado o algoritmo *Convex Hull* para detectar apenas pontos pertencentes às extremidades dos planos e usá-los para criar uma métrica de alinhamento entre os planos com base no valor da REMQ ponto a ponto. Para estabelecer essa medida, é calculada a REMQ das distâncias entre os pontos do plano chave da nuvem LiDAR 3D de Referência e o ponto mais próximo no plano chave da nuvem LiDAR 3D de Pesquisa.

FIGURA 18 - PAR DE PLANOS ANTES (a) E APÓS (b) A APLICAÇÃO DOS PARÂMETROS DE TRANSFORMAÇÃO PRELIMINARES À NUVEM DE PONTOS LIDAR 3D DE PESQUISA (VERMELHO) NO PROCESSO DE *SLIDING*. NUVEM DE PONTOS LIDAR 3D DE REFERÊNCIA EM AZUL.

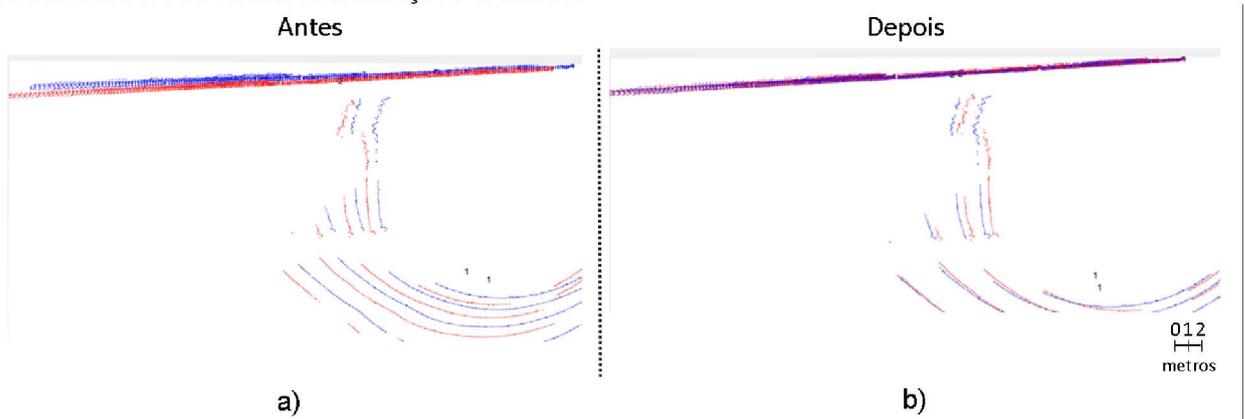


FONTE: O autor (2020)

O próximo passo no processo de *Sliding* consiste em definir a direção bidimensional que o plano chave na nuvem de pontos LiDAR 3D de Pesquisa deve seguir para que a sobreposição entre os dois planos chave correspondentes seja a maior possível. Essa direção é obtida com a diferença de posição dos centroides dos planos chave como um vetor normalizado e com a componente vertical nula. Em seguida, esse vetor de direção é multiplicado por um vetor de escalares chamado de *list* com um intervalo de valores variando (*setSliding*) centímetros entre si.

Assim, os pontos detectados nas extremidades do plano chave da nuvem de pontos LiDAR 3D de Pesquisa são transladados de acordo com o conjunto de vetores determinados em função da direção. Por fim, é escolhido o vetor que resultar na menor REMQ ponto a ponto e caso esse valor fique abaixo de um limiar V , então essa translação é adicionada à inicial. A Figura 19 mostra, em uma perspectiva vista de cima, o resultado antes a após a aplicação dos parâmetros de transformação com o método de aproximação *Sliding* para a mesmo par de nuvens de pontos LiDAR 3D mostrado na Figura 18. Assim, a componente translacional determinada a partir do método de *Sliding* é adicionada ao vetor de translação correspondente ao método de registro grosseiro proposto neste trabalho.

FIGURA 19 – PLANOS CHAVE ANTES (a) E APÓS (b) A APLICAÇÃO DE TODOS OS COMPONENTES TRANSLACIONAIS NA APROXIMAÇÃO *SLIDING*.



FONTE: O autor (2020)

A seguir, uma descrição do registro fino utilizado na presente pesquisa.

3.2.1.3.6. Registro Fino

Tendo em vista a impossibilidade na garantia da presença de planos e que estes formem um Espaço Canônico tridimensional nas nuvens de pontos LiDAR 3D, faz-se necessário fazer um refinamento dos parâmetros de transformação do registro grosseiro visando a obtenção dos parâmetros de transformação ótimos. Para isso, nesta pesquisa foi aplicado o algoritmo ICP para o registro fino. Para o alinhamento inicial são usados os parâmetros de transformação obtidos pelo método de registro grosseiro proposto nesta pesquisa. Conforme o alinhamento inicial se aproxima do alinhamento ótimo, menos iterações são necessárias e consequentemente menor o custo computacional.

A parte iterativa do algoritmo ICP aplicado nesta pesquisa apresenta os seguintes passos:

- 1) Para cada ponto da nuvem de pontos LiDAR 3D de Referência são localizados os vizinhos mais próximos na nuvem de pontos LiDAR 3D de Pesquisa. Nessa etapa são estabelecidos os pares de pontos pseudo-conjugados. Para acelerar esse processo essa busca é otimizada com o emprego da estrutura de dados *kd-tree*.
- 2) Utilizando uma função chamada de *Outlier Filter* (RUSU, 2011), o algoritmo remove correspondências consideradas incorretas.

3) Os parâmetros de transformação entre as duas nuvens de pontos LiDAR 3D são calculados pelo método proposto por Horn (1987) e aplicados na nuvem de pontos LiDAR 3D de Pesquisa. O algoritmo retorna uma Matriz Homogênea composta por uma matriz de transformação de corpo rígido 3D e um vetor de translação.

4) Por fim, é calculado a REMQ ponto a ponto entre as duas nuvens e comparado com o critério de convergência. Se o valor da REMQ não alcançar o limiar mínimo (δ) ou o número máximo (t) de iterações não for alcançado, o processo é repetido novamente. Caso contrário, o algoritmo encerra sua execução.

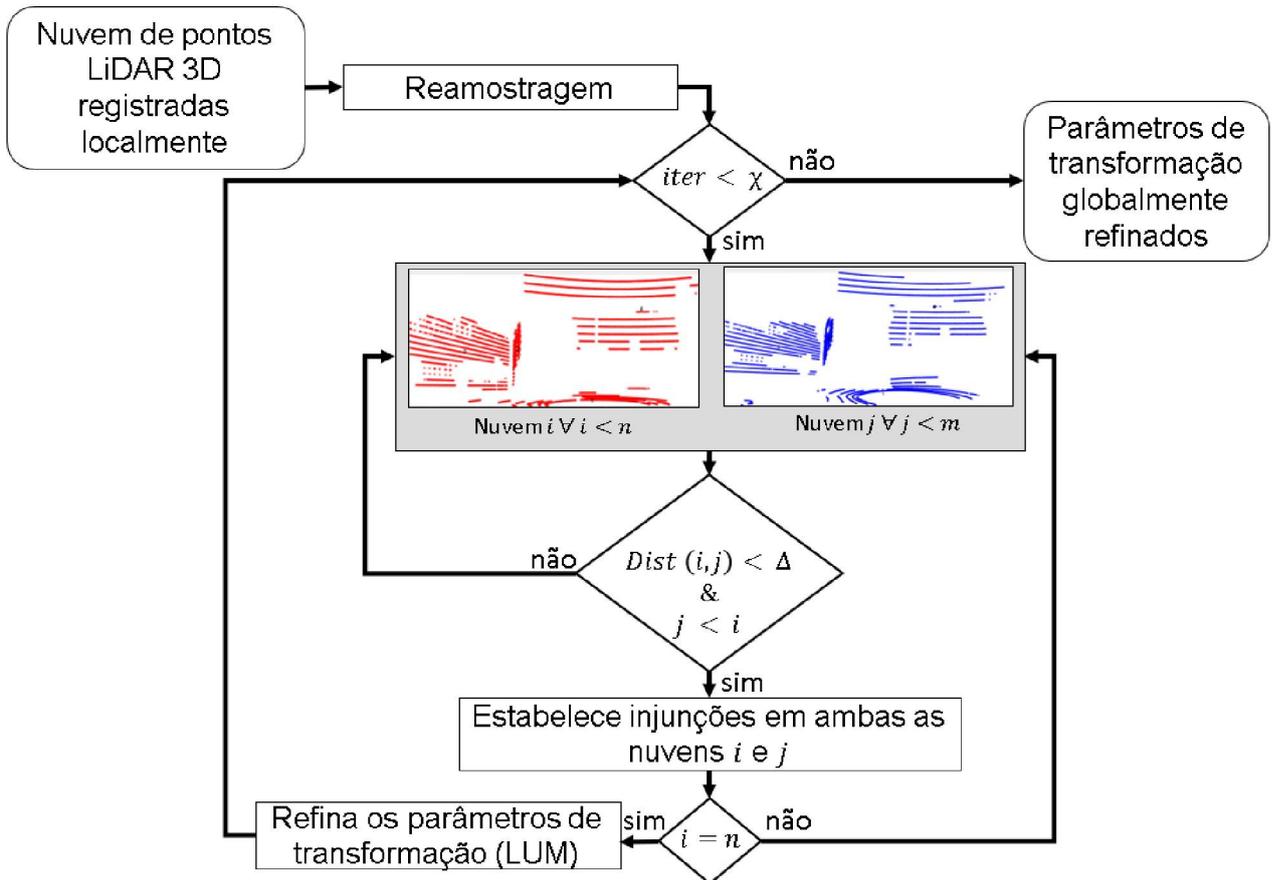
A partir da última iteração do algoritmo e dependendo de uma avaliação dos resultados a partir do valor da REMQ ponto a plano, os parâmetros de transformação refinados entre as nuvens de pontos LiDAR 3D de Referência e de Pesquisa são calculados e aplicados às nuvens de pontos LiDAR 3D de Pesquisa.

3.2.2. Refinamento Global

Devido ao acúmulo de erros ao se registrar localmente pares de nuvens de pontos LiDAR 3D faz-se necessário aplicar um método de Refinamento Global para garantir consistência da nuvem global gerada. Nesta pesquisa foi usado o algoritmo LUM desenvolvido por Lu e Millios (1997) e adaptado por Borrmann *et al*, (2008). O LUM é um método de registro baseado em grafos onde os vértices representam as posições e são definidos pelas nuvens de pontos LiDAR 3D já registradas localmente e as arestas (direcionais) representam injunções entre as posições definidas pela associação de correspondências entre duas nuvens de pontos LiDAR 3D.

A primeira nuvem de pontos LiDAR 3D é considerada a origem do sistema de coordenadas global e todas as nuvens são alinhadas simultaneamente a ela. Assim, os parâmetros de transformação são refinados globalmente a partir da distribuição homogênea dos erros aos vértices. Para o estabelecimento das injunções, a estrutura de dados *kd-tree* é usada e pontos são correspondidos com o critério de proximidade, ou seja, aqueles pontos mais próximos entre si. Na Figura 20 uma descrição dos processos relacionados ao Refinamento Global realizado nesta pesquisa.

FIGURA 20 – FLUXOGRAMA DO MÉTODO DE REFINAMENTO GLOBAL APLICADO NESTA PESQUISA COM BASE NO ALGORITMO LUM



FONTE: O autor (2020)

Primeiramente o par de nuvens de pontos LiDAR 3D é submetido a um processo de reamostragem por voxelização, em seguida, como forma de detecção de *loop closures*, para cada centroide da nuvem de pontos LiDAR i é calculada a distância para os centroides das nuvens de pontos LiDAR j , imediatamente anteriores a i , e, caso atenda ao critério de distância mínima Δ entre os centroides das nuvens de pontos LiDAR 3D sendo ajustadas, correspondências entre pontos pertencentes a ambas as nuvens são estabelecidas. Para evitar atribuições equivocadas, é estabelecido um raio de busca (*setRange*) para a atribuição dessas correspondências entre pontos. Por fim, determinadas as inunções para todas as nuvens de pontos LiDAR 3D, os parâmetros de correção dos parâmetros de transformação são determinados a partir das arestas e dos vértices do grafo até que se chegue ao número de iterações χ . A partir desses parâmetros de correção é possível gerar um mapa global consistente.

Cada vértice i do grafo é representado pelo conjunto de parâmetros de transformação obtidos com o método de registro local proposto nesta pesquisa e refinados globalmente pelo algoritmo LUM, como mostra a Matriz Homogênea de transformação definida pela Equação (124).

$$P_{O_i} = \begin{bmatrix} R_i & \vec{t}_i \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}_i \quad (124)$$

sendo R_i a matriz de Rotação e \vec{t}_i o vetor de translação da nuvem de pontos LiDAR 3D NP_i e P_{O_1} definida pela matriz de identidade e o vetor de translação nulo, respectivamente. Assim, para referenciar todas as n nuvens de pontos LiDAR 3D (NP_0, NP_1, \dots, NP_n) ao centroide da NP_0 , considerada a origem do sistema de coordenadas local adotado nesta pesquisa, basta recalcular a posição P_{O_i} por meio do produto de todas as Matrizes Homogêneas de transformação, como mostra a Equação (125):

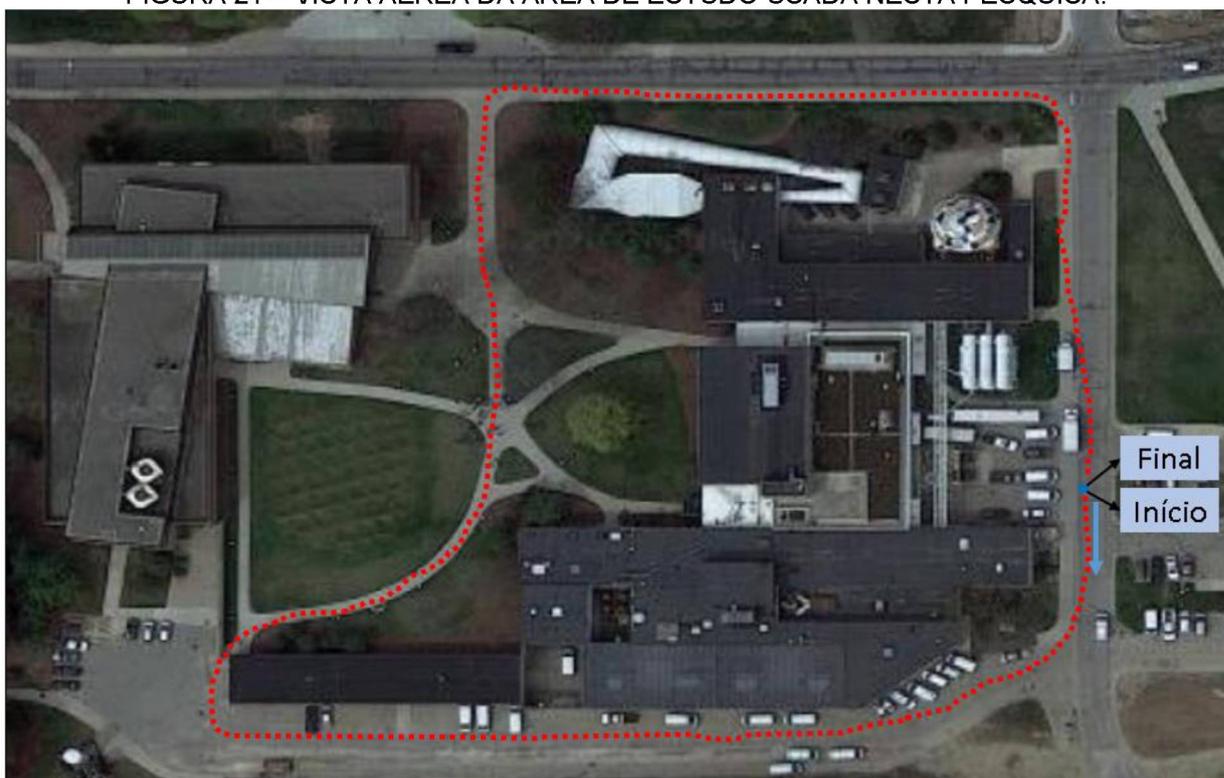
$$P_{O_i} = \prod_{i=1}^n P_{O_i} * P_{O_1} \quad (125)$$

A seguir serão apresentados os ensaios realizados e discutidos os resultados obtidos.

4. EXPERIMENTOS E ANÁLISE DOS RESULTADOS

A área de estudo para a realização desta pesquisa foi selecionada a partir de um conjunto de nuvens de pontos LiDAR 3D obtidas com o sensor Velodyne embarcado em uma plataforma Segway referentes a um levantamento realizado na Universidade de Michigan, Campus Norte, no dia 17 de março de 2012 (<http://robots.engin.umich.edu/nclt/>). A seguir, na Figura 21, uma imagem aérea obtida com o Google Earth do local onde fica a área de estudo usada para realizar essa pesquisa.

FIGURA 21 – VISTA AÉREA DA ÁREA DE ESTUDO USADA NESTA PESQUISA.



Fonte: Adaptada de GOOGLE EARTH.

Na Figura 21 está destacado o início e o final da trajetória percorrida pelo sensor Velodyne embarcado no robô Segway e responsável pela geração das nuvens de pontos LiDAR 3D utilizadas nesta pesquisa. Como pode ser visto na Figura 8 a), o levantamento completo contém sincronizadas às coletas dos dados de odometria de 24.827 nuvens de pontos LiDAR 3D com um total de aproximadamente 700 milhões de pontos. Já a região

do recorte utilizado, como mostra a Figura 21 apresenta 2.700 nuvens de pontos LiDAR 3D com um total de aproximadamente 80 milhões de pontos, entretanto, devido a superabundância de informações, apenas um terço das nuvens de pontos LiDAR 3D da área de estudo foram utilizadas resultando em 900 nuvens de pontos LiDAR 3D com um total de aproximadamente 25 milhões de pontos (após o processo de reamostragem e remoção de pontos espúrios). Os parâmetros utilizados nos processos descritos no Capítulo 3 para a realização do mapeamento tridimensional da área de estudo investigada nesta pesquisa são descritos a seguir na Tabela 2.

TABELA 2 - PARÂMETROS UTILIZADOS PELOS MÉTODOS CITADOS NO CAPÍTULO 3.

Método/Algoritmo	Parâmetro	Descrição	Valor
Remoção de pontos espúrios	<i>setMeanK</i>	vide-texto	30
	<i>setStddevMulThresh</i>	vide-texto	1
Reamostragem	<i>setGridSize</i>	vide-texto	0,1 m
RANSAC	ζ	número máximo de iterações	10
	ξ	número de vizinhos mais próximos	50
	ζ	distância máxima para o plano	0,1 m
Crescimento de Região	τ	referentes à curvatura	0,01
	<i>setMinClusterSize</i>	vide-texto	30
	<i>setMaxClusterSize</i>	vide-texto	10.000
	<i>setNumberOfNeighbours</i>	vide-texto	20
	<i>setSmoothnessThreshold</i>	vide-texto	2°
	<i>setCurvatureThreshold</i>	vide-texto	10
	<i>setMinDist</i>	vide-texto	0,1 m
	<i>setMinPoints</i>	vide-texto	50
Correspondência dos planos	<i>setMinDistCor</i>	vide-texto	15 m
	<i>setMinProb</i>	vide-texto	85%
Registro grosseiro	δ	Limiar da REMQ ponto a ponto	0,5 m
	η	Limiar da REMQ ponto a plano	0,3 m
	ϕ	Limiar inicial da REMQ ponto a ponto que descarta o refinamento	8 cm
	<i>setMaxDist</i>	vide-texto	0,01 m
	<i>setMaxDistCent</i>	vide-texto	10 m
Espaço Canônico	ν	Projeção do espaço canônico definido a partir do limiar	0,5
Espaço Canônico bidimensional	γ	Distância considerada para o estabelecimento do ponto como pertencente a zona de busca	0,3 m
<i>Sliding</i>	<i>setCard</i>	vide-texto	0,7
	<i>setSliding</i>	vide-texto	0,05 m
Refinamento Global	χ	número de iterações	15 m
	<i>setRange</i>	vide-texto	1 m
	Δ	distância mínima entre os centroides	5

FONTE: O autor (2020)

Como forma de demonstrar o potencial do método proposto para a correspondência entre os planos, dois experimentos foram conduzidos com base nos valores da REMQ ponto a plano após o registro com o método de estimativa dos parâmetros de transformação proposto nesta pesquisa. Assim, correspondências foram atribuídas como incorretas se os resultados dos valores da REMQ ponto a plano após o registro ficassem acima de 0,5 metro. O primeiro experimento considerou o alcance entre as nuvens de pontos LiDAR 3D, assim, é aplicado o método de correspondência proposto nesta pesquisa para até 8 nuvens à frente da nuvem de pontos LiDAR 3D de Referência (nuvem 2 → nuvem 0; nuvem 4 → nuvem 0; nuvem 6 → nuvem 0; nuvem 8 → nuvem 0), sendo estes índices de nuvens de pontos LiDAR 3D.

De acordo com o gráfico mostrado na Figura 22, a partir da taxa de correspondências com a nuvem subsequente com 100% de sucesso, essa taxa diminui 4,4% para a quarta nuvem de pontos LiDAR 3D à frente, 6,3% para a sexta e 10% para a oitava. Como pode ser visto no gráfico mostrado na Figura 22, conforme aumenta-se a distância entre as nuvens de pontos LiDAR 3D, aumentam as incidências de correspondências incorretas, devido a forma como são calculadas as funções de probabilidades relacionadas as distribuições espaciais dos centroides dos planos e as mudanças de orientação do sensor.

Um segundo experimento foi conduzido com o intuito de avaliar a atualização da Matriz de Probabilidades P por Inferência Bayesiana. Assim, o algoritmo de correspondência foi aplicado com e sem o uso da atualização da Matriz de Probabilidades em uma Estimativa Máxima *a posteriori*. Os resultados encontrados para o número de correspondências corretas e incorretas segundo essas diretrizes são mostrados na Figura 23.

Como pode ser visto na Figura 23, o não uso da técnica de atualização da Matriz de Probabilidades P usando diretamente Inferência Bayesiana e uma configuração ótima *a priori* resultou em 65 registros locais entre nuvens de pontos LiDAR 3D incorretos, o que representa 7,5% da quantidade total.

FIGURA 22 – AVALIAÇÃO DA DESEMPENHO DAS CORRESPONDÊNCIAS ENTRE PLANOS PRESENTES EM ATÉ 8 NUVENS À FRENTE.

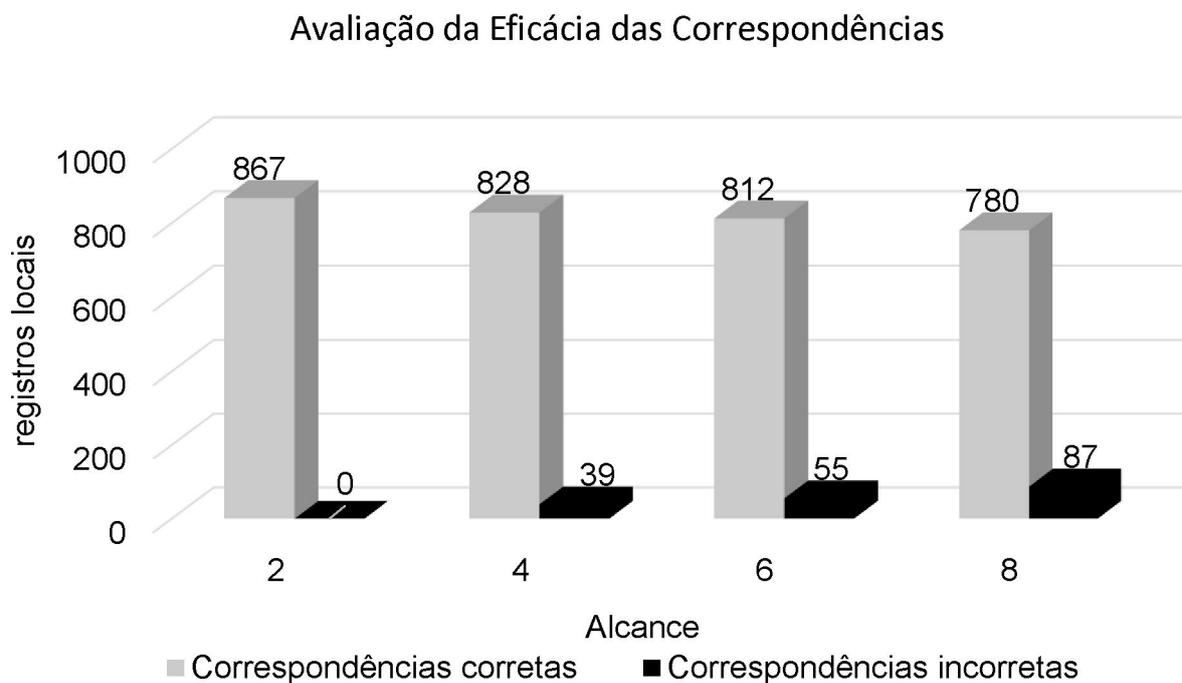
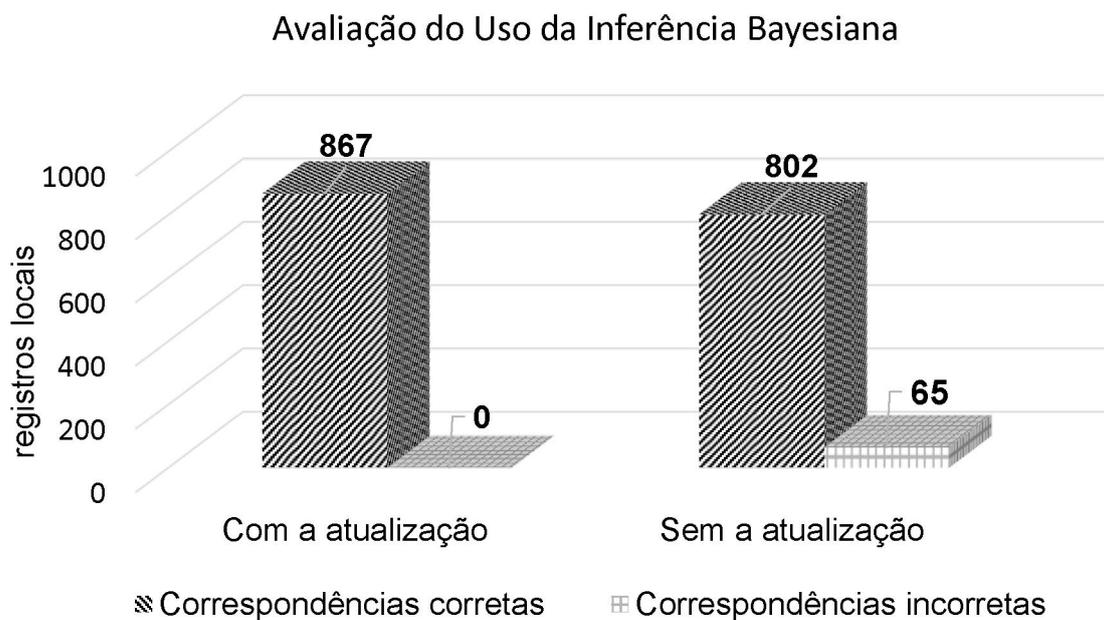


FIGURA 23 – AVALIAÇÃO DO USO DA INFERÊNCIA BAYESIANA NO PROCESSO DE CORRESPONDÊNCIA ENTRE PLANOS.



Para avaliar quantitativamente o registro local, nesta pesquisa foi calculado a REMQ ponto a plano, como mostra a Equação (126).

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}} \quad (126)$$

sendo, x_i os erros ponto a plano calculados, \bar{x} o valor médio do vetor de erros e n o número de pontos presentes nos planos na nuvem de pontos LiDAR 3D de Pesquisa.

Primeiramente foram calculados os valores da REMQ ponto a plano diante as seguintes condições:

- I) Antes do registro local.
- II) Após a aplicação dos parâmetros de transformação às nuvens de pontos LiDAR 3D de Pesquisa obtidos com o método de registro grosseiro proposto nesta pesquisa.
- III) Após a aplicação dos parâmetros de transformação às nuvens de pontos LiDAR 3D de Pesquisa obtidos com o método de registro grosseiro-para-fino.

Para facilitar a visualização dos resultados da REMQ ponto a plano antes e após a realização do registro local proposto e nos comparativos com os algoritmo ICP e NDT proposto neste trabalho, o trecho completo com as 900 nuvens de pontos LiDAR 3D foi dividido em duas partes, apenas para fins de visualização. Portanto, na Figura 24 são apresentados os valores da REMQ para cada registro local par a par realizado antes (preto), com o método grosseiro (vermelho) e com o método grosseiro-para-fino (azul), entre as nuvens de pontos LiDAR 3D para a 1ª parte do trajeto (primeira metade do trajeto completo).

Já na Figura 25 a 2ª parte do trajeto (segunda metade do trajeto completo) com as mesmas configurações. Percebe-se que o eixo das ordenadas foi limitado a 1 metro visando dar maior destaque aos resultados da REMQ após o registro local proposto, entretanto, como pode ser visto, por diversas vezes esses valores da REMQ anteriores à aplicação dos parâmetros de transformação extrapolam esse limite com casos chegando a mais de 5 metros (.

FIGURA 24 – VALORES DA REMQ PONTO A PLANO ANTES E APÓS O REGISTRO LOCAL PARA O 1º TRECHO.

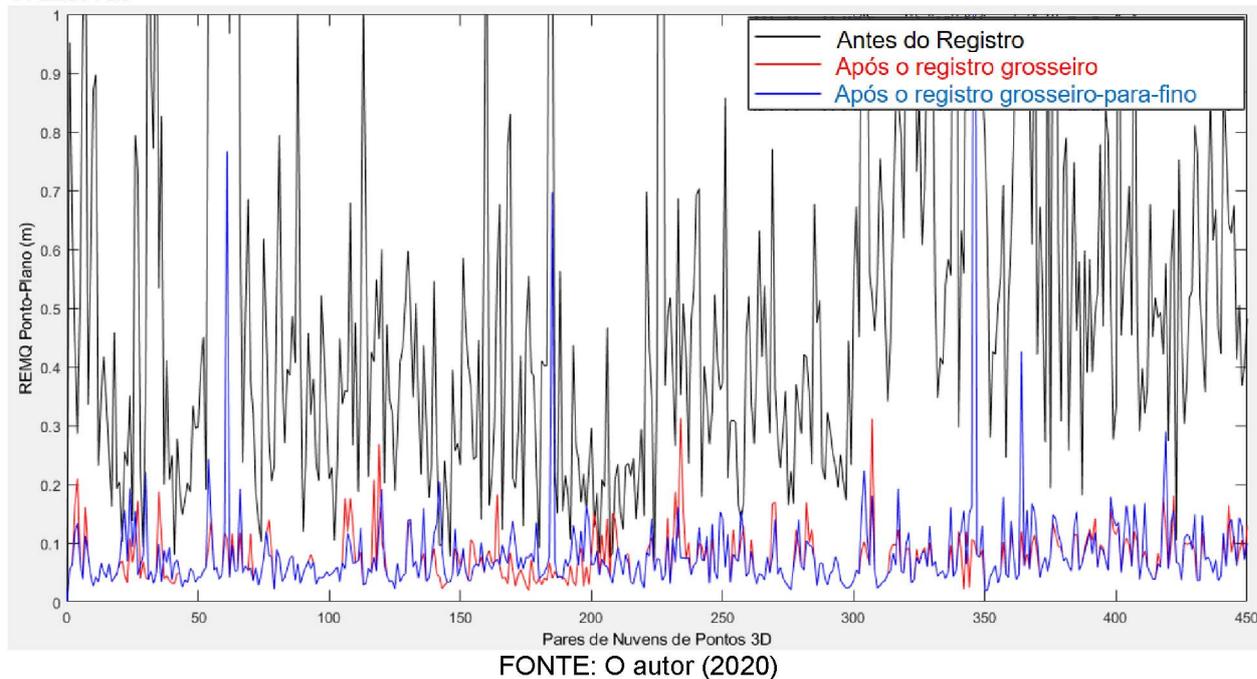
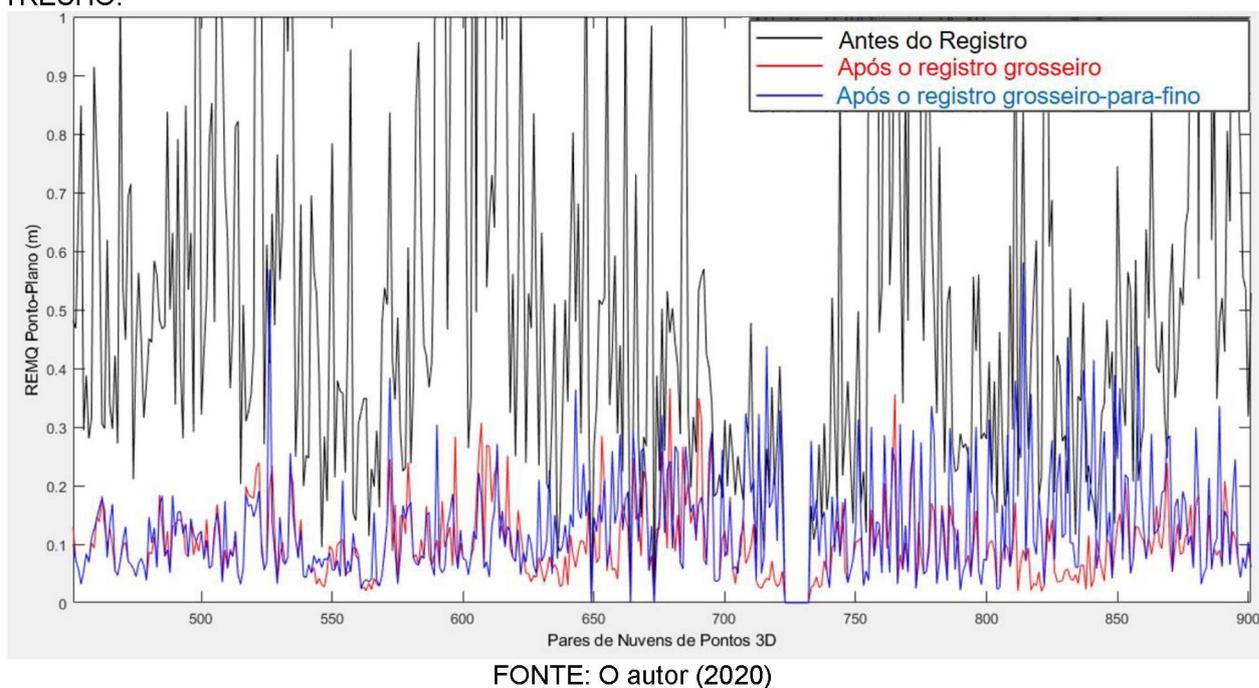


FIGURA 25 - VALORES DA REMQ PONTO A PLANO ANTES E APÓS O REGISTRO LOCAL PARA O 2º TRECHO.



Analisando os dois gráficos mostrados na Figura 24 e na Figura 25 quando comparado ao erro antes da aplicação dos parâmetros de transformação, percebe-se

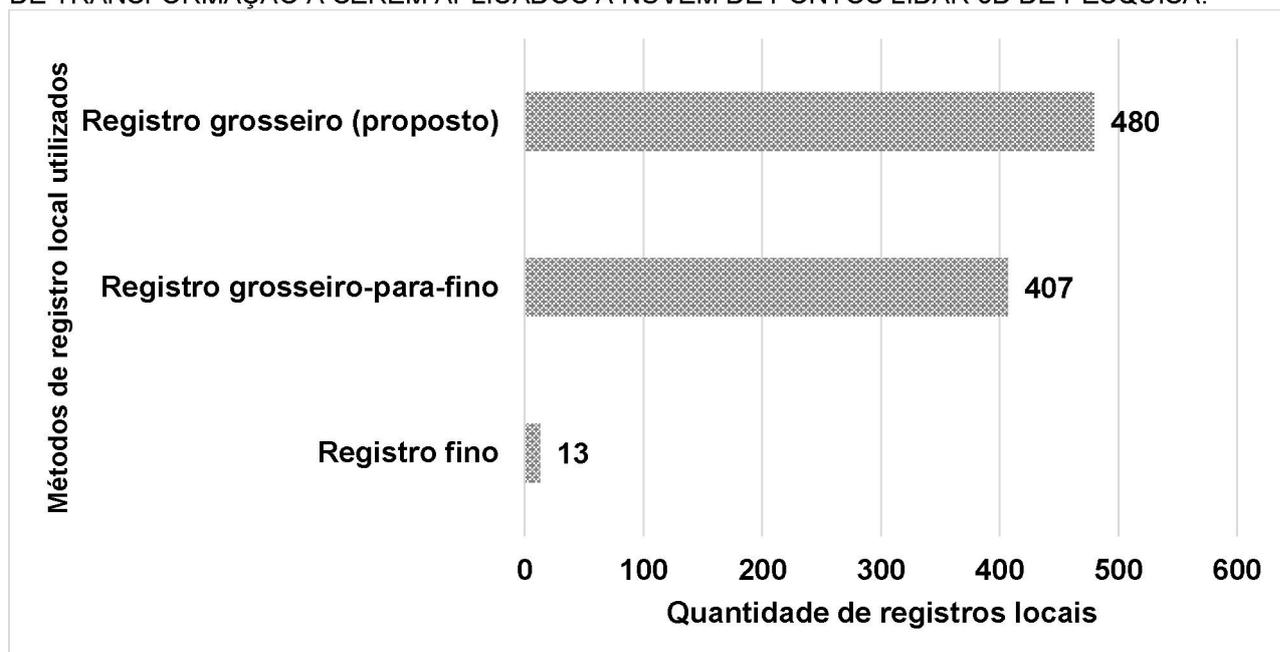
uma redução significativa dos valores da REMQ ponto a plano após a aplicação do método de registro grosseiro ou grosseiro-para-fino proposto nesta pesquisa, sendo uma diminuição de aproximadamente 85% para a média e 88% para o desvio padrão.

Para alguns casos, o valor da REMQ ponto a plano para o método grosseiro-para-fino (azul) nas Figuras 24 e 25 ultrapassam os valores do método de registro grosseiro (vermelho). Assim, os parâmetros de transformação obtidos com o método de registro grosseiro-para-fino são substituídos pelos resultados do método de registro grosseiro proposto nesta pesquisa. Além disso, caso os valores da REMQ ponto a plano após o método de registro grosseiro sejam menores que o limiar ϕ usado (ver Tabela 2), o refinamento é descartado, reduzindo assim o custo computacional da aplicação do método iterativo ICP. Para fins de visualização nesses casos, nas Figuras 24 e 25 os valores da REMQ do método de registro grosseiro-para-fino são atribuídos com o valor do método de registro grosseiro proposto. Como pode ser visto no gráfico mostrado na Figura 26, os parâmetros de transformação obtidos pelo método de registro grosseiro proposto nesta pesquisa foram os mais utilizados devido a ocorrências desses dois eventos citados anteriormente.

Como pode ser visto na Figura 26, o método de registro grosseiro foi utilizado em 54% dos pares de nuvens de pontos LiDAR 3D, enquanto que o refinamento ocorreu em 45% dos registros locais e a aplicação apenas do registro fino para a determinação dos parâmetros em 1,5%, devido a ausência de planos correspondentes nas nuvens de pontos LiDAR 3D de Pesquisa ou de Referência. Portanto, em mais da metade dos registros locais o valor encontrado para a REMQ ponto a plano após o registro grosseiro proposto nesta pesquisa ou ficou substancialmente abaixo do valor encontrado pelo método grosseiro-para-fino ou abaixo do limiar ϕ .

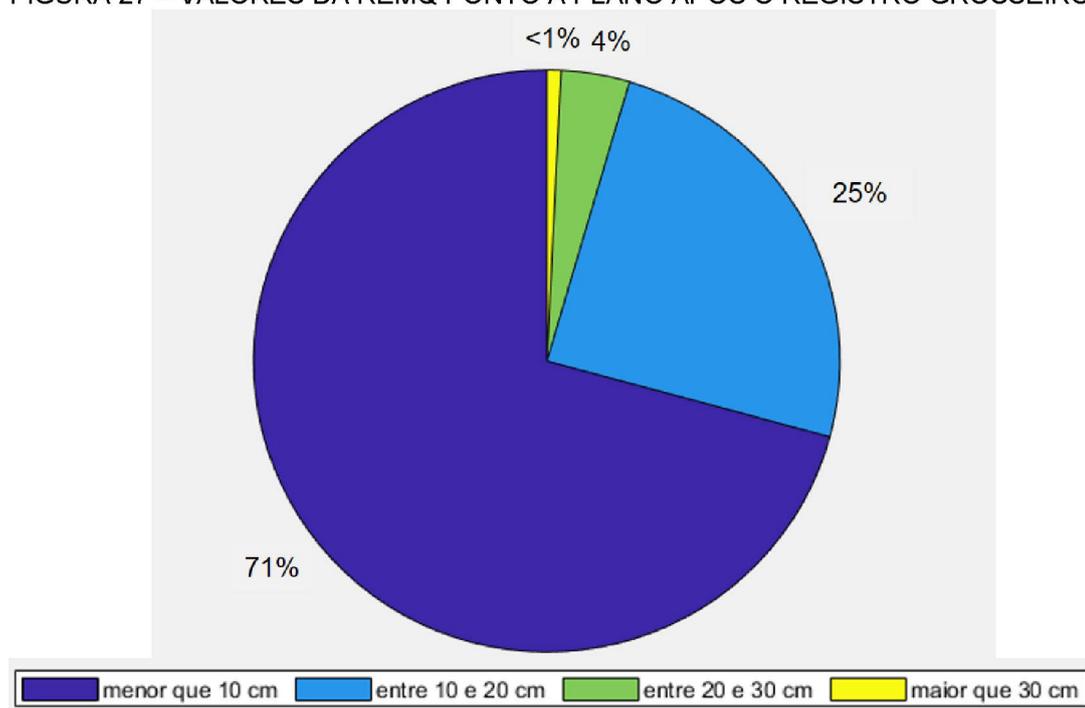
Para demonstrar o potencial do método de registro local proposto nesta pesquisa, nas Figuras 27 e 28 são apresentados dois gráficos com as porcentagens dos valores da REMQ calculados, em diferentes intervalos, para os métodos grosseiro e grosseiro-para-fino. Como pode ser visto nos gráficos, em ambos os casos mais de 95% dos valores da REMQ ponto a plano ficaram abaixo de 20 cm. Além disso, com o método grosseiro-para-fino todos os valores da REMQ ponto a plano ficaram abaixo dos 30 cm.

FIGURA 26 – FREQUÊNCIAS DOS MÉTODOS ESCOLHIDOS PARA DETERMINAR OS PARÂMETROS DE TRANSFORMAÇÃO A SEREM APLICADOS À NUVEM DE PONTOS LIDAR 3D DE PESQUISA.



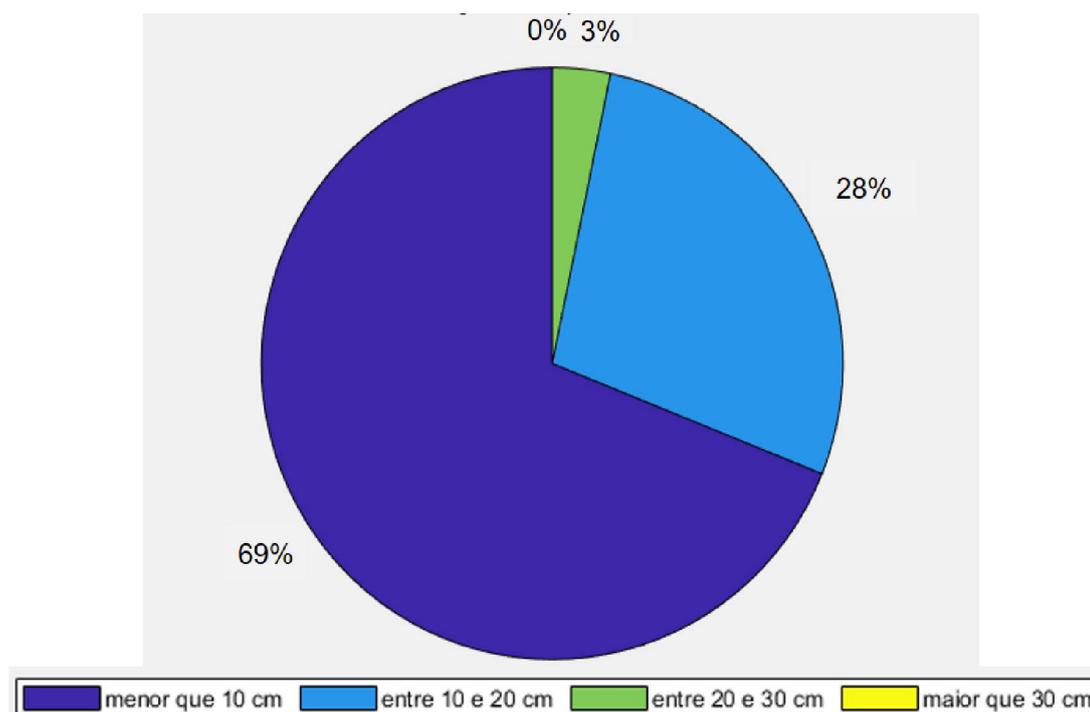
FONTE: O autor (2020)

FIGURA 27 – VALORES DA REMQ PONTO A PLANO APÓS O REGISTRO GROSSEIRO.



FONTE: O autor (2020)

FIGURA 28 - VALORES DA REMQ PONTO A PLANO APÓS O MÉTODO DE REGISTRO GROSSEIRO-PARA-FINO.



FONTE: O autor (2020)

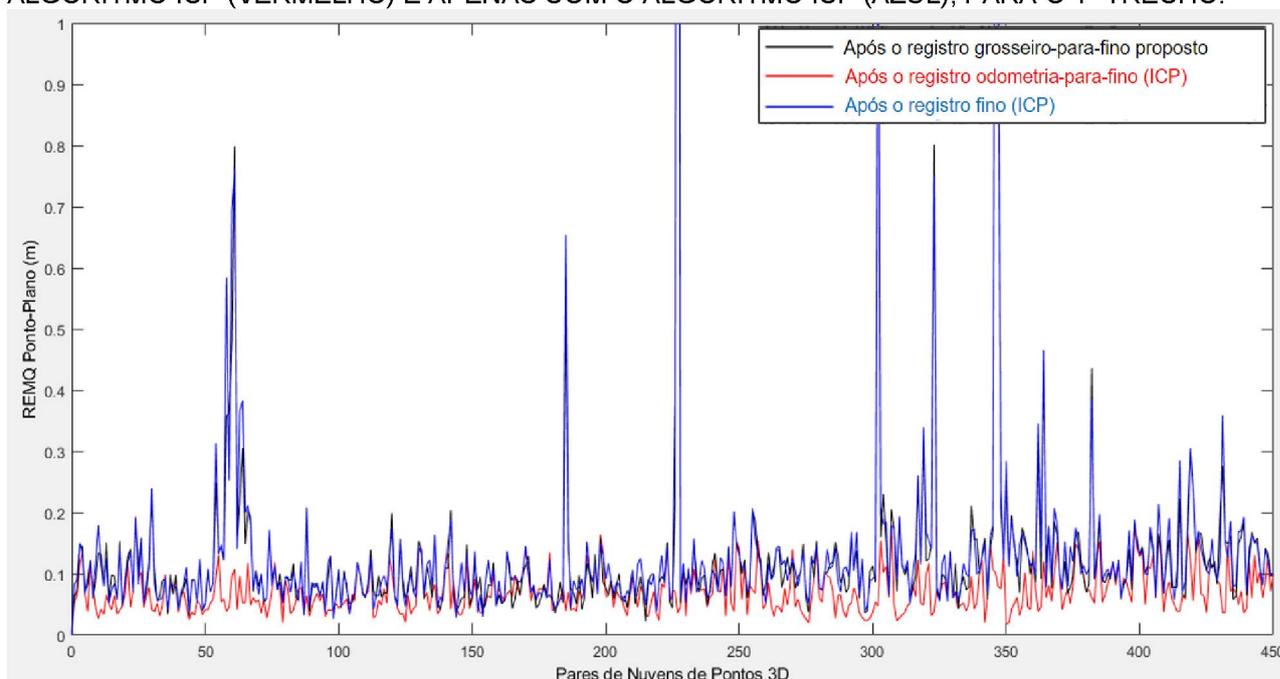
Para fins de comparação com o método de registro local proposto nesta pesquisa, foram realizados experimentos para a mesma área de estudo com os seguintes cenários:

- 1) Após a aplicação dos parâmetros de transformação obtidos com o método de registro local proposto.
- 2) Após a aplicação dos parâmetros de transformação às nuvens de pontos LiDAR 3D de Pesquisa obtidos com os dados de odometria associados a cada nuvem de pontos LiDAR 3D e refinados com o algoritmo ICP.
- 3) Após a aplicação dos parâmetros de transformação às nuvens de pontos LiDAR 3D de Pesquisa obtidos com os dados de odometria associados a cada nuvem de pontos LiDAR 3D e refinados com o algoritmo NDT.
- 4) Após a aplicação dos parâmetros de transformação às nuvens de pontos LiDAR 3D de Pesquisa obtidos com o algoritmo ICP.
- 5) Após a aplicação dos parâmetros de transformação às nuvens de pontos LiDAR 3D de Pesquisa obtidos com o algoritmo NDT.

Os fatores a serem considerados para avaliar qualitativamente o registro local foram baseados nos valores da REMQ ponto a plano após a aplicação dos parâmetros de transformação nos diferentes cenários aos pontos pertencentes aos planos da nuvem de pontos LiDAR 3D de Pesquisa, e, o custo computacional a partir da medição do tempo de todo o processo envolvido em cada cenário.

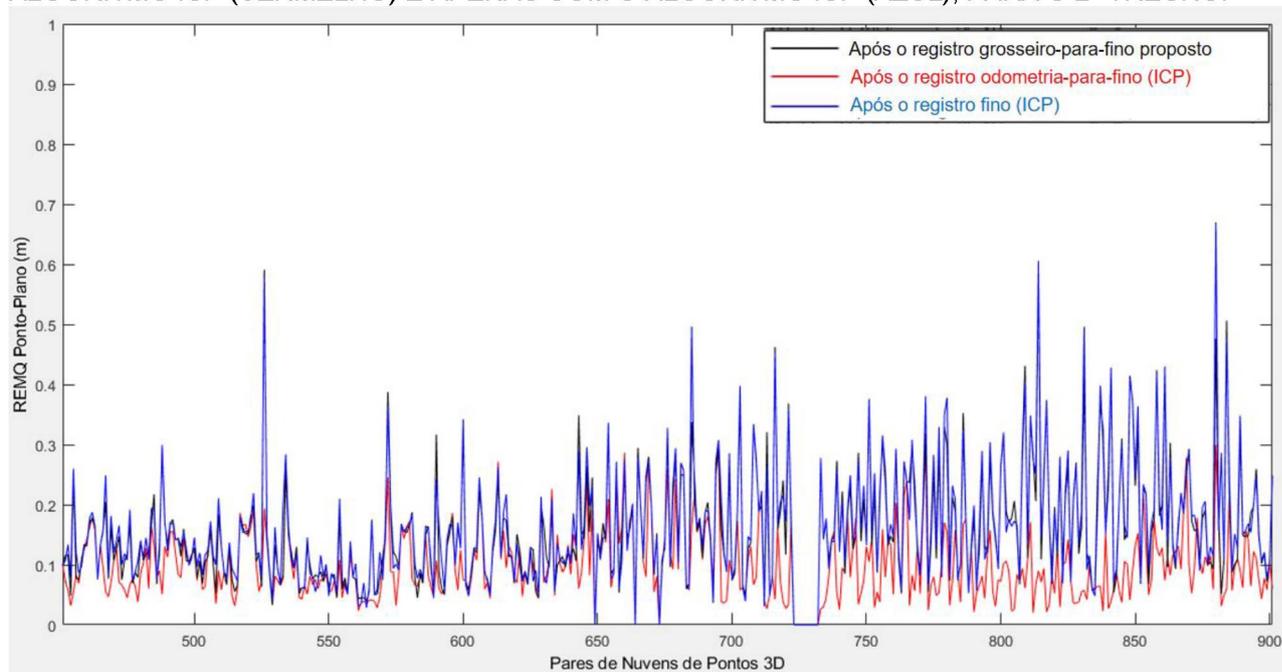
Na Figura 29 e na Figura 30 são mostrados os resultados para a REMQ ponto a plano obtidos utilizando o método de registro local proposto nesta pesquisa e utilizando os dados de odometria refinados com o algoritmo ICP, além, dos obtidos utilizando apenas o algoritmo ICP para o registro local. Novamente, esses experimentos foram separados em dois trechos para facilitar a visualização dos resultados. Na Figura 29 é mostrado o primeiro trecho com os 450 primeiros registros locais, já na Figura 30 é mostrada a 2ª parte do trajeto com as mesmas configurações. Para os cenários utilizando algoritmo NDT, os valores da REMQ ponto a plano calculados são mostrados nas Figuras 31 e 32, para o primeiro e segundo trecho, respectivamente, segundo as mesmas configurações.

FIGURA 29 - VALORES DA REMQ PONTO A PLANO OBTIDOS COM O MÉTODO DE REGISTRO LOCAL PROPOSTO NESTA PESQUISA (PRETO), COM OS DADOS DE ODOMETRIA REFINADOS COM O ALGORITMO ICP (VERMELHO) E APENAS COM O ALGORITMO ICP (AZUL), PARA O 1º TRECHO.



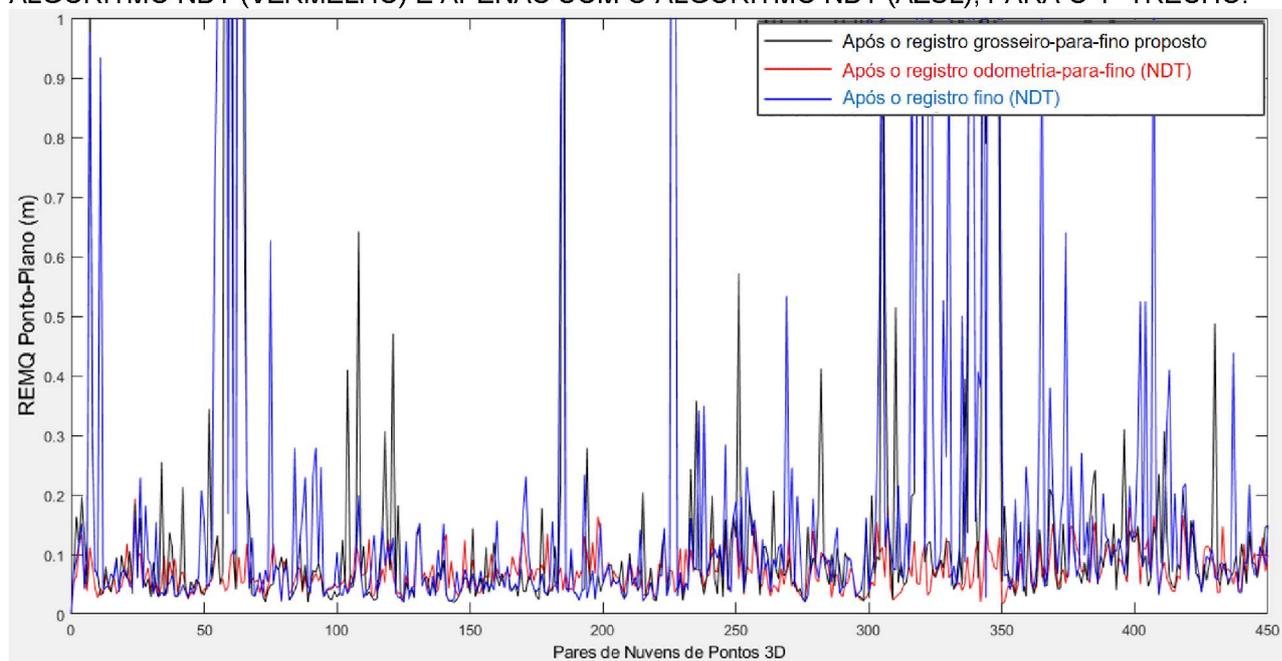
FONTE: O autor (2020)

FIGURA 30 - VALORES DA REMQ PONTO A PLANO OBTIDOS COM O MÉTODO DE REGISTRO LOCAL PROPOSTO NESTA PESQUISA (PRETO), COM OS DADOS DE ODOMETRIA REFINADOS COM O ALGORITMO ICP (VERMELHO) E APENAS COM O ALGORITMO ICP (AZUL), PARA O 2º TRECHO.



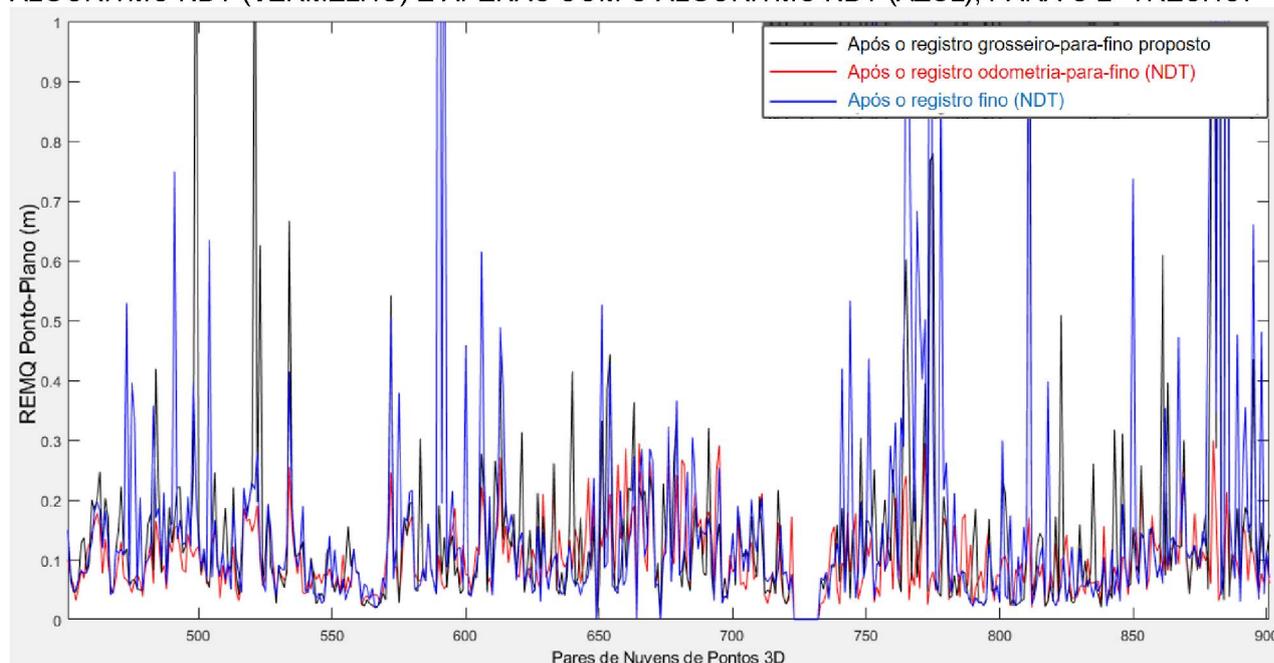
FONTE: O autor (2020)

FIGURA 31 - VALORES DA REMQ PONTO A PLANO OBTIDOS COM O MÉTODO DE REGISTRO LOCAL PROPOSTO NESTA PESQUISA (PRETO), COM OS DADOS DE ODOMETRIA REFINADOS COM O ALGORITMO NDT (VERMELHO) E APENAS COM O ALGORITMO NDT (AZUL), PARA O 1º TRECHO.



FONTE: O autor (2020)

FIGURA 32 - VALORES DA REMQ PONTO A PLANO OBTIDOS COM O MÉTODO DE REGISTRO LOCAL PROPOSTO NESTA PESQUISA (PRETO), COM OS DADOS DE ODOMETRIA REFINADOS COM O ALGORITMO NDT (VERMELHO) E APENAS COM O ALGORITMO NDT (AZUL), PARA O 2º TRECHO.

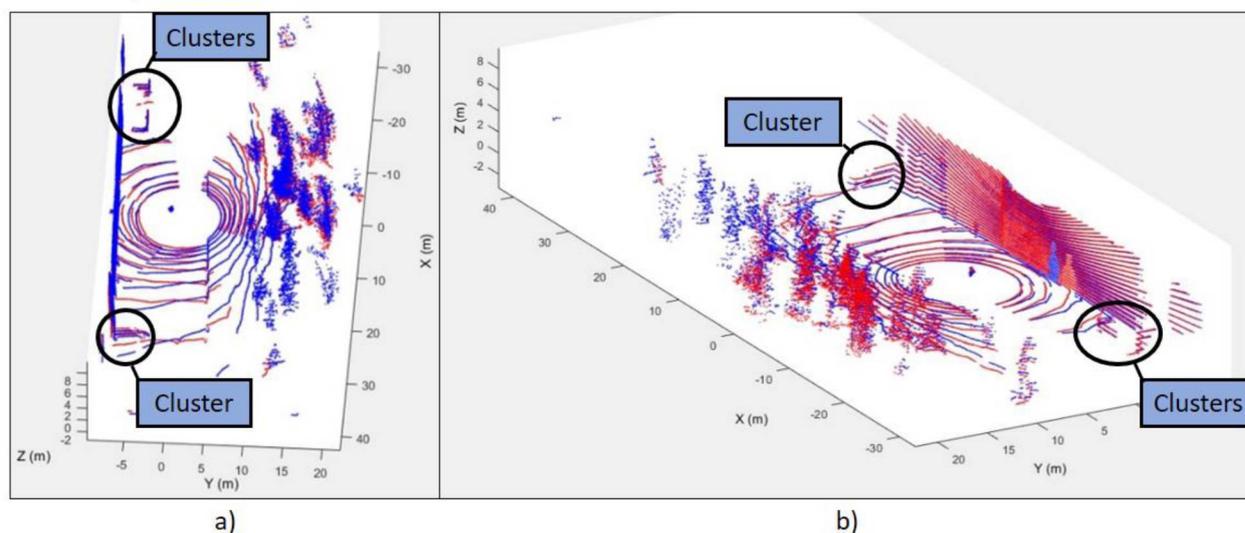


FONTE: O autor (2020)

Como pode ser visto nas Figuras 29 e 30, tanto quando comparado ao uso de dados de odometria como parâmetros de transformação grosseiros seguidos de um refinamento desses parâmetros com o algoritmo ICP, quanto utilizando apenas o ICP, na imensa maioria dos registros locais, os valores da REMQ ponto a plano obtidos com o método proposto nesta pesquisa apresenta valores inferiores, além disso, em alguns casos, ambos os valores da REMQ ponto a plano com o algoritmo ICP, com ou sem odometria, apresentaram valores destoantes, por diversas vezes passando dos 0,5 metros. Isso ocorre devido a alta dependência do algoritmo de sobreposição máxima das nuvens de pontos LiDAR 3D, o que não é garantido durante um levantamento urbano sujeito às inúmeras adversidade já discutidas anteriormente.

A Figura 33 ilustra duas imagens do mesmo par de nuvens de pontos LiDAR 3D de Referência (vermelho) e Pesquisa (azul), em diferentes perspectivas, após o alinhamento utilizando o método de registro local com um Espaço Canônico bidimensional por meio de *clusters* proposto nesta pesquisa, com destaque nas imagens para os *clusters*.

FIGURA 33 – IMAGEM EM DUAS PERSPECTIVAS DE UM PAR DE NUVENS DE PONTOS LIDAR 3D APÓS A APLICAÇÃO DO MÉTODO DE REGISTRO GROSSEIRO PROPOSTO NESTA PESQUISA.COM UM ESPAÇO CANÔNICO BIDIMENSIONAL GERADO.

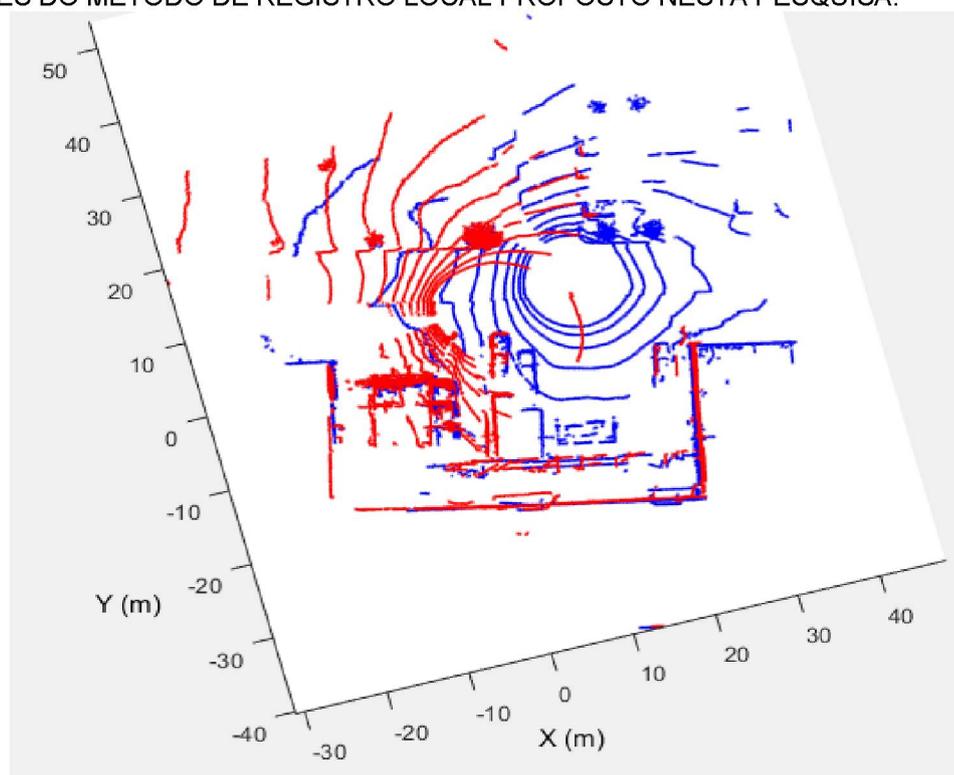


FONTE: O autor (2020)

Para o registro local mostrado na Figura 33, o valor calculado da REMQ ponto a plano com o método de registro grosseiro proposto nesta pesquisa foi de 8,2 cm, já o valor encontrado após o método grosseiro-para-fino foi de 83 cm, enquanto que o valor calculado após a aplicação apenas do algoritmo ICP foi de 48 cm. Neste caso, os parâmetros do registro grosseiro substituem os do grosseiro-para-fino para a aplicação à nuvem de pontos LiDAR 3D. Outro exemplo da superioridade do método de registro grosseiro sobre o algoritmo ICP, amplamente utilizado pela comunidade científica, é mostrado na Figura 34 vista de topo.

Como pode ser visto na Figura 34, a taxa de sobreposição entre as nuvens de pontos LiDAR 3D de Referência (vermelho) e de Pesquisa (azul) é um pouco maior que a metade, assim, após a aplicação apenas do algoritmo ICP a porcentagem de redução da REMQ ponto a plano comparado com o valor anterior a aplicação dos parâmetros de transformação foi de 38% para o algoritmo ICP, 34% para o método grosseiro-para-fino, e, 88% para o método grosseiro proposto nesta pesquisa com o Espaço Canônico tridimensional.

FIGURA 34 – IMAGEM VISTA DE CIMA DE UM PAR DE NUVENS DE PONTOS LIDAR 3D APÓS A APLICAÇÕES DO MÉTODO DE REGISTRO LOCAL PROPOSTO NESTA PESQUISA.

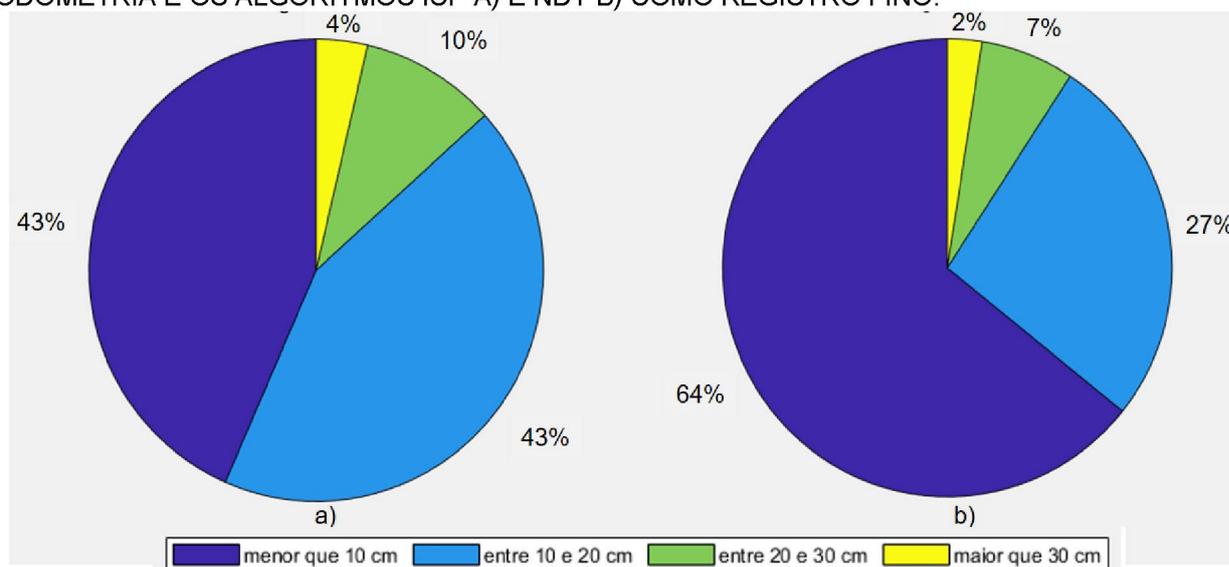


FONTE: O autor (2020)

Em relação ao algoritmo NDT com e sem o uso de dados de odometria como parâmetros de transformação iniciais, o método proposto nesta pesquisa se sobressai ainda mais comparado aos cenários com o ICP, como pode ser visto nas Figuras 31 e 32, a qual, com maior frequência, aparecem valores da REMQ ponto a plano extremos, enquanto que o método de registro local proposto nesta pesquisa apresenta-se estável em todos os registros.

Para uma melhor visualização dos valores da REMQ ponto a plano utilizando os métodos ICP e NDT com os dados de odometria como parâmetros de transformação iniciais, os mesmo intervalos de valores das Figuras 27 e 28 foram aplicados e as porcentagens encontradas são mostradas na Figura 35. Comparando com os gráficos mostrados nas Figuras 27 e 28 com os gráficos da Figura 35, percebe-se um aumento substancial de valores na faixa acima dos 20 cm da REMQ ponto a plano com os algoritmos ICP a) e NDT b), de 14% e 9%, respectivamente. Enquanto que com o método de registro grosseiro proposto foram encontrados menos de 5%, e com o grosseiro-para-fino apenas 3%.

FIGURA 35 - VALORES DA REMQ PONTO A PLANO APÓS O REGISTRO UTILIZANDO DADOS DE ODOMETRIA E OS ALGORITMOS ICP A) E NDT B) COMO REGISTRO FINO.



FONTE: O autor (2020)

Na Tabela 3 são apresentadas as medidas de tendência central e de dispersão dos valores da REMQ ponto a plano considerando todos os cenários (I, II, III, IV e V).

TABELA 3 – MEDIDAS DE TENDÊNCIA CENTRAL E VARIABILIDADE DOS RESULTADOS DA REMQ PONTO A PLANO ANTES E APÓS OS CENÁRIOS CONSIDERADOS NESTA PESQUISA.

Métodos/Algoritmos	Média (m)	Desvio Padrão (m)
Antes do registro	0,5682	0,5554
(II) Odometria + NDT	0,1762	0,418
(IV) NDT	0,2127	0,4944
(I) Odometria + ICP	0,1492	0,1641
(III) ICP	0,1553	0,1732
Grosseiro (proposto)	0,0904	0,0695
Grosseiro-para-fino (proposto)	0,0915	0,0658

FONTE: O autor (2020)

Como pode ser visto na Tabela 3, quando comparado ao valor médio e do desvio padrão antes da aplicação dos parâmetros de transformação, tanto os valores obtidos com o método de registro grosseiro quanto o grosseiro-para-fino, produzem resultados consideravelmente melhores para essas medidas de tendência central e variabilidade. Além disso, quando comparadas as médias dos resultados dos outros cenários, muitos valores são próximos a metade, e até 8 vezes menores quanto ao desvio padrão. A seguir são analisados os tempos de processamento em um comparativo com os algoritmos ICP e NDT.

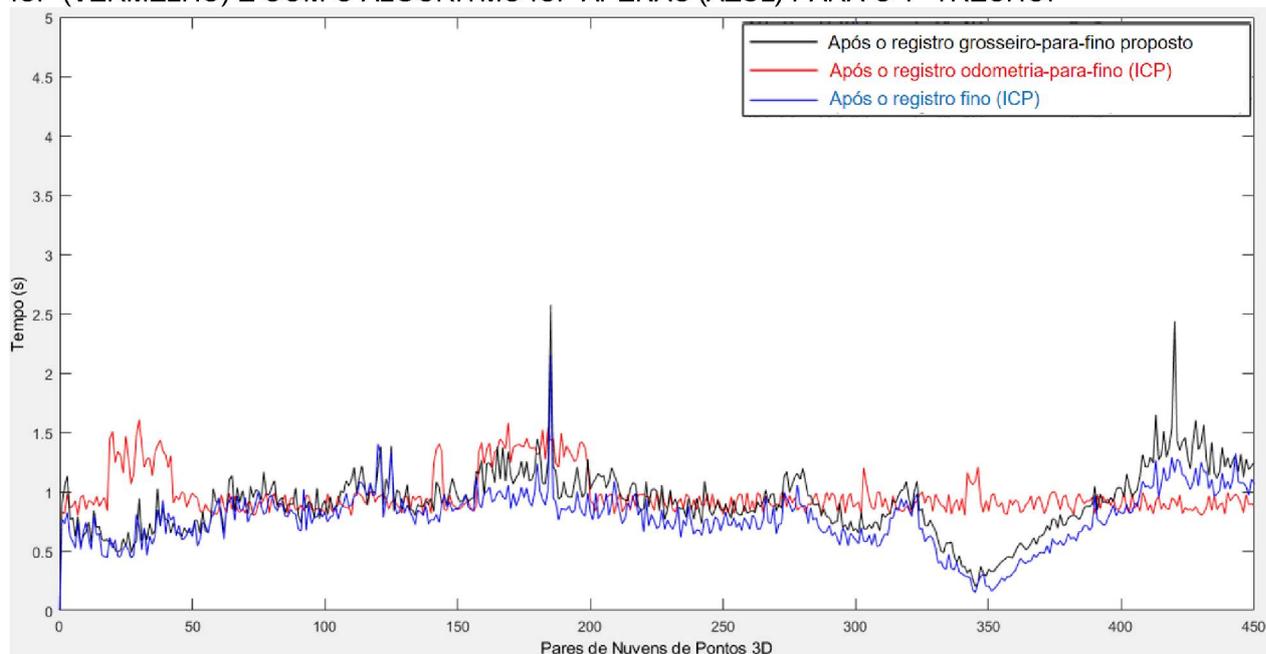
Analisando o comportamento das medições de tempo para a realização dos registros locais entre as nuvens de pontos LiDAR 3D nas Figuras 36 e 37 e as medidas de tendência central mostradas na Tabela 4 entre o método proposto nesta pesquisa com o algoritmo ICP, com e sem o uso de odometria, percebe-se uma certa similaridade entre os valores, entretanto, a dispersão apresentando pelo método proposto é menor. Já o algoritmo NDT apresenta-se mais custoso computacionalmente comparado ao método proposto neste trabalho, como mostram as Figuras 38 e 39, e que pode ser confirmado na Tabela 4.

TABELA 4 – MEDIÇÕES DE TEMPO DOS MÉTODOS DE REGISTRO GROSSEIRO E GROSSEIRO-PARA-FINO E OS ALGORITMOS USADOS COMO COMPARATIVO, ICP E NDT.

Métodos/Algoritmos	média (s)	desvio padrão (s)
Filtragem, Segmentação e Extração de planos	0,90	0,05
Odometria + NDT	2,51	1,08
Odometria + ICP	1,00	0,33
Grosseiro	0,08	0,16
Grosseiro-para-fino	0,15	0,21

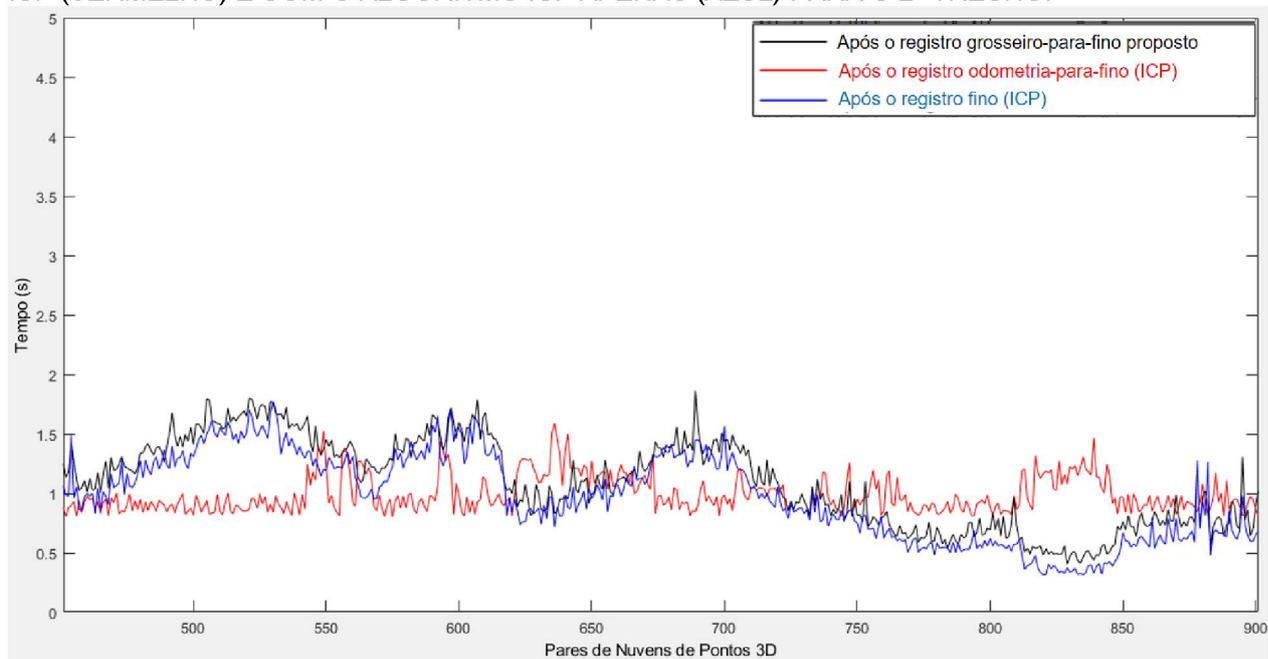
FONTE: O autor (2020)

FIGURA 36 – MEDIÇÕES DE TEMPO PARA OS REGISTRO LOCAIS COM O MÉTODO PROPOSTO GROSSEIRO-PARA-FINO (PRETO), COM OS DADOS DE ODOMETRIA ASSOCIADOS AO ALGORITMO ICP (VERMELHO) E COM O ALGORITMO ICP APENAS (AZUL) PARA O 1º TRECHO.



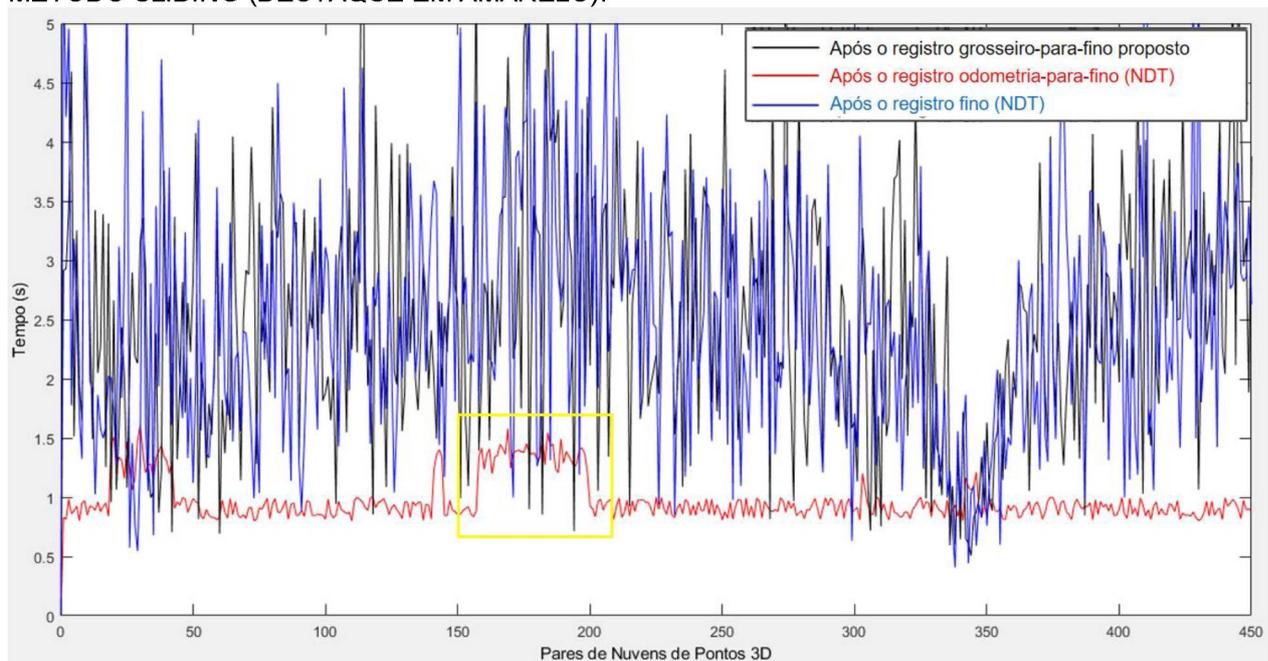
FONTE: O autor (2020)

FIGURA 37 - MEDIÇÕES DE TEMPO PARA OS REGISTRO LOCAIS COM O MÉTODO PROPOSTO GROSSEIRO-PARA-FINO (PRETO), COM OS DADOS DE ODOMETRIA ASSOCIADOS AO ALGORITMO ICP (VERMELHO) E COM O ALGORITMO ICP APENAS (AZUL) PARA O 2º TRECHO.



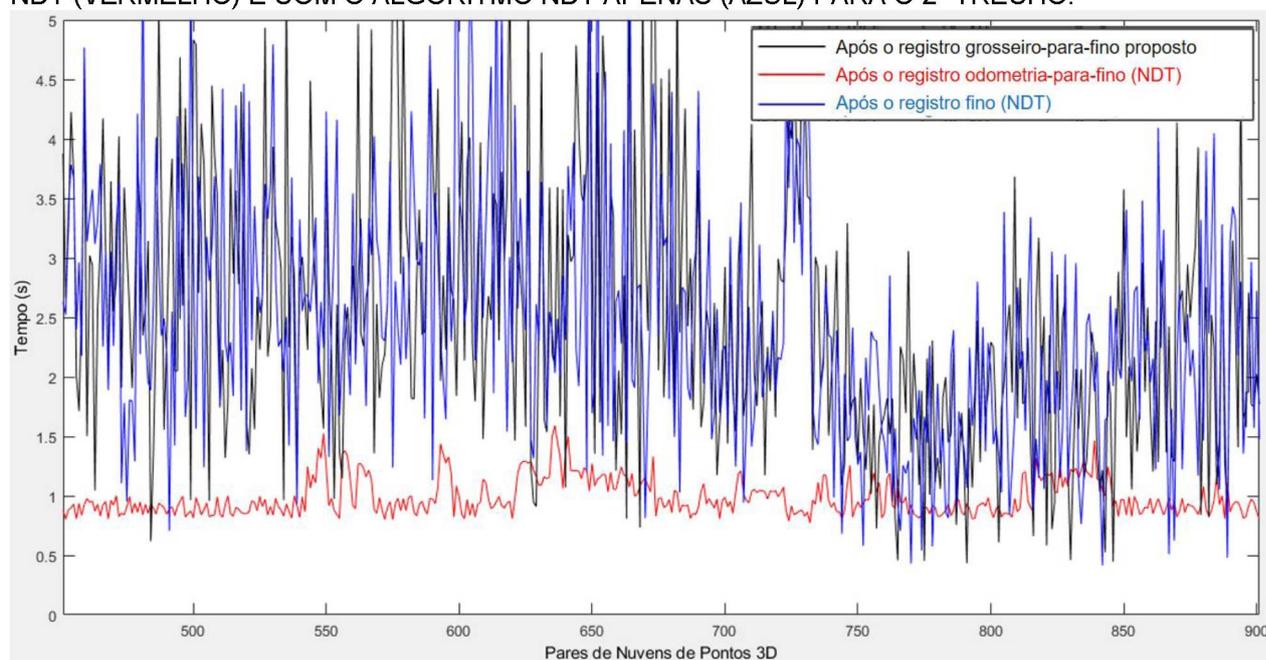
FONTA: O autor (2020)

FIGURA 38 - MEDIÇÕES DE TEMPO PARA OS REGISTRO LOCAIS COM O MÉTODO PROPOSTO GROSSEIRO-PARA-FINO (PRETO), COM OS DADOS DE ODOMETRIA ASSOCIADOS AO ALGORITMO NDT (VERMELHO) E COM O ALGORITMO NDT APENAS (AZUL) PARA O 1º TRECHO. EXECUÇÃO DO MÉTODO SLIDING (DESTAQUE EM AMARELO).



FONTA: O autor (2020)

FIGURA 39 - MEDIÇÕES DE TEMPO PARA OS REGISTRO LOCAIS COM O MÉTODO PROPOSTO GROSSEIRO-PARA-FINO (PRETO), COM OS DADOS DE ODOMETRIA ASSOCIADOS AO ALGORITMO NDT (VERMELHO) E COM O ALGORITMO NDT APENAS (AZUL) PARA O 2º TRECHO.



FONTE: O autor (2020)

Como pode ser visto na Figura 38 há um destaque em amarelo para um intervalo nos resultados do método de registro grosseiro-para-fino proposto com medições levemente alteradas. Neste trecho não foi detectado um Espaço Canônico tridimensional nem a presença de *clusters*, portanto, o método de *Sliding* foi aplicado. Assim, percebe-se um aumento médio de 0,5 segundo, porém, os valores da REMQ ponto a plano calculados para estes registros locais tiveram uma média abaixo dos 10 cm. Como pode ser visto na Tabela 4, a média do tempo de consumo dos processos inerentes ao método de registro grosseiro proposto nesta pesquisa que incluem: Cálculo dos parâmetros, remoção de ambiguidades e correspondência, de planos; Estimativa dos parâmetros de transformação utilizando a álgebra de Dual Quaternions com um Espaço Canônico tridimensional ou bidimensional, incluindo *clusters* e o *Sliding*; ficou abaixo de um décimo de segundo.

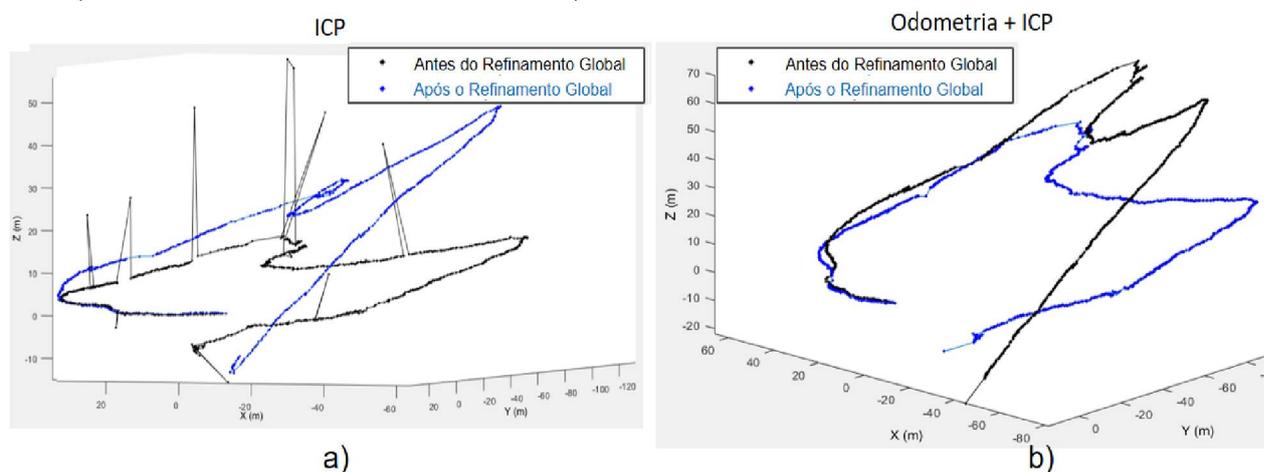
Na Figura 25 percebe-se erros nulos no intervalo entre os registros locais 700 e 750. Isso ocorre pois nessas nuvens de pontos LiDAR 3D não foram detectados planos, portanto, não foi possível aplicar a métrica da REMQ ponto a plano. Para esses pares, apenas o registro fino foi aplicado e posteriormente calculado a REMQ ponto a ponto a

partir do estabelecimento do ponto mais próximo entre as nuvens de pontos LiDAR 3D de Pesquisa e Referência utilizando a estrutura de dados *kd-tree*. Comparado com o erro antes da aplicação dos parâmetros de transformação, houve uma redução de aproximadamente 40% para a média e de mais de 50% para o desvio padrão, com o algoritmo ICP, e, os mesmo 40% para a média, mas praticamente nulo para o desvio padrão, com o algoritmo NDT.

Após a aplicação dos parâmetros de transformação obtidos no registro local às 900 nuvens de pontos LiDAR 3D de Pesquisa por meio da Equação (125), foram obtidas as trajetórias, com o algoritmo ICP (com e sem o uso da odometria), com o algoritmo NDT (*idem*) e com o método de registro local proposto nesta pesquisa, mostradas nas Figuras 40, 41 e 42, respectivamente.

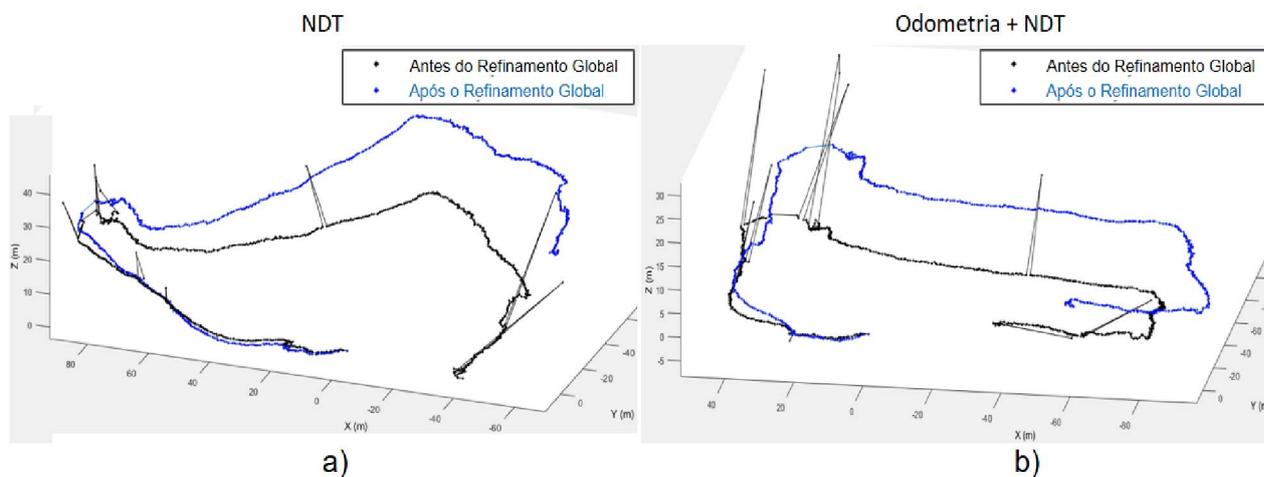
A partir das trajetórias foram geradas as nuvens de pontos LiDAR 3D globais, com o algoritmo ICP mostradas na Figura 44 e NDT mostradas na Figura 45, ambas com os dados de odometria como parâmetros iniciais. Já com o método proposto, as nuvens são mostradas nas Figuras 46 e 47. Nessas imagens, o terreno está em amarelo e as estruturas acima dele estão em azul. Como nesta pesquisa não estão sendo utilizadas informações de trajetória verdadeiras para avaliar a acurácia absoluta da nuvem de pontos LiDAR 3D global, foram calculados os erros de fechamento dos parâmetros de transformação entre a última nuvem de pontos LiDAR 3D e a primeira, portanto, é assumido que o robô retorna a exata posição e orientação de onde partiu.

FIGURA 40 - TRAJETÓRIA OBTIDA COM OS DADOS DE ODOMETRIA ASSOCIADOS AO ALGORITMO ICP B) E COM O ALGORITMO ICP APENAS A).



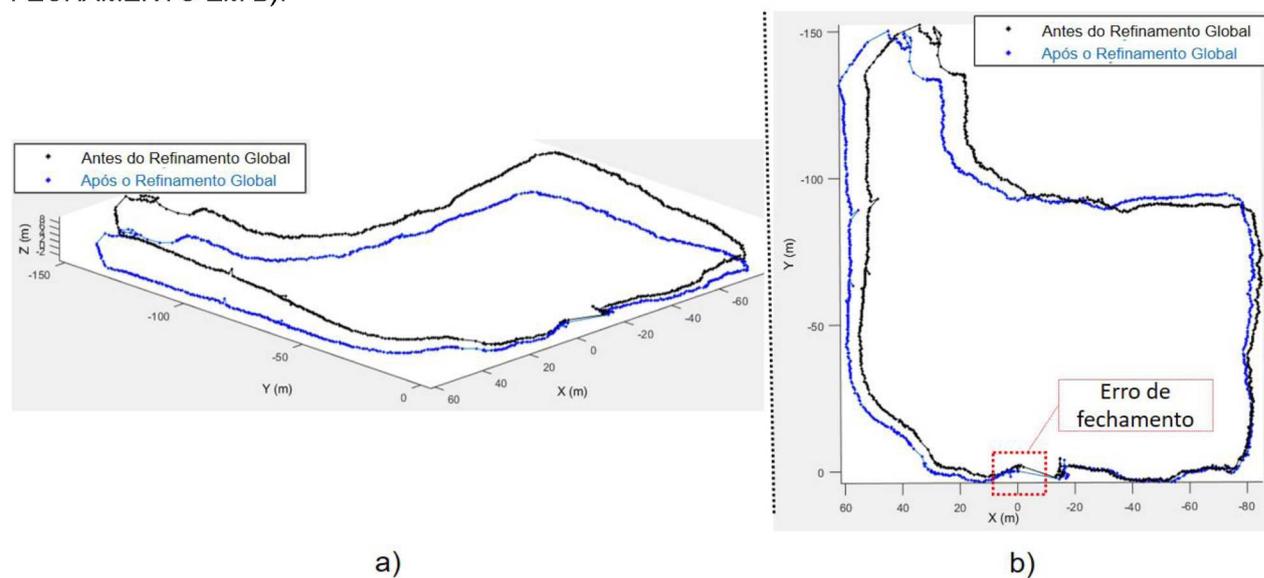
FONTE: O autor (2020)

FIGURA 41 - TRAJETÓRIA OBTIDA COM OS DADOS DE ODOMETRIA ASSOCIADOS AO MÉTODO NDT B) E COM O MÉTODO NDT APENAS A).



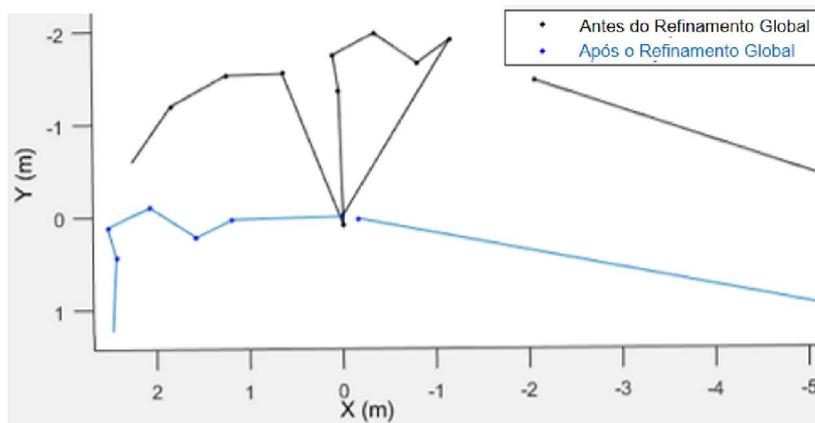
FONTE: O autor (2020)

FIGURA 42 – TRAJETÓRIA OBTIDA ANTES E APÓS A APLICAÇÃO DOS PARÂMETROS DE REFINAMENTO GLOBAL EM DIFERENTES PERSPECTIVAS, COM DESTAQUE AO ERRO DE FECHAMENTO EM B).



FONTE: O autor (2020)

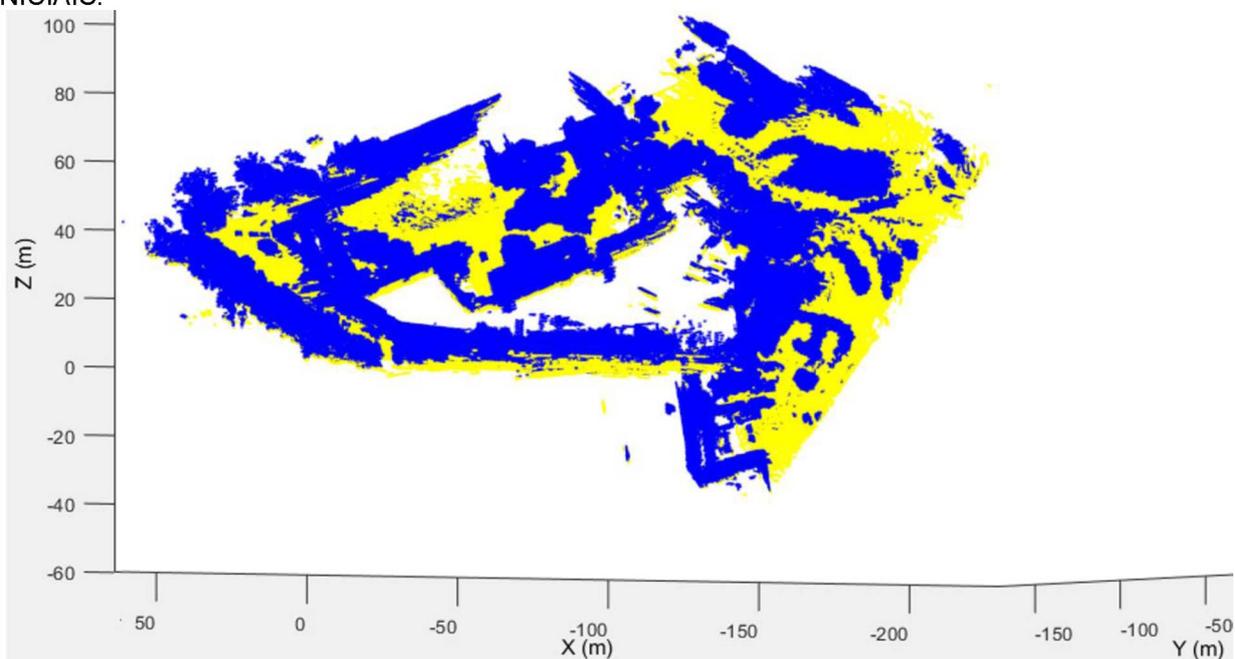
FIGURA 43 – DESTAQUE DA FIGURA 42 COM FOCO NO ERRO DE FECHAMENTO.



FONTE: O autor (2020)

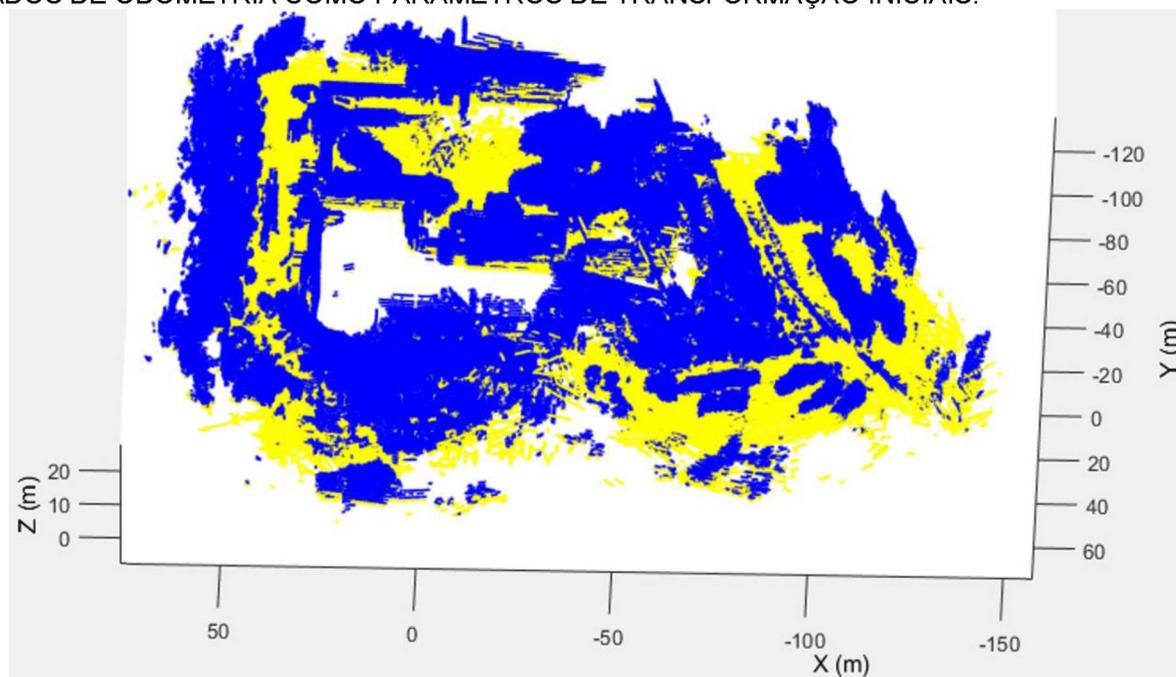
Na Figura 42 é mostrada a trajetória estimada com o método de registro local proposto nesta pesquisa em diferentes perspectivas. Como pode ser visto na Figura 43, que mostra em detalhes o erro de fechamento da trajetória (ver Figura 21), e comprovado na Tabela 5, tanto os valores encontrados antes quanto após o refinamento global são inferiores aos encontrados para todos os outros algoritmos utilizados para fins comparativos.

FIGURA 44 - NUVEM DE PONTOS LIDAR 3D GLOBAL OBTIDA COM O ALGORITMO ICP CONSIDERANDO OS DADOS DE ODOMETRIA COMO PARÂMETROS DE TRANSFORMAÇÃO INICIAIS.



FONTE: O autor (2020)

FIGURA 45 - NUVEM DE PONTOS LIDAR 3D GLOBAL OBTIDA COM O ALGORITMO NDT COM OS DADOS DE ODOMETRIA COMO PARÂMETROS DE TRANSFORMAÇÃO INICIAIS.



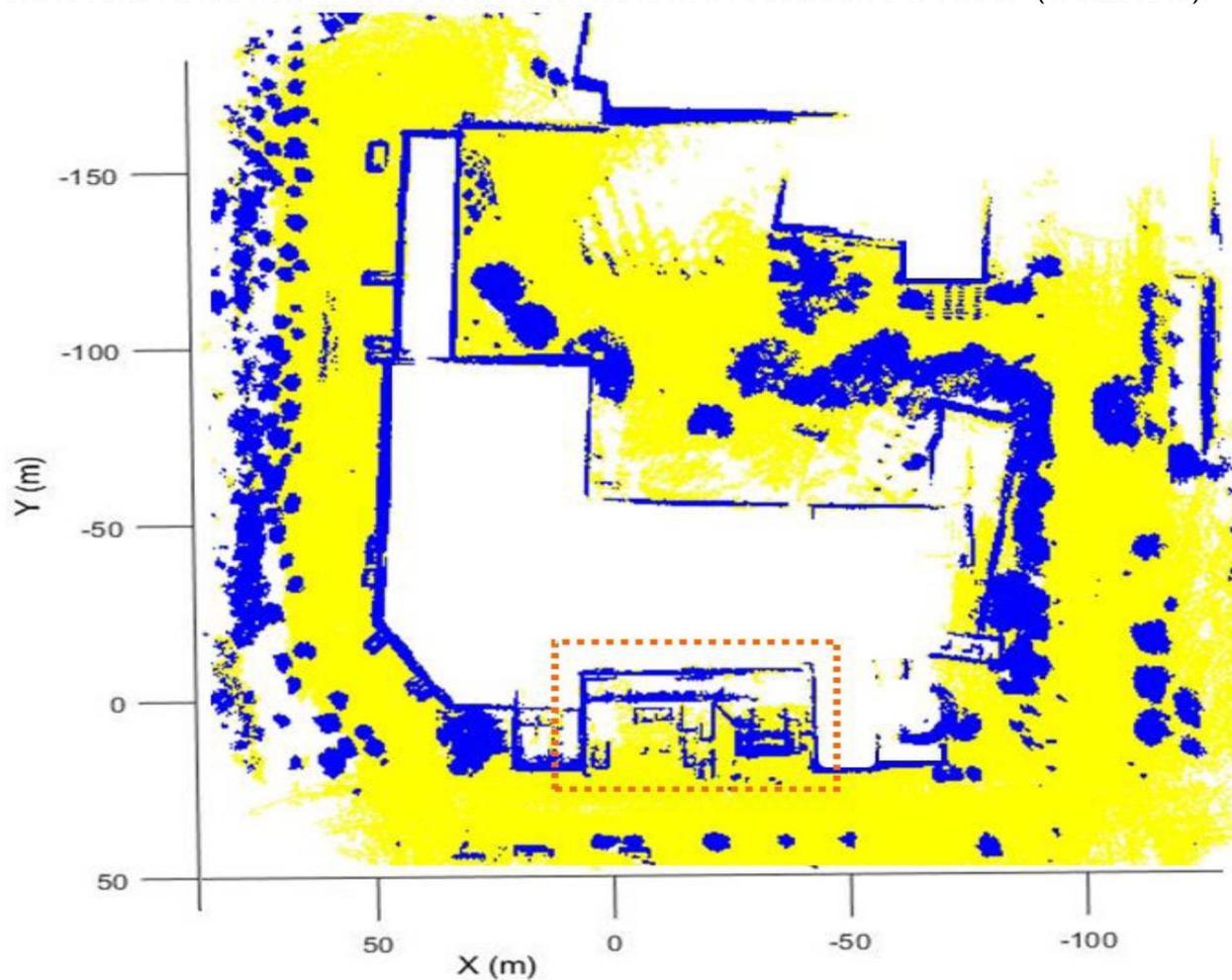
FONTE: O autor (2020)

Essa disparidade evidenciada visualmente pelas Figuras 44 (ICP) e 45 (NDT) com relação a Figura 47 (método proposto) é devido às próprias características dos algoritmos ICP e NDT, baseados em pontos, onde por várias ocasiões falham e os parâmetros de transformação entre as nuvens de pontos LiDAR 3D obtidos resultam em erros grosseiros no processo de refinamento global, levando a uma nuvem global de qualidade inferior, como mostram as Figuras 44 e 45 evidenciadas pelas trajetórias mostradas nas Figuras 40 b) e 41 b), mesmo utilizando dados de odometria como parâmetros de transformação iniciais.

Para fins de visualização, as nuvens de pontos LiDAR 3D mostradas nas Figuras 46 e 47 em diferentes perspectivas foram reamostradas para uma renderização de melhor qualidade resultando em um total de aproximadamente 6,7 milhões de pontos.

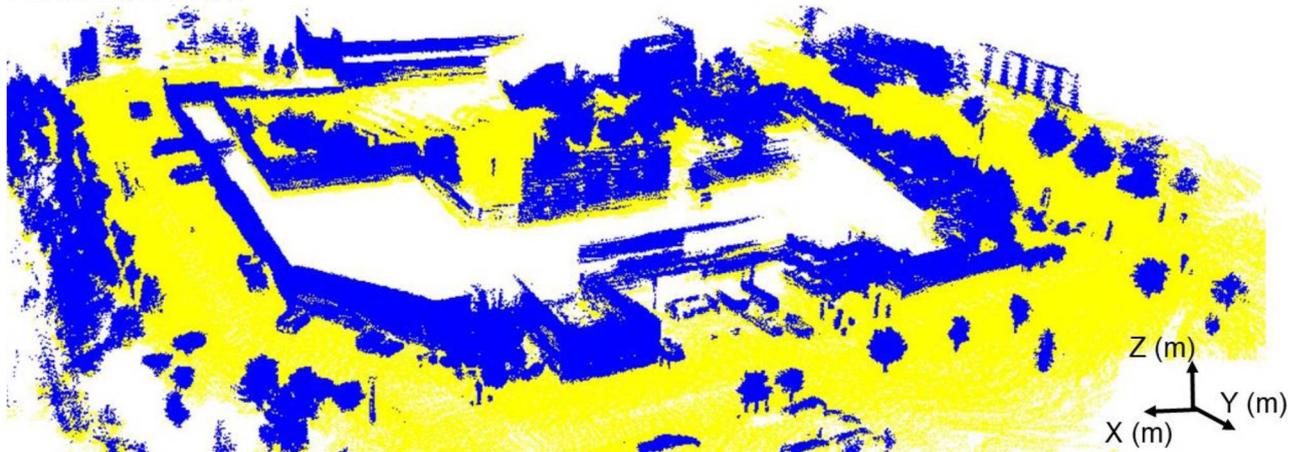
O destaque em vermelho na Figura 46 é ilustrado nas Figuras 48 e 49 em diferentes perspectivas, ao qual também pode ser verificado na Tabela 5 como o erro de fechamento, utilizando o método de registro local proposto entre a primeira (vermelho) e a última (preto) nuvem de pontos LiDAR 3D após o refinamento global.

FIGURA 46 – NUVEM DE PONTOS LIDAR 3D GLOBAL OBTIDA COM O MÉTODO PROPOSTO SOB UMA PERSPECTIVA 2D. COM DESTAQUE À REGIÃO DE FECHAMENTO DO LOOP (VERMELHO).



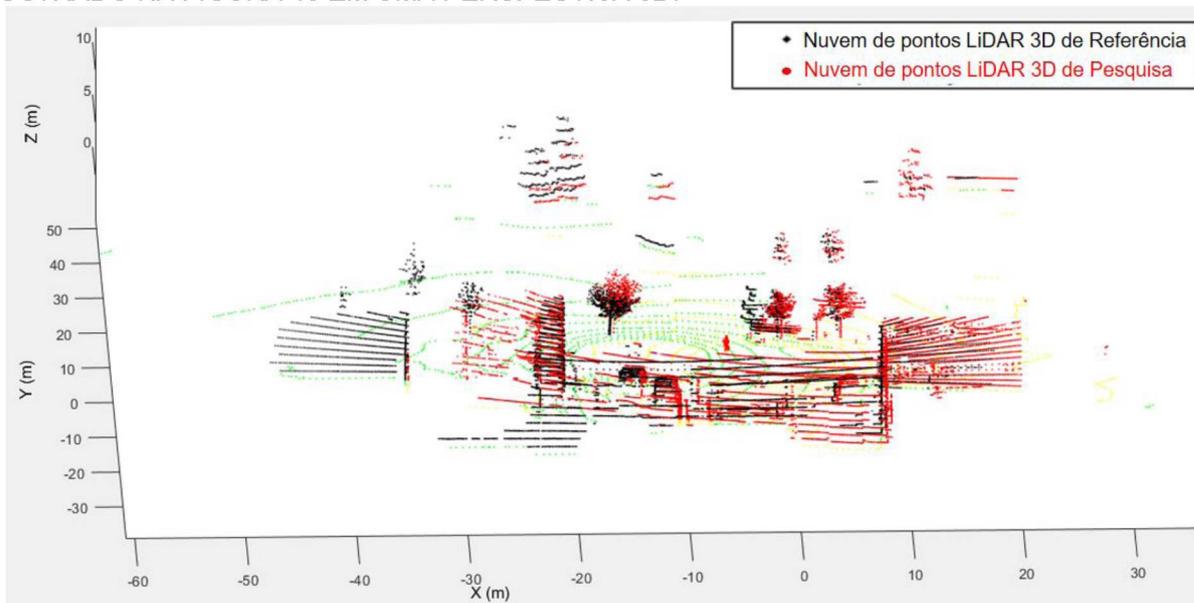
FONTE: O autor (2020)

FIGURA 47 - NUVEM DE PONTOS LIDAR 3D GLOBAL OBTIDA COM O MÉTODO PROPOSTO SOB A PERSPECTIVA 3D.



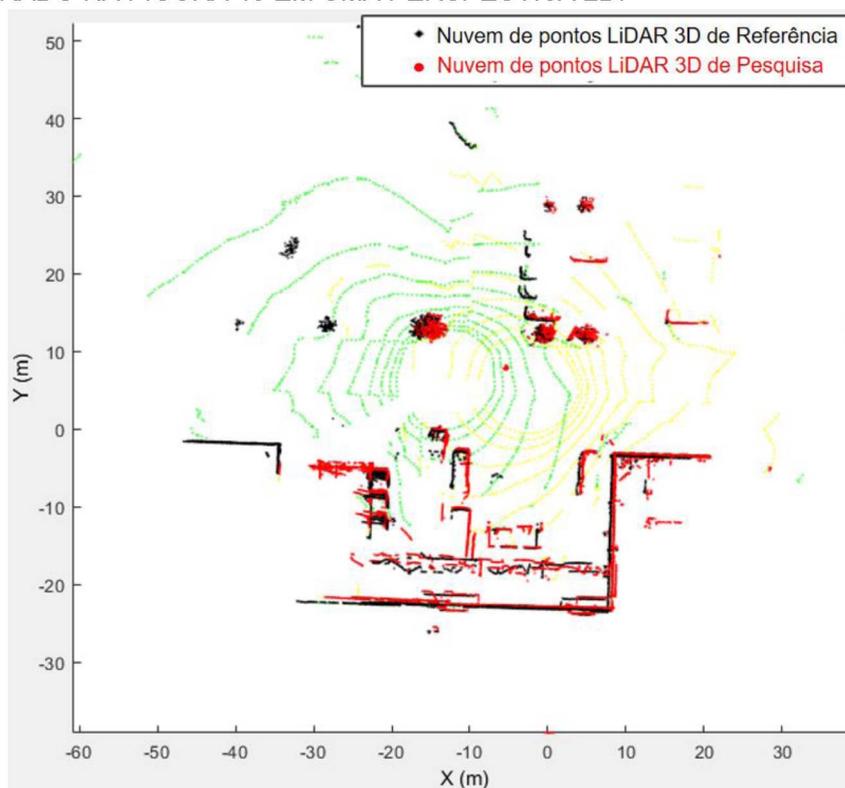
FONTE: O autor (2020)

FIGURA 48 – REGISTRO LOCAL ENTRE A PRIMEIRA (VERMELHO) E A ÚLTIMA (PRETO) NUVEM DE PONTOS LIDAR 3D ILUSTRANDO O ERRO DE FECHAMENTO APÓS O REFINAMENTO GLOBAL MOSTRADO NA FIGURA 43 EM UMA PERSPECTIVA 3D.



FONTE: O autor (2020)

FIGURA 49 - REGISTRO LOCAL ENTRE A PRIMEIRA (VERMELHO) E A ÚLTIMA (PRETO) NUVEM DE PONTOS LIDAR 3D ILUSTRANDO O ERRO DE FECHAMENTO DO LOOP APÓS O REFINAMENTO GLOBAL MOSTRADO NA FIGURA 43 EM UMA PERSPECTIVA 2D.



FONTE: O autor (2020)

TABELA 5 – ERROS ANGULARES E TRANSLACIONAIS DE FECHAMENTO ANTES E APÓS O REFINAMENTO GLOBAL.

Parâmetros		Métodos/Algoritmos				
		ICP	NDT	Odometria e ICP	Odometria e NDT	Método Proposto
t_x (m)	Antes	14,7749	69,6180	45,1678	58,6950	2,0726
	Depois	1,6484	42,7354	32,6660	37,3314	0,1775
t_y (m)	Antes	35,0003	1,6310	14,8862	10,8836	1,4648
	Depois	24,2950	13,2031	11,5096	3,9904	0,0200
t_z (m)	Antes	11,4067	39,3206	21,8976	11,0145	0,0605
	Depois	8,9917	2,4880	6,9240	4,1237	0,0168
ω (°)	Antes	46,4858	25,2965	36,6240	15,4393	10,3565
	Depois	8,1236	7,2255	5,5363	0,6868	0,0366
φ (°)	Antes	15,9672	7,0129	52,4930	11,7772	3,7978
	Depois	22,8862	7,7854	27,4642	13,2491	0,0111
k (°)	Antes	75,9124	63,1057	79,1608	58,6433	3,4609
	Depois	88,9785	78,7640	59,0951	51,3946	0,3344

FONTE: O autor (2020)

As trajetórias mostradas na Figura 40b e na Figura 41b geraram as nuvens globais mostradas nas Figuras 44 e 45 e os erros de fechamento mostrados na Tabela 5 comprovam a ineficácia de algoritmos baseados apenas na primitiva ponto para a realização de um mapeamento urbano com dados obtidos com o sensor LiDAR 3D onde possam haver obstruções do sensor por pedestres ou veículos e definido a própria geometria das linhas de varredura do sensor. Portanto, como pode se visto nos experimentos: REMQ ponto a plano, custo computacional, trajetória (com destaque a Tabela 5) e as nuvens globais; o método proposto nesta pesquisa para o registro local de nuvens de pontos LiDAR 3D se mostrou superior aos algoritmos ICP e NDT em todos os aspectos, com e sem o uso de dados de odometria como parâmetros iniciais. A seguir, as conclusões desta pesquisa e as recomendações para trabalhos futuros.

5. CONCLUSÕES E RECOMENDAÇÕES

Na presente pesquisa é apresentado um método para mapeamento de ambientes urbanos utilizando dados de varredura a LASER móvel terrestre. As principais contribuições do presente trabalho são:

- 1) Proposta de um método para a correspondência entre planos presentes nas nuvens de pontos LiDAR 3D, fundamentado nos algoritmos de rotulação por relaxação, a partir de uma Matriz de Probabilidades *a priori* obtida pela concatenação de funções de probabilidades que consideram características geométricas dos planos, e, a sua posterior atualização por Inferência Bayesiana a partir de relações locais dos planos correspondidos.
- 2) Método analítico de determinação dos parâmetros de transformação entre os pares de nuvens de pontos LiDAR 3D diretamente com os planos e a álgebra de Dual Quaternions.
- 3) Se apenas um Espaço Canônico bidimensional for definido pelos planos, é proposto o uso de *clusters* para criar uma zona de busca de pontos virtuais de interesse que, de forma direta e com a mesma álgebra de Dual Quaternions, determinam os parâmetros de transformação ótimos.
- 4) Caso não se detecte *clusters* com a direção do Espaço Canônico ausente é proposto um método de aproximação chamado de *Sliding*, em que um plano de referência acima do terreno é deslizado sobre seu correspondente até um mínimo global das distâncias dos pontos nas suas extremidades.

O método proposto para a correspondência entre os planos homólogos pelos pares de nuvens de pontos LiDAR 3D a partir de uma Matriz de Probabilidades *a priori* e sua posterior atualização por Inferência Bayesiana apresentou resultados consistentes, dado certo limite de intervalo entre as nuvens. Nenhum tipo de intervenção pelo operador foi necessária para os pares adjacentes, mesmo em situações onde os planos foram demasiadamente degradados pelos processos de filtragem e segmentação. Isso também se deve ao processo de remoção de ambiguidades entre planos, essencial para a correta correspondência dessas feições.

Um problema recorrente em projetos de mapeamento urbano é o Espaço Canônico gerado pelas observações. A Literatura específica utiliza uma MC das normais dos planos e de uma parametrização por SVD. Entretanto, de uma forma mais simples e intuitiva neste trabalho a avaliação do Espaço Canônico é efetuada através do produto interno das normais dos planos. Caso os vetores normais diverjam para todas as direções, um método analítico inédito para a extração dos parâmetros de transformação é empregado por meio da álgebra de Dual Quaternions.

Entretanto, caso as observações formem um Espaço Canônico bidimensional, um método híbrido é proposto para estabelecer uma correspondência biunívoca entre os centroides dos *clusters* na nuvem de pontos LiDAR 3D de Referência e pontos virtuais nas nuvens de pontos LiDAR 3D de Pesquisa por intermédio de relações de distância ortogonais com planos de referência. Esse método atendeu as expectativas diante a pluralidade de cenários e objetos presentes na área de estudo considerada nesta pesquisa.

Porém, nos casos em que os *clusters* estavam indisponíveis, foi aplicada uma técnica de aproximação a partir do plano de referência acima do terreno deslizando-o sobre seu homólogo até alcançar um mínimo global das distâncias das suas extremidades. Devido a deteriorização dos planos pelos processos de filtragem e segmentação, que ocorre justamente nas extremidades dos planos, o valor da REMQ ponto a ponto é condicionado a um limiar pré-estabelecido. Caso não alcance o limiar, apenas a Rotação é aplicada à nuvem de pontos LiDAR 3D de Pesquisa por meio de Quaternions seguido do registro fino com o algoritmo ICP.

O motivo pela qual um Espaço Canônico unidimensional não tenha sido abordado, baseia-se no fato de que em todas as nuvens de pontos LiDAR 3D utilizadas nesta pesquisa foram detectados planos no terreno. Isso se deve ao pressuposto que o terreno por onde a plataforma desenvolveu sua trajetória é regular, ou seja, segmentos planares estão disponíveis, com um certo grau de curvatura, pelo menos atrás da posição da plataforma. Assim, para aqueles trechos onde há fachadas ou paredes, e portanto planos, isso garante um Espaço Canônico bidimensional, uma vez que planos no terreno foram detectados e extraídos. Já em regiões muito arborizadas ou com prédios fora do alcance do sensor, o algoritmo ICP foi aplicado diretamente, uma vez que o uso apenas dos

planos do terreno é inviável devido as características do sensor Velodyne citadas anteriormente.

Os resultados obtidos com os valores da REMQ ponto a plano mostrados no Capítulo 4 com destaque aos erros de fechamento descritos na Tabela 5 após o refinamento global, demonstram o potencial do método proposto, principalmente quando comparado aos resultados obtidos com os algoritmos ICP e NDT, com e sem o uso de dados de Odometria.

Como dito no Capítulo I, a etapa mais onerosa em projetos de mapeamento tridimensional baseados em feições é a segmentação e a extração dessas primitivas, sendo responsáveis por cerca de 90% do custo computacional do método proposto na presente pesquisa. Todos esses processos: remoção de pontos espúrios, voxelização, RANSAC e o algoritmo Crescimento de Região; apresentaram ótimos resultados para as nuvens de pontos LiDAR 3D obtidas com o sensor Velodyne, entretanto, esses dois últimos processos são relativamente onerosos e necessitam de alguns ajustes para lidar principalmente com planos no terreno. Assim, uma importante recomendação para trabalhos futuros é a aplicação de métodos que exploram a própria geometria das linhas de varredura dos sensores LASER do Velodyne, como proposto em Borrmann *et al*, (2011) e Grant *et al*, (2013).

Além disso, também é uma recomendação para trabalhos futuros o uso de alguma técnica baseada em pontos de interesse adaptada aos dados obtidos com o sensor LiDAR 3D para lidar com regiões onde não há planos disponíveis.

Embora eficaz para este conjunto de nuvens de pontos LiDAR 3D, o método LUM utilizado para o refinamento global dos parâmetros de transformação estabelece, pelo critério de proximidade, correspondências entre pontos para o processo de estabelecimento de injunções no grafo. Além de não ser indicado para nuvens de pontos LiDAR 3D obtidas com o Velodyne, o processo é consideravelmente oneroso. Assim, uma recomendação para trabalhos futuros é o uso de métodos mais robustos e eficientes para o refinamento global.

6. REFERÊNCIAS

- AIGER, D.; MITRA, N. J.; COHEN-OR, D. 4-Points Congruent Sets for Robust Pairwise Surface Registration. **ACM Transactions on Graphics**. v. 27, n. 3, p. 1, 2008.
- AKCA, D.; FREEMAN, M.; SARGENT, I.; GRUEN, A. Quality assessment of 3D building data. **The Photogrammetric Record**. v. 25. ed. 132, 2010.
- ARUN K. S.; HUANG T. S.; BLOSTEIN S. D. Least Square fitting of two 3-D point sets. **IEEE trans. Patt. Anal. Machine Intell.** v. PAMI-9, n. 95, p. 698-700, 1987.
- BAE, K.; LICHTI, D. D. A Method for Automated registration of Unorganised Point Clouds. **ISPRS Journal of Photogrammetry and Remote Sensing**. v. 63, p. 36- 54, 2008.
- BERGEN, G. Dual Numbers: Simple Math, Easy C++ Coding, and Lots of Tricks, GDC Europe, 2009. Disponível em: <www.gdcvault.com/play/10103/Dual-NumbersSimple-Math-Easy>. Acesso em 03 de Maio de 2018.
- BESL, P.; MCKAY, N. A Method for Registration of 3-D Shapes. **IEEE Transactions on Pattern Analysis and Machine Intelligence**. v. 14, p. 239-256, 1992.
- BIBER, P.; STRASSER, W. The Normal Distributions Transform: A New Approach To Laser Scan Matching. **Proceedings of IEEE/Rsj International Conference On Intelligent Robots And Systems (Iros)**. v. 3, p. 2743–2748, 2003.
- BLAIS G.; LEVINE M. D. Registering multiview range data to create 3D computer objects. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 17, n. 8, p. 820-824, 1995.
- BLANCO, J. L.; GONZALEZ, J.; FERNANDEZ-MADRIGAL, J. A. Optimal Filtering for Non-Parametric Observation Models: Applications to Localization and SLAM. **The International Journal of Robotics Research (IJRR)**. 29 (14): p. 1726–1742, 2010.
- BORRMANN, D.; ELSEBERG, J.; LINGEMANN, K.; NÜCHTER, A.; HERTZBERG, J. Globally consistent 3D mapping with scan matching. **Robotics and Autonomous Systems**. v. 56, n. 2, p. 130–142, 2008.
- CHAN, T.; LICHTI, D. D. Article Automatic In Situ Calibration Of A Spinning Beam LiDAR System In Static And Kinematic Modes. **Remote Sensing**, v. 7, p. 10480-10500, 2015.
- CHASLES, M. Note Sur Les Propriétés Générales Du Système De Deux Corps Semblables Entr'eux Et Placés D'une Manière Quelconque Dans L'espace. **Et Sur Le Déplacement Fini Ou Infiniment Petit D'un Corps Solide Libre**. Bull. Sci. Math. Astron. Phys. Chim. 14 (Xiv), p. 321–326, 1830.

CHEN, Y.; MEDIONI, G. Object modeling by registration of multiple range images. **IEEE International Conference on Robotics and Automation Image**. Sacramento CA. Vision Comput., v. 10, p. 145-155, 1992.

CLIFFORD, W. *Mathematical Papers*. London: Macmillan, 1882.

CLIFFORD, W. K. Preliminary Sketch Of Bi-Quaternions. **Proceedings Of The London Mathematical Society**. S1-4(1), p. 381-395, 1873.

DESCHAUD, E. J. IMLS-SLAM: scan-to-model matching based on 3D data. **Proceedings - IEEE International Conference on Robotics and Automation**, p. 2480-2485, 2018.

DOS SANTOS, D. R.; BASSO, M. A.; KHOSHELHAM, K.; OLIVEIRA JR., E. M.; PAVAN, N. L.; VOSSelman, G. Mapping indoor spaces by adaptive coarse-to-fine registration of RGB-D data. **IEEE geoscience and remote sensing letters**, v. 13, ed. 2, p. 262-266, 2016.

DREAMSTIME. Imagem de fundo. Créditos: Elena Pimukova. ID: 113309995. 2020. Disponível em: <<https://www.dreamstime.com>>. Acesso em 04/05/2020.

FISCHLER, M. A; BOLLES, R. C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. **Communications of the ACM**. v. 24, n. 6, p. 381-395, 1981.

FRIEDMAN, J. H.; BENTLEY, J. L.; FINKEL, R. A. An algorithm for finding best matches in logarithmic expected time. **ACM Transaction on Mathematical Software**. 3(3), p. 209-226, 1977.

GALO, M; TOZZI, C. A representação de matrizes de rotação e o uso de quatérnios em ciências geodésicas. **Série em Ciências Geodésicas - 30 anos da Pós-Graduação em Ciências Geodésicas**. Mitishita, E. A. (Editor), v-1, p. 214-231, 2001.

GEMAEL, C. Introdução ao ajustamento de observações: Aplicações geodésicas. Curitiba, **Editora UFPR**, 1994.

GRANT, D. S.; BETHEL, J.; CRAWFORD, M. Point-to-plane registration of terrestrial LASER scans. **ISPRS Journal of Photogrammetry and Remote Sensing**, 72(0), p. 16-26, 2012.

GRANT, D. S. Cloud To Cloud Registration For 3d Point Data. Tese apresentada a Universidade de Purdue, Indiana, EUA, 2013.

GRANT, W. S.; VOORHIES, R. C.; ITTI, L. Finding planes in LiDAR point clouds for real-time registration. In: **2013 IEEE/RSJ international conference on intelligent robots and systems (IROS)**, p. 4347-4354, 2013.

GRANT, W. S.; · RANDOLPH C.; VOORHIES, R. C.;· ITTI, L. Efficient Velodyne SLAM with point and plane features. **Autonomous Robots**. Springer, 2019.

GRESSIN, A.; MALLET, C.; DEMANTKÉ, J.; DAVID, N. Towards 3D LiDAR point cloud registration improvement using optimal neighborhood knowledge. **ISPRS Journal of Photogrammetry and Remote Sensing**, v. 19, p. 240-251, 2013

GRISSETTI, G.; GRZONKA, S.; STACHNISS, C.; PFAFF, P.; BURGARD, W. Efficient Estimation of Accurate Maximum Likelihood Maps in 3D. **In Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**, 2007.

GRISSETTI, G.; KUMMERLE, R.; STACHNISS, C.; BURGARD, W. A tutorial on graph-based SLAM. **IEEE Intelligent Transportation Systems Magazine**. v. 2, n. 4, p. 31–43, 2010.

GRUEN, A.; AKCA, D. Least squares 3D surface and curve matching. **ISPRS Journal of Photogrammetry and Remote Sensing**. v. 59, n. 3, p. 151-174, 2005.

HAMILTON, W. R. Elements Of Quaternions. **Longmans, Green & Co.**, London. 1886.

HORN, B. K. P. Closed-form solution of absolute orientation using unit quaternions. **Journal of the Optical Society of America**. vol. 4, p. 629–642, 1987.

HUANG, G. P.; MOURIKIS, A. I.; ROUMELIOTIS, S. I. Analysis and improvement of the consistency of extended Kalman filter based SLAM. **Robotics and Automation**. ICRA 2008. IEEE International Conference on. p. 473–479, 2008.

HUMMEL, R. A; ZUCKER, S. W. On the foundations of relaxation labeling processes. **IEEE-PAMI**, v. 5, n. 3, p. 267-287, 1983.

IEZZI, G. Fundamentos De Matemática Elementar. Geometria Analítica - Volume 7.Ed Atual, 2006.

KENWRIGHT, B. A Beginners Guide To Dual-Quaternions What They Are, How They Work, And How To Use Them For 3D Character Hierarchies. **School Of Computing Science**, Newcastle University, 2020. Disponível em: <https://Cs.Gmu.Edu/~Jmlie/Teaching/Cs451/Uploads/Main/Dual-uaternion.Pdf>>. Acesso em 06 Agosto de 2019.

KOTELNIKOV, A. P. Screw calculus and some applications to geometry and mechanics. *Annal. Imp. Univ. Kazan*, 1895.

KUMMERLE, R.; GRISSETTI, G.; STRASDAT, H.; KONOLIGE, K.; BURGARD, W. g2o: A General Framework for Graph Optimization 2011. **IEEE International Conference on Robotics and Automation**. Shanghai International Conference Center May 9-13, Shanghai, China, 2011.

LICHTI, D., PFIEFER, N., MAAS, H.-G. ISPRS Journal of Photogrammetry and Remote Sensing theme issue "Terrestrial Laser Scanning". **ISPRS Journal of Photogrammetry and Remote Sensing** **63 (1)**, 1–3, 2008.

LOW, K. Linear Least-Squares Optimization For Point-To-Plane ICP Surface Registration. **Technical Report Tr04-004**. Department Of Computer Science, University Of North Carolina At Chapel Hill, 2004.

LU, F.; MILIOS, E. Globally consistent range scan alignment for environment mapping. **Autonomous Robots** **4**. p. 333–349, 1997.

MASUDA, T.; SAKAUE, K.; YOKOYA, N. Registration and Integration of Multiple Range Images for 3-D Model Construction. In: **13^a International Conference on Pattern Recognition**. v. 1, p. 879-883, 1996.

MCAULAY, A. Octonions: A Development Of Clifford's Biquaternions, 1898.

MCCARTHY, J. M. An Introduction To Theoretical Kinematics, **MIT PRESS**, p. 62–5, 1990.

MICHIGAN. NCLT Dataset: The University of Michigan North Campus Long-Term Vision and LIDAR Dataset. Por Nicholas Carlevaris-Bianco, Arash Ushani e Ryan M. Eustice, 2012. Disponível em: <<http://robots.engin.umich.edu/nclt/>>. Acesso em: 22 de junho de 2019.

NUCHTER, A.; LINGEMANN, K.; HERTZBERG, J. Cached k-d tree Search for ICP Algorithms. In: **Proc. of the IEEE Conference 3-D Digital Imaging and Modeling**. p. 419-426, 2007.

PATHAK, K.; BIRK, A.; VASKEVICIUS, N. Online three-dimensional SLAM by registration of large planar surface segments and closed-form graph relaxation. **Journal of Field Robotics**. v. 27, n. 1, p. 52–84, 2010.

PAVAN, N. L.; DOS SANTOS, D. R. A Global Closed-Form Refinement for Consistent TLS Data Registration. **IEEE GEOSCIENCE AND REMOTE SENSING LETTERS**. v. 14, n. 7, p. 1131–1135, 2017.

PFEIFER, N.; BÖHM, J. Early stages of LiDAR data processing. Advances in Photogrammetry, Remote Sensing and Spatial Information Sciences: **ISPRS Congress Book**, 2008.

RABBANI, T.; VAN DEN HEUVEL, F.; VOSSELMANN, G. Segmentation of point clouds using smoothness constraint. **International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences**, v. 36, n.5, p. 248-253, 2006.

RUSINKIEWICZ, S.; LEVOY, M. Efficient variants of the ICP algorithm. **Proceedings of Third International Conference on 3-D Digital Imaging and Modeling**. IEEE Computer Soc., p. 145-152, 2001.

RUSU. C. PCL (Point Cloud Library), 2011. Disponível em: <<https://pointclouds.org/>>. Acesso em 26 de junho de 2018.

SANDERSON, C.; CURTIN, R. Armadillo: a template-based C++ library for linear algebra. **Journal of Open Source Software**, v. 1, p. 26, 2016.

SCHWARZ, B. Mapping The World In 3D. **Nat. Photonics**, 4, p. 429-430, 2010.

SEGAL, A.; HAEHNEL, D.; THRUN, S. Generalized-ICP. In: **Robotics: Science and Systems**. 2009.

SHAN, J. TOTH, C. Topographic Laser Ranging and Scanning: Principles and Processing. **CRC Press Taylor & Francis Group**, Boca Raton, FL, 33487-2742, 2010.

SIMON J. J.; UHLMANN, J. K. Using covariance intersection for SLAM. **Robotics and Autonomous Systems**, 2007.

SLAMA, C. C.; THEURER, C.; HENRIKSEN S. W. Manual Of Photogrammetry. **American Society Of Photogrammetry**. Falls Church, Va., 1980.

STUDY, E.; BEWEGUNGEN, V. D. *Mathematische Annalen*, v. 39: p. 441–566, 1891.

THEILER, P. W.; WEGNER, J. D.; SCHINDLER, K. Keypoint-based 4-Points Congruent Sets - Automated marker-less registration of laser scans. **ISPRS Journal of Photogrammetry and Remote Sensing**. v. 96, p. 149–163, 2014.

VELODYNE HDL-32E. User's Manual. 2012. Disponível em: <http://Velodynelidar.Com/Lidar/Products/Manual/63-9113%20hdl-32e%20manual_Rev%20e_Nov2012.Pdf>. Acesso em 01 de agosto de 2019.

VLAMINCK, M.; LUONG, H.; PHILIPS, W. Have I Seen This Place Before? A Fast and Robust Loop Detection and Correction Method for 3D LiDAR SLAM. **Sensors**, 19(1), 23, 2019.

WALKER, M. W.; SHAO, L.; VOLZ, R. A. Estimating 3-D location parameters using dual number quaternions. **CVGIP: Image Understanding**. 54 (3), p. 358–367, 1991.

WANG, Y.; WANG Y.; WU, K.; YANG, H., ZHANG, H. A Dual Quaternion-Based, Closed-Form Pairwise Registration Algorithm For Point Clouds. **International Society for Photogrammetry and Remote Sensing**. v. 94 p. 63–69, 2014.

WEHR, A.; LOHR, U. Airborne Laser Scanning: an Introduction and Overview, **ISPRS Journal Of Photogrammetry And Remote Sensing**. (54) 2-3, p. 68-82, 1999.

YAN-BIN, Jia. Dual Quaternions. 2020. Disponível em: <<http://web.cs.iastate.edu/~cs577/handouts/dual-quaternion.pdf>>. Acesso em 03 de agosto de 2017.

APÊNDICE

Dito que:

$$[L_{\vec{n}_i^r}]Q = [L_q] \begin{bmatrix} 0 \\ \vec{n}_i^r \end{bmatrix}$$

Assim,

$$\begin{aligned} W_4 &= \sum_{i=1}^n \left([L_{\vec{n}_i^r}]Q Q^T [L_{\vec{n}_i^r}]^T \right) = \sum_{i=1}^n \left([L_{\vec{n}_i^r}]Q \left([L_{\vec{n}_i^r}]Q \right)^T \right) = \\ &= \sum_{i=1}^n \left([R_q] \begin{bmatrix} 0 \\ \vec{n}_i^r \end{bmatrix} \left([R_q] \begin{bmatrix} 0 \\ \vec{n}_i^r \end{bmatrix} \right)^T \right) = \sum_{i=1}^n \left([R_q] \begin{bmatrix} 0 \\ \vec{n}_i^r \end{bmatrix} [0 \quad \vec{n}_i^{rT}] [R_q]^T \right) = \\ &= [R_q] \left(\sum_{i=1}^n \left(\begin{bmatrix} 0 \\ \vec{n}_i^r \end{bmatrix} [0 \quad \vec{n}_i^{rT}] \right) \right) [R_q]^T = [R_q] \left(\sum_{i=1}^n \left(\begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \vec{n}_i^r \vec{n}_i^{rT} \end{bmatrix} \right) \right) [R_q]^T = \\ &= [R_q] \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \left(\sum_{i=1}^n \vec{n}_i^r \vec{n}_i^{rT} \right) \end{bmatrix} [R_q]^T = [R_q] \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & A^T A \end{bmatrix} [R_q]^T = [R_q] \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & N \end{bmatrix} [R_q]^T \end{aligned}$$

com,

$$A = \begin{bmatrix} \vec{n}_1^{rT} \\ \vdots \\ \vec{n}_n^{rT} \end{bmatrix}$$

Sabendo que $[R_q] \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix} = Q$ assim, com base na Equação 115, fica estabelecido que:

$$\begin{bmatrix} (W_4 + W_4^T) & Q \\ Q^T & 0 \end{bmatrix} = \begin{bmatrix} 2W_4 & Q \\ Q^T & 0 \end{bmatrix} = [R_q] \begin{bmatrix} 0 & \mathbf{0}^T & 1 \\ \mathbf{0} & 2N & \mathbf{0} \\ 1 & \mathbf{0}^T & 0 \end{bmatrix} [R_q]^T$$

$$\left(\begin{bmatrix} 0 & \mathbf{0}^T & 1 \\ \mathbf{0} & 2N & \mathbf{0} \\ 1 & \mathbf{0}^T & 0 \end{bmatrix} \right)^{-1} = \begin{bmatrix} b_{11} & \mathbf{b}_1^T & b_{15} \\ \mathbf{b}_2 & B & \mathbf{b}_3 \\ b_{51} & \mathbf{b}_4^T & b_{55} \end{bmatrix}$$

$$\begin{bmatrix} 1 & \mathbf{0}^T & 0 \\ \mathbf{0} & I & \mathbf{0} \\ 0 & \mathbf{0}^T & 1 \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{0}^T & 1 \\ \mathbf{0} & 2N & \mathbf{0} \\ 1 & \mathbf{0}^T & 0 \end{bmatrix} \begin{bmatrix} b_{11} & \mathbf{b}_1^T & b_{15} \\ \mathbf{b}_2 & B & \mathbf{b}_3 \\ b_{51} & \mathbf{b}_4^T & b_{55} \end{bmatrix} = \begin{bmatrix} b_{51} & \mathbf{b}_4^T & b_{55} \\ 2N\mathbf{b}_2 & 2NB & 2N\mathbf{b}_3 \\ b_{11} & \mathbf{b}_1^T & b_{15} \end{bmatrix}$$

sendo,

$$\begin{cases} b_{51} = 1 \\ \mathbf{b}_4 = \mathbf{0} \\ b_{55} = 0 \\ 2N\mathbf{b}_2 = \mathbf{0} \\ 2NB = I \\ 2N\mathbf{b}_3 = \mathbf{0} \\ b_{11} = 0 \\ \mathbf{b}_1 = \mathbf{0} \\ b_{15} = 1 \end{cases}$$

Por outro lado,

$$\begin{bmatrix} 1 & \mathbf{0}^T & 0 \\ \mathbf{0} & I & \mathbf{0} \\ 0 & \mathbf{0}^T & 1 \end{bmatrix} = \begin{bmatrix} b_{11} & \mathbf{b}_1^T & b_{15} \\ \mathbf{b}_2 & B & \mathbf{b}_3 \\ b_{51} & \mathbf{b}_4^T & b_{55} \end{bmatrix} \begin{bmatrix} 0 & \mathbf{0}^T & 1 \\ \mathbf{0} & 2N & \mathbf{0} \\ 1 & \mathbf{0}^T & 0 \end{bmatrix} = \begin{bmatrix} b_{15} & 2\mathbf{b}_1^T N & b_{11} \\ \mathbf{b}_3 & 2BN & \mathbf{b}_2 \\ b_{55} & 2\mathbf{b}_4^T N & b_{51} \end{bmatrix}$$

sendo,

$$\begin{cases} b_{15} = 1 \\ 2N\mathbf{b}_1 = \mathbf{0} \\ b_{11} = 0 \\ \mathbf{b}_3 = \mathbf{0} \\ 2BN = I \\ \mathbf{b}_2 = \mathbf{0} \\ b_{55} = 0 \\ 2N\mathbf{b}_4 = \mathbf{0} \\ b_{51} = 1 \end{cases}$$

Logo,

$$\begin{bmatrix} b_{11} & \mathbf{b}_1^T & b_{15} \\ \mathbf{b}_2 & B & \mathbf{b}_3 \\ b_{51} & \mathbf{b}_4^T & b_{55} \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{0}^T & 0 \\ \mathbf{0} & \frac{1}{2}N^{-1} & \mathbf{0} \\ 0 & \mathbf{0}^T & 1 \end{bmatrix}$$

Temos assim:

$$W_5 Q = \left(\sum_{i=1}^n ((d_i^r - d_i^p) [L_{\vec{n}_i^r}]) \right) Q = \sum_{i=1}^n ((d_i^r - d_i^p) [L_{\vec{n}_i^r}] Q) =$$

$$\begin{aligned}
&= \sum_{i=1}^n \left((d_i^r - d_i^p) [R_q] \begin{bmatrix} 0 \\ \vec{n}_i^r \end{bmatrix} \right) = \begin{bmatrix} 0 \\ [R_q] \left(\sum_{i=1}^n (d_i^r - d_i^p) \vec{n}_i^r \right) \end{bmatrix} = \\
&= [R_q] \begin{bmatrix} 0 \\ \left(\sum_{i=1}^n \vec{n}_i^r (d_i^r - d_i^p) \right) \end{bmatrix} = [R_q] \begin{bmatrix} 0 \\ A^T L \end{bmatrix}
\end{aligned}$$

Com,

$$L = \begin{bmatrix} d_1^r - d_1^p \\ \vdots \\ d_n^r - d_n^p \end{bmatrix}$$

Portanto, dada a Equação 116:

$$\begin{aligned}
\begin{bmatrix} \hat{Q} \\ \lambda \end{bmatrix} &= \left(\begin{bmatrix} (W_4 + W_4^T) & Q \\ Q^T & \mathbf{0} \end{bmatrix} \right)^{-1} \begin{bmatrix} -W_5 Q \\ 0 \end{bmatrix} = \\
&= \left([R_q] \begin{bmatrix} 0 & \mathbf{0}^T & 1 \\ \mathbf{0} & 2N & \mathbf{0} \\ 1 & \mathbf{0}^T & 0 \end{bmatrix} [R_q]^T \right)^{-1} [R_q] \begin{bmatrix} 0 \\ A^T L \\ 0 \end{bmatrix} = \\
&= [R_q] \left(\begin{bmatrix} 0 & \mathbf{0}^T & 1 \\ \mathbf{0} & 2N & \mathbf{0} \\ 1 & \mathbf{0}^T & 0 \end{bmatrix} \right)^{-1} [R_q]^T [R_q] \begin{bmatrix} 0 \\ A^T L \\ 0 \end{bmatrix} = \\
&= [R_q] \begin{bmatrix} 1 & \mathbf{0}^T & 0 \\ \mathbf{0} & \frac{1}{2} N^{-1} & \mathbf{0} \\ 0 & \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} 0 \\ A^T L \\ 0 \end{bmatrix} = [R_q] \begin{bmatrix} 0 \\ \frac{1}{2} N^{-1} A^T L \\ 0 \end{bmatrix}
\end{aligned}$$

Assim, fica demonstrado que a translação T é determinada a partir das seguintes relações:

$$T = 2[R_q]^T \hat{Q} = 2[R_q]^T [R_q] \begin{bmatrix} 0 \\ \frac{1}{2} N^{-1} A^T L \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ N^{-1} A^T L \end{bmatrix}$$